

# Exercises Database Design (DBD)

January 28, 2025

Universitat Politècnica de Catalunya (BarcelonaTech), Barcelona



# Contents

1	Relational Translation . . . . .	5
1.1	Problems . . . . .	5
1.1.1	Relationships . . . . .	6
2	Normalization . . . . .	8
2.1	Problems . . . . .	8
3	Data Warehousing and OLAP . . . . .	11
3.1	Theoretical questions . . . . .	11
3.2	Problems . . . . .	11
3.2.1	Grouping Sets . . . . .	13
4	Not Only SQL . . . . .	16
4.1	Theoretical questions . . . . .	16
4.2	Problems . . . . .	16
5	Views . . . . .	17
5.1	Theoretical questions . . . . .	17
5.2	Problems . . . . .	17
6	Physical Design (Access structures) . . . . .	24
6.1	Theoretical questions . . . . .	24
6.2	Problems . . . . .	24
7	Query Optimization . . . . .	26
7.1	Theoretical questions . . . . .	26
7.2	Problems . . . . .	26
7.2.1	Joins . . . . .	33
7.2.2	Bitmap indexes . . . . .	35
8	Parametrization and Tuning . . . . .	38
8.1	Problems . . . . .	38
9	Transactions . . . . .	40
9.1	Theory . . . . .	40
9.2	Problems . . . . .	40



# 1 Relational Translation

## 1.1 Problems

1. Consider the two relational schemas below (primary key attributes underlined). Would the instances of these schemas represent the **same reality**? Briefly explain why and exemplify it.
  - a) Flight(flightNum: String, origin: Airport, destination: Airport)
  - b) FlightEnd(flightNum: String, type: {Origin, Destination}, place: Airport)
2. Give an example of schematic discrepancy (different from that in the slides), where data in one source is represented as metadata in the other. Draw and briefly explain it.
3. (Solució disponible) Donat el funcionament de la lògica de tres valors, dóna com quedarien les taules de veritat de les operacions XOR (o exclusiva) i la implicació.

	B			
A				

A XOR B

	B			
A				

A  $\Rightarrow$  B

Quines de les propietats següents es mantenen en la lògica de tres valors? Per a cada una, indica SÍ o NO:

- $A \text{ XOR } B \equiv (A \text{ OR } B) \text{ AND NOT}(A \text{ AND } B)$ :
- $A \Rightarrow B \equiv \text{NOT } A \text{ OR } B$ :
- $\text{NOT}(A \text{ AND } B) \equiv \text{NOT } A \text{ OR NOT } B$ :
- $\text{NOT}(A \text{ OR } B) \equiv \text{NOT } A \text{ AND NOT } B$ :

Indica breument com has comprovat si es mantenen les propietats.

4. (Solució disponible) Transformeu les sentències següents en unes d'equivalents que no facin servir RIGHT OUTER JOIN. Si no és possible, indiqueu-ho explícitament.

```
SELECT * FROM T1 RIGHT JOIN T2 ON (a1 = a2) LEFT JOIN T3 ON (b2 = b3)
```

```
SELECT * FROM T1 RIGHT JOIN T2 ON (a1 = a2) RIGHT JOIN T3 ON (b2 = b3)
```

```
SELECT * FROM T1 RIGHT JOIN T2 ON (a1 = a2) FULL JOIN T3 ON (b2 = b3)
```

```
SELECT * FROM T1 LEFT JOIN T2 ON (a1 = a2) RIGHT JOIN T3 ON (b2 = b3)
```

```
SELECT * FROM T1 FULL JOIN T2 ON (a1 = a2) RIGHT JOIN T3 ON (b2 = b3)
```

5. Considerant les dues taules donades, digues si les dues **consultes són equivalents**. Si creus que ho són, justifica per què; i si creus que no ho són, dona un contraexemple.

```
CREATE TABLE t1 (a int, b int);
CREATE TABLE t2 (a int, b int);

SELECT * FROM t1 LEFT OUTER JOIN t2 ON t1.a=t2.a AND t1.b=t2.b;

SELECT * FROM t1 LEFT OUTER JOIN t2 ON t1.a=t2.a WHERE t1.b=t2.b;
```

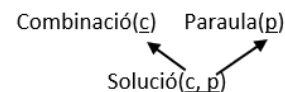
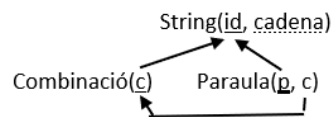
6. Suposa que definim un surrogate amb la sentència: “CREATE SEQUENCE s START WITH 1 INCREMENT BY 1;”. És possible que la taula “t” contingui només les tuples 1 i 3 si sempre s’ha utilitzat la sentència “INSERT INTO t VALUES (s.NEXTVAL);”, no s’ha fet “s.NEXTVAL” en cap altra situació, no s’ha fet cap “ALTER SEQUENCE” i no s’ha fet cap “DELETE”, ni cap “UPDATE” de la taula? Si la resposta és afirmativa, explica com i si és negativa, explica perquè.

### 1.1.1 Relationships

1. El Paraulògic és un joc de cadenes de caràcters que consisteix en, donada una tira de caràcters que anomenem combinació, trobar paraules formades per caràcters de la combinació que anomenem solucions de la combinació. Per exemple, algunes solucions de la combinació MHITOAP són HIPOPOTAM, MAI, MATA i algunes de ACTMEBI són MATA, TAMBE, MAI.

Volem un esquema relacional per guardar combinacions i paraules que en són solució però, per evitar repetir cadenes, volem assignar un identificador a cadascuna que és el que farem servir on calgui que aparegui la cadena identificada.

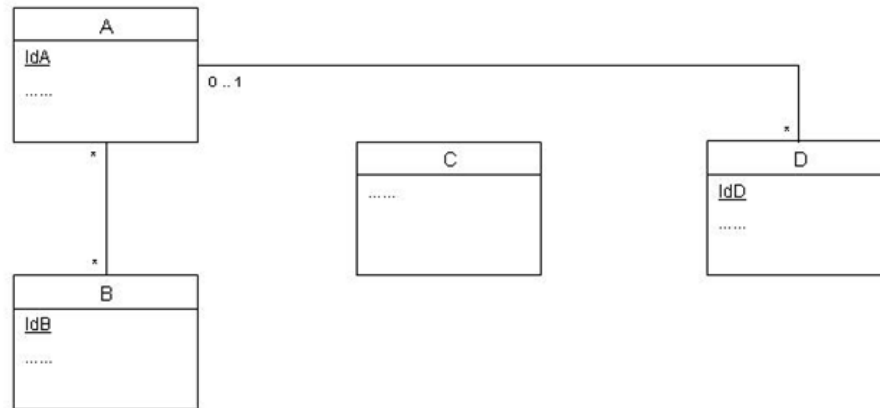
Considereu aquestes dues propostes:



Dibuixeu, per a cadascuna, el diagrama de classes corresponent:

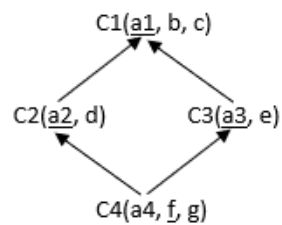
Si us hi fixeu, veureu que cap de les dues propostes és una bona solució. Dibuixeu el diagrama de classes correcte i doneu l’esquema relacional que li correspon

2. (Solució disponible) Donats el models conceptual i lògic següents, **afegiu** a cadascun el que calgui per a fer que el segon sigui una traducció del primer.



A(IdA, ...)  
 B(IdB, ...)  
 C(IdA, IdB, ...) FK: IdA  $\rightarrow$  A, FK: IdB  $\rightarrow$  B  
     NOT NULL: IdA, NOT NULL: IdB  
 D(IdD, IdA, IdB, ...) FK: (IdA, IdB) C  
     NOT NULL: IdA, NOT NULL: IdB

3. En un esquema relacional, res no impedeix que un mateix atribut sigui clau forana que referencia més d'una taula. Si aquest esquema és resultat d'un procés de disseny lògic com el que hem vist a l'assignatura, aquesta situació es pot produir en casos poc freqüents, però n'hi ha algun. Dibuixeu un diagrama de classes que pugui donar com a resultat el disseny lògic l'esquema següent:



## 2 Normalization

### 2.1 Problems

1. Given this table:

Orders(OrderID , orderItem1 , orderItem2 , orderItem3 , orderDateTime , CustomerName , CustomerCity )
---

where *OrderID* is an autoIncrement PK, *orderItemN* is the id of the ordered product, *orderDateTime* is the date and time when the order is placed, *CustomerName* is the name of the customer and *CustomerCity* is the city of the costumer.

- Justify whether it is in 1NF, 2NF, 3NF and BCNF or not. Normalize it up to BCNF if necessary.
- Give an alternative schema to hold any number of items in each order.

2. Considereu l'extensió següent per a una relació R(A1, A2, A3, A4, A5)

A1	A2	A3	A4	A5
1	1	1	30	10
2	2	1	30	10
3	1	2	30	15
4	2	2	40	15
5	1	3	20	10
6	2	3	30	10

Suposant que aquesta és l'única extensió possible, digueu per a cada forma normal (1NF, 2NF, 3NF i BCNF) si és satisfeta o no i justifiqueu la resposta:

Si no està en BCNF, indica el resultat de normalitzar-la:

3. (Solució disponible) Considereu la relació R(a1, a2, a3, a4, a5) i dependència funcional  $\{a3\} \rightarrow \{a5\}$  (i que no hi ha cap altra dependència, tret de les que es desprenen de la PK).

- En quina forma normal es troba R? Normalitzeu fins a BCNF.
- En quina forma normal es troba R si eliminem la dependència  $\{a3\} \rightarrow \{a5\}$  i afegim  $\{a1\} \rightarrow \{a5\}$ ? Normalitzeu a BCNF.
- En quina forma normal es troba R si eliminem la dependència  $\{a1\} \rightarrow \{a5\}$  i afegim  $\{a1, a4\} \rightarrow \{a5\}$ ? Normalitzeu a BCNF.

4. Doneu un exemple (o digueu que no és possible) per a cada un dels següents casos, justificant breument la resposta.

- Una relació en *Third Normal Form* (3NF) que no sigui possible normalitzar a *Boyce-Codd Normal Form* (BCNF).
- Una relació en *Third Normal Form* (3NF) que es pugui normalitzar a *Boyce-Codd Normal Form* (BCNF) sense perdre cap dependència. Doneu la relació i com queda normalitzada.



- c) Una relació en *Third Normal Form* (3NF) que es pugui normalitzar a *Boyce-Codd Normal Form* (BCNF), però perdent alguna dependència. Doneu la relació, com queda normalitzada i una dependència perduda.

5. (Solució disponible) The 3NF<sup>1</sup> definition given by Carlo Zaniolo in 1982 states that a relation is in 3NF if and only if for each of its functional dependencies  $X \rightarrow Y$ , at least one of the following conditions holds:

- (a) X contains Y
- (b) X is a superkey (set of attributes that uniquely identifies each tuple of a relation)
- (c) every element of the set difference between Y and X ( $Y \setminus X$ ) is contained in some candidate key (a superkey that can't be reduced to a simpler superkey by removing an attribute).

Sigueu la relació

R(dni, id, assignatura, nota, ciutat)

on id és un identificador dels estudiants propi de la universitat i ciutat és on l'estudiant té residència. De totes les dependències funcionals  $X \rightarrow Y$  que hi ha a la relació, enumereu aquelles que són plenes i que només tenen un atribut a Y. Per a cadascuna, indiqueu amb SÍ o NO si compleixen les condicions de la definició.

Dependència	Compleix (a)	Compleix (b)	Compleix (c)

Està en 3NF? Respon amb SÍ o NO:

Si no ho està, normalitza-la a partir de les dependències que has enumerat i que violen la definició. Indica les claus candidates i foranes.

6. (Solució disponible) Com bé sabeu, l'estructura de les formes normals és tal que cada una implica l'anterior. Demostreu que la de Boyce-Codd implica la tercera seguint els següents passos.

- a) Indiqueu la condició de la BCNF:
- b) Indiqueu la condició de la 3NF:
- c) Argumentació que (a)  $\rightarrow$  (b):

7. És possible deduir a partir de les dependències funcionals  
 $XY \rightarrow ZW$ ,  $ZX \rightarrow WY$ ,  $YW \rightarrow ZY$ ,  $Y \rightarrow W$ ,  $X \rightarrow Z$   
aquest altre conjunt de dependències  
 $X \rightarrow Y$ ,  $Y \rightarrow Z$ ,  $X \rightarrow W$   
utilitzant les regles d'Armstrong?

<sup>1</sup>[https://en.wikipedia.org/wiki/Third\\_normal\\_form](https://en.wikipedia.org/wiki/Third_normal_form)

En cas que sigui possible descriu els passos que vas seguint dient, per a cadascun, les dependències de partida, la regla aplicada i les dependències obtingudes. En cas contrari, justifica la resposta.

8. (*Solució disponible*) Donada la relació “vacuna(id, fabricant, malaltia, pais, cost, durada, tecnica)” i les dependències

<p>fabricant <math>\rightarrow</math> pais tecnica , malaltia <math>\rightarrow</math> durada tecnica , pais <math>\rightarrow</math> cost</p>
--

apliqueu-hi l’algoritme d’anàlisi. Dibuixeu el graf que reflexa totes les execucions possibles, tal i com es fa en els materials de teoria de l’assignatura.

### 3 Data Warehousing and OLAP

#### 3.1 Theoretical questions

1. Explain in which sense each of the four characteristics of a DW in W. Inmon's definition increases the amount of data in the company.
  - 1) **Subject-oriented:**
  - 2) **Integrated:**
  - 3) **Time-variant:**
  - 4) **Non-volatile:**
2. Explain the difference between "Time-variant" and "Nonvolatile" characteristics in the DW definition of W. Inmon. Illustrate it with an example different from that in the materials of the course.
3. Name the three possible meanings of the NULL value in standard SQL.

#### 3.2 Problems

1. Identify factual and dimensional information in the following CSV sample file, and create a conceptual multidimensional schema.

ATTESTATION OF TRAFFIC ACCIDENTS

Month	Month name	Year	Police Region (RP)	Transit Regional Area (ART)	Number
1	Gener	2011	RP PIRINEU OCCIDENTAL	ART PIRINEU OCCIDENTAL	3
1	Gener	2011	RP GIRONA	ART GIRONA	10
1	Gener	2011	RP PONENT	ART PONENT	2
1	Gener	2011	RP CENTRAL	ART CENTRAL	7
1	Gener	2011	RP METROPOLITANA NORD	ART METROPOLITANA NORD	20
1	Gener	2011	RP METROPOLITANA SUD	ART METROPOLITANA SUD	10
1	Gener	2011	RP TERRES DE L'EBRE	ART TERRES DE L'EBRE	4
1	Gener	2011	RP CAMP DE TARRAGONA	ART TARRAGONA	5
2	Febrer	2011	RP PIRINEU OCCIDENTAL	ART PIRINEU OCCIDENTAL	3
2	Febrer	2011	RP GIRONA	ART GIRONA	6
2	Febrer	2011	RP PONENT	ART PONENT	4
2	Febrer	2011	RP CENTRAL	ART CENTRAL	1

2. Identify factual and dimensional information in the following CSV sample file, and create a conceptual multidimensional schema.

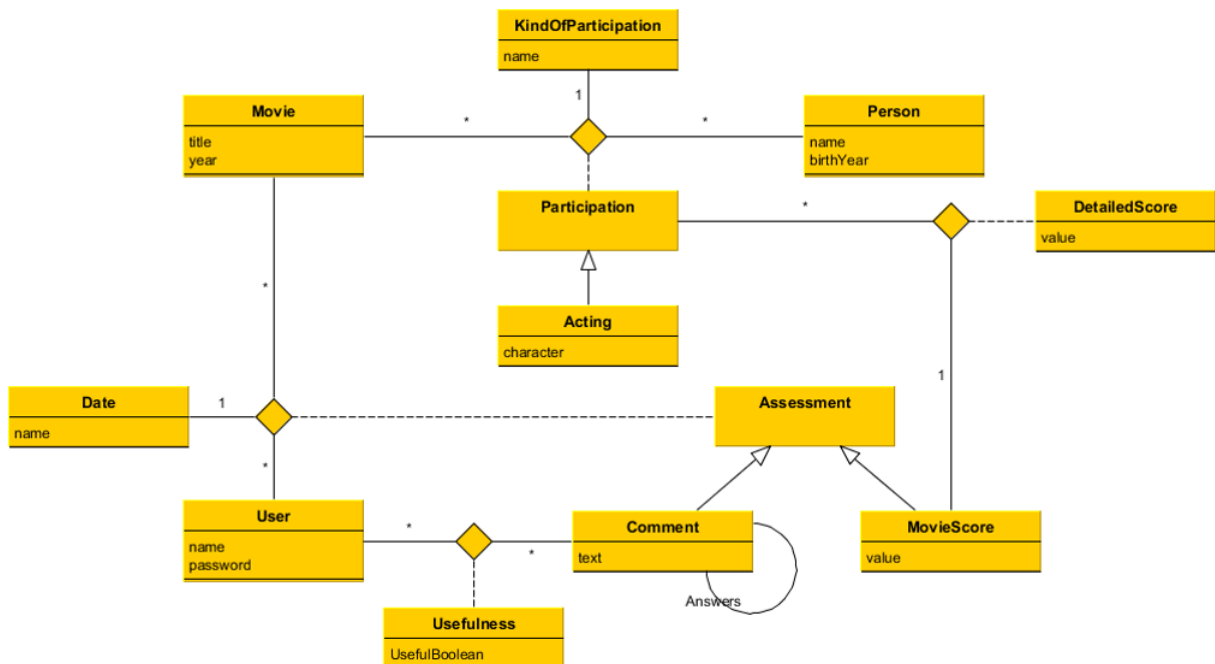
CITIZEN SECURITY

DISTRICTS	RELATED TO PEOPLE	RELATED TO PROPERTY	WEAPON POSSESSION	DRUG POSSESSION	DRUG CONSUMPTION
CENTRO	48	127	10	190	82
ARGANZUELA	61	45	3	10	6
RETIRO	0	19	4	12	0
SALAMANCA	18	58	0	7	0
CHAMARTÍN	12	29	0	16	9

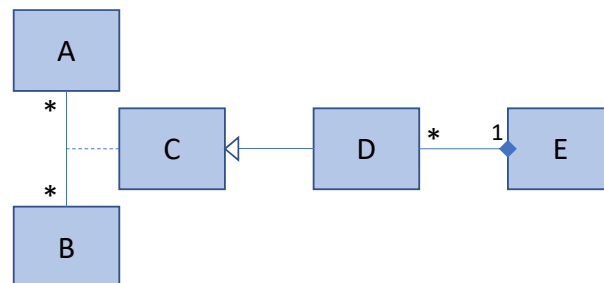
3. (Solució disponible) Create a **multidimensional logical design** (i.e., relational tables and the corresponding constraints) for the following CSV file, assuming that `CountPlayers` indicates the number of players whose data are represented in the row, and `SumX` their statistics. Explicit any assumption you make.

Team	Gender	AgeRange	PlayedEuroleague	Month	SumPoints	SumRebounds	SumFaults	CountPlayers	SumMinutes
Barcelona	M	20-30	T	Oct23	146	57	40	4	300
Barcelona	M	20-30	F	Oct23	30	3	2	1	30
Barcelona	M	-20	F	Oct23	7	1	1	1	5
Lyon	F	-20	F	Oct23	5	7	3	2	17
Avenida	F	+30	T	Sep22	10	2	4	1	20
Joventut	M	-20	F	Dec22	8	2	3	1	9

4. Identify some multidimensional schema (i.e., fact subject of analysis and its corresponding analysis dimensions) in the next UML class diagram.

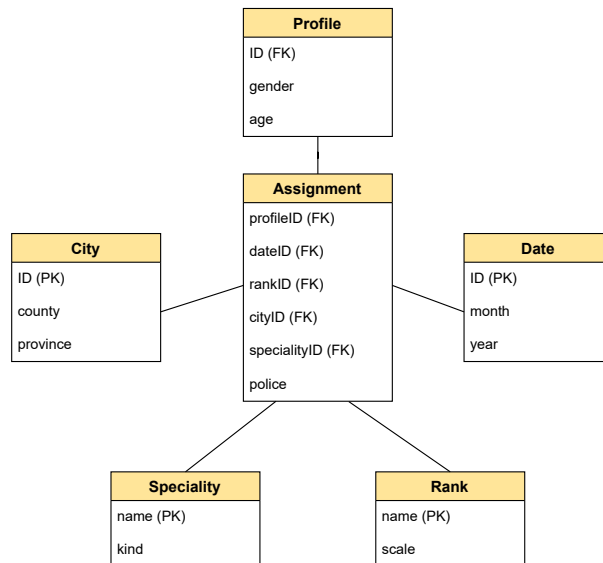


5. Assumeix el següent esquema conceptual, que correspon a una certa base de dades operacional. Aplicant la implementació relacional vista a classe, digues si pot existir un **esquema conceptual multidimensional** la implementació del qual correspongui exactament a una possible implementació relacional d'aquest (es a dir, mateixes taules i mateixes restriccions d'integritat). Si creus que existeix, dona'l; i si creus que no, explica breument el perquè. Explicita les teves assumpcions.



### 3.2.1 Grouping Sets

- Using the new GROUP BY modifiers and the following star schema, create some queries to obtain the result shown in the following exercises:



- (Solució disponible)* For each age range, sum of all the Police with the rank of “Caporal” assigned to each city in the county “Baix Llobregat” with speciality “Support”.

Rank	Age	City	Police
Caporal	19-40	Castelldefels	1
Caporal	19-40	El prat	1
Caporal	19-40	Gava	1
Caporal	19-40	NULL	3
Caporal	41-	Gava	1
Caporal	41-	Viladecans	1
Caporal	41-	NULL	2
Caporal	NULL	NULL	5

- (Solució disponible)* Average female police of any rank in the “Basic” scale assigned every year in the provinces of “Barcelona” and “Lleida”, belonging to specialities “TEDAX” and “Scientific”.

Year	Gender	Province	Speciality	Police
2002	Female	Barcelona	Scientific	6.2
2002	Female	Barcelona	TEDAX	8.6
2002	Female	Barcelona	NULL	14.8
2002	Female	Lleida	Scientific	8.1
2002	Female	Lleida	TEDAX	2.3
2002	Female	Lleida	NULL	10.4
2002	Female	NULL	NULL	25.2
2003	Female	Barcelona	Scientific	150.1
2003	Female	Barcelona	TEDAX	200.5
2003	Female	Barcelona	NULL	350.6
2003	Female	Lleida	Scientific	201.2
2003	Female	Lleida	TEDAX	50.0
2003	Female	Lleida	NULL	251.2
2003	Female	NULL	NULL	601.8
NULL	Female	NULL	NULL	627.0

- c) (*Solució disponible*) Average female police of any rank in the “Basic” scale assigned every year to the province of “Barcelona”, belonging to speciality “TEDAX”.

Year	Gender	Province	Speciality	Police
2002	Female	Barcelona	TEDAX	6.2
2002	Female	Barcelona	NULL	6.2
NULL	Female	Barcelona	NULL	6.2
2002	NULL	Barcelona	TEDAX	6.2
2002	NULL	Barcelona	NULL	6.2
NULL	NULL	Barcelona	NULL	6.2
2002	Female	NULL	TEDAX	6.2
2002	Female	NULL	NULL	6.2
NULL	Female	NULL	NULL	6.2
2002	NULL	NULL	TEDAX	6.2
2002	NULL	NULL	NULL	6.2
NULL	NULL	NULL	NULL	6.2

- d) Average number of police of speciality “Transit” and any profile assigned in 2003 to the cities of “Badalona”, “L’Hospitalet”, “Lleida” and “Ponts”, with ranks of “Caporal” and “Mosso”.

Year	Rank	Province	City	Police
2003	Caporal	Barcelona	Badalona	6.2
2003	Caporal	Barcelona	L’Hospitalet	8.6
2003	Caporal	Barcelona	NULL	14.8
2003	Caporal	Lleida	Lleida	8.1
2003	Caporal	Lleida	Ponts	2.3
2003	Caporal	Lleida	NULL	10.4
2003	Caporal	NULL	NULL	25.2
2003	Mosso	Barcelona	Badalona	150.1
2003	Mosso	Barcelona	L’Hospitalet	200.5
2003	Mosso	Barcelona	NULL	350.6
2003	Mosso	Lleida	Lleida	201.2
2003	Mosso	Lleida	Ponts	50.0
2003	Mosso	Lleida	NULL	251.2
2003	Mosso	NULL	NULL	601.8

e) Average male police assigned every year to a given city with a certain rank.

Year	Rank	Province	City	Police
2003	Caporal	Barcelona	Gava	6.2
2003	Caporal	Barcelona	Barcelona	8.6
2003	Caporal	Barcelona	NULL	14.8
2003	Caporal	Lleida	Lleida	8.1
2003	Caporal	Lleida	Ponts	2.3
2003	Caporal	Lleida	NULL	10.4
2003	Caporal	NULL	NULL	25.2
2003	Mosso	Barcelona	Gava	150.1
2003	Mosso	Barcelona	Barcelona	200.5
2003	Mosso	Barcelona	NULL	350.6
2003	Mosso	Lleida	Lleida	201.2
2003	Mosso	Lleida	Ponts	50.0
2003	Mosso	Lleida	NULL	251.2
2003	Mosso	NULL	NULL	601.8
2003	NULL	Barcelona	NULL	365.4
2003	NULL	Lleida	NULL	261.6
2003	NULL	NULL	NULL	627.0

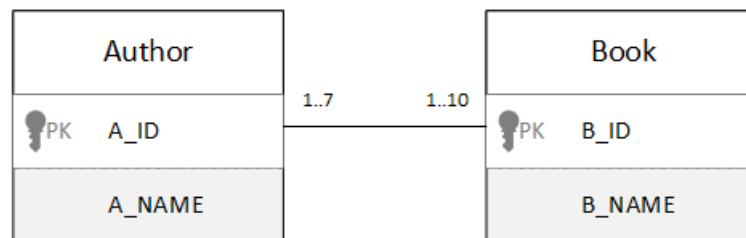
## 4 Not Only SQL

### 4.1 Theoretical questions

1. Briefly explain what “physical independence” is, which is the general position of NOSQL systems on this, and why they take this position.

### 4.2 Problems

1. Consider the following conceptual schema and propose several design alternatives to translate it to a logical representation (e.g., using JSON notation). Then, explain which is the best alternative and why.



2. Considera la següent taula relacional que conté només un identificador i un document JSON en cada fila.

ID	Dades
1	<code>{"nom": "Alejandro", "edat": 17, "telefons": ["123456789", "234567890"], "esports": ["Rugbi", "Ciclisme"]}</code>
2	<code>{"nom": "Eva", "telefons": ["345678901"], "esports": ["Hockey", "Escalada", "Running"]}</code>

- (a) Justifica breument en quina forma normal està, considerant només fins a BCNF.
- (b) Justifica breument ara si està o no en 4NF. Si creus que no ho està, fes el que calgui per a que ho estigui i torna a representar les mateixes dades a sota.



## 5 Views

### 5.1 Theoretical questions

1. Which are the three conditions for a query to be solved from a materialized view without accessing the source tables.
2. Per cadascun dels aspectes, digueu si les vistes (materialitzades o no) **poden** contribuir a millorar-lo. En cas afirmatiu, especifiqueu de quina manera.

Aspecte	S/N	En cas afirmatiu, com?
Concurrència		
Consistència		
Control accés		
Eficiència d'actualitzacions		
Eficiència de consultes		
Redundància		

### 5.2 Problems

1. (*Solució disponible*) Consider a system with two different tables (i.e., Hops and Planes) and the corresponding views (“get\_date” is a function with the obvious meaning):

Hops(from, to, timestamp, plane, mainIncident)

Planes(registry, sits)

- CREATE VIEW Flights(origin, destination, day, aircraft) AS  
SELECT from, to, get\_date(timestamp), plane FROM Hops;
- CREATE VIEW Incidents(aircraft, day, ATAcodes) AS  
SELECT plane, get\_date(timestamp), mainIncident FROM Hops;
- CREATE VIEW Aircrafts(aircraft, capacity) AS  
SELECT registry, sits FROM Planes;

Give the **SQL query** after applying view expansion to the following query:

```
SELECT DISTINCT f.day, f.aircraft
FROM Aircrafts a
JOIN Flights f ON a.aircraft = f.aircraft
JOIN Incidents i ON f.aircraft = i.aircraft AND f.day=i.day
WHERE a.capacity>200 AND i.ATAcodes=21 AND f.origin='BCN';
```

2. Tenemos las siguientes tablas que relacionan distintos superhéroes con la compañía que posee la franquicia y los poderes que tiene cada uno.

```
CREATE TABLE superhero (
  name varchar(20) PRIMARY KEY,
  publisher varchar(10)
);
CREATE TABLE sh_powers (
  sh_name varchar(20) REFERENCES superhero(name),
  power varchar(100),
  PRIMARY KEY (sh_name, power)
);
```

Ahora, creamos la siguiente vista que muestra cuántos superhéroes con cierto superpoder posee una misma compañía:

```
CREATE VIEW powers_by_publisher AS
SELECT s.publisher, p.power, COUNT(*) AS heroes
FROM superhero s, sh_powers p
WHERE s.name = p.sh_name
GROUP BY s.publisher, p.power
ORDER BY p.power;
```

Superhero		
ID	NAME	PUBLISHER
1	superman	DC
2	spiderman	Marvel
3	hulk	Marvel

Sh_power		
ID	SH_NAME	POWER
1	hulk	super-strength
2	spiderman	climbing walls
3	spiderman	high reflexes
4	spiderman	super-strength
5	superman	flying
6	superman	super-strength

ID	PUBLISHER	POWER	HEROES
1	Marvel	climbing walls	1
2	DC	flying	1
3	Marvel	high reflexes	1
4	DC	super-strength	1
5	Marvel	super-strength	2

Queremos hacer las siguientes operaciones sobre la vista. Indica cuáles de ellas serán válidas (es decir, no-ambiguas) y cuáles no (es decir, ambiguas). Para las válidas, indica qué variación produciría cada operación en las tablas originales. En el caso de las inválidas, justifica por qué lo son (da dos modificaciones diferentes en las tablas que darían el resultado requerido en la vista).

(a)

```
INSERT INTO powers_by_publisher(publisher,power,heroes)
VALUES ('Marvel', 'superior intelligence',1);
```

(b)

```
UPDATE powers_by_publisher
SET publisher = 'DC Comics'
WHERE publisher = 'DC';
```

(c)

```
DELETE FROM powers_by_publisher
WHERE power = 'flying';
```

3. (Solució disponible) Justify if we can answer or not the queries using the materialized views.

1) We **have** the following materialized view:

```
CREATE MATERIALIZED VIEW euroSales ENABLE QUERY REWRITE
AS
SELECT d1.city, d2.product, SUM(f.euros) AS sumEuros, COUNT(*) AS salesCounter
FROM sales f, stores d1, products d2
WHERE f.storeId = d1.Id AND f.productId = d2.Id
GROUP BY d1.city, d2.product;
```

We **want** this query:

```
SELECT d1.city , AVG(f.euros) AS avgSales
FROM sales f, stores d1
WHERE f.storeId = d1.Id
GROUP BY d1.city;
```

2) We **have** the following materialized view:

```
CREATE MATERIALIZED VIEW euroSales ENABLE QUERY REWRITE
AS
  SELECT d1.city , d2.productId , SUM(f.euros) AS sumEuros , COUNT(*) AS salesCounter
  FROM sales f, stores d1, products d2
  WHERE f.storeId = d1.Id AND f.productId = d2.Id
  GROUP BY d1.city , d2.productId;
```

And, we **want** this query:

```
SELECT d1.city , p2.productId , f.customerId , SUM(f.euros) AS sales
FROM sales f, stores d1, product p2
WHERE f.storeId = d1.Id AND f.productId = p2.id
GROUP BY d1.city , p2.productId , f.customerId;
```

4. (Solució disponible) Considereu aquesta taula i vistes:

```
CREATE TABLE CentMilResp(
  ref INTEGER PRIMARY KEY,
  pobl INTEGER NOT NULL,
  edat INTEGER NOT NULL,
  cand INTEGER NOT NULL,
  val INTEGER NOT NULL);

CREATE MATERIALIZED VIEW V1 AS
  SELECT cand , pobl , AVG(val) AS av , MAX(val) AS mx,
  COUNT(*) AS n FROM CentMilResp GROUP BY cand , pobl;

CREATE MATERIALIZED VIEW V2 AS
  SELECT pobl , MAX(val) AS mx FROM CentMilResp
  GROUP BY cand , pobl;

CREATE MATERIALIZED VIEW V3 AS
  SELECT cand , edat , AVG(val) AS av , MAX(val) AS mx,
  COUNT(*) AS n FROM CentMilResp GROUP BY cand , edat;
```

Com s'hauria de reescriure cadascuna de les dues consultes següents:

```
C1: SELECT cand , AVG(val) FROM CentMilResp GROUP BY cand;
```

```
C2: SELECT cand , MAX(val) FROM CentMilResp GROUP BY cand;
```

5. Com ja sabeu, una de les aplicacions de les vistes materialitzades és la de materialitzar resultats parcials de consultes multidimensionals gràcies a la capacitat dels SGBD de reescriure consultes expressades sobre les taules en consultes sobre les vistes que tenen un cost menor que les consultes originals. El cas més simple és aquell en què una consulta es reescriu fent servir una única vista materialitzada, però també és possible reescriure una consulta fent servir més d'una. Donada la taula CentMilResp(ref, pobl, edat, cand, val) i les consultes següents, reescriuiu, si és possible, cada consulta en funció de les altres (una o més) o indiqueu "no és possible" quan escaigui.

```

(C1) SELECT cand, edat, MIN(val) AS mn FROM CentMilResp WHERE edat > 50 GROUP BY cand,
    edat;

(C2) SELECT cand, edat, COUNT(*) AS n, MAX(val) AS mx FROM CentMilResp WHERE edat > 40
    GROUP BY cand, edat;

(C3) SELECT cand, edat, MAX(val) AS mx, MIN(val) AS mn FROM CentMilResp WHERE edat > 50
    GROUP BY cand, edat;

(C4) SELECT cand, edat, COUNT(*) AS n, MAX(val) AS mx, MIN(val) AS mn FROM CentMilResp
    WHERE edat <= 60 GROUP BY cand, edat;

(C5) SELECT edat, COUNT(*) AS n FROM CentMilResp GROUP BY edat;

```

6. (Solució disponible) Tenim una taula Comandes(client, data, hora, origen, producte, quantitat, import) que ocupa 1000 blocs. Definim una vista no materialitzada Imports(client, data, import). El sistema processa tant insercions de noves comandes com consultes que calculen punts de descompte dels clients. Aquests punts depenen dels imports de les comandes del client, excepte les dues darreres, per si hi ha devolucions, i la consulta es defineix sobre la vista. Si decidim fer que la vista passi a ser materialitzada, amb quins paràmetres la definiries? Encercla els que creguis convenients:

REFRESH:      Never      Fast      Complete      Force

                 On Demand      On Commit      Next <date>

Què podem esperar que passarà? Per cada casella sota el títol VARIACIÓ, posa "0" si creus que no canviaria, "+" si creus que augmentaria i "-" si creus que baixaria. Acompanya "+" i "-" amb un número si ets capaç de fer alguna estimació (els costos els expressem com accessos a disc i l'espai en blocs, suposant que tots els atributs ocupen igual i que la informació de control ocupa com un atribut). Dóna un raonament de la teva resposta sota MOTIU.

	Variació	Motiu
Cost inserció		
Cost consulta		
Espai		
Throughput (Transaccions/temps)		

Torna a contestar suposant que els punts depenen de totes les comandes, sense excloure les últimes:

REFRESH:      Never      Fast      Complete      Force

                 On Demand      On Commit      Next <date>

	Variació	Motiu
Cost inserció		
Cost consulta		
Espai		
Throughput (Transaccions/temps)		

7. Let's suppose we have a table T(ID, A, B, C) with 1,000 blocks and 100 records per block. We have a materialized view V(ID, A, B, C) with 100 blocks and also 100 records per block that contains a selection over T by a predicate involving A and B. Assuming that neither the table nor the materialized view have any physical order (i.e., they are not clustered), for each of the following **operations**, would it be better a **complete** or an **incremental update** of the view? Briefly explain your answer.

Assumptions:

- Statistical independence of attributes
- Uniform distribution of all values in the different data blocks
- No benefit from sequential scan (i.e., no pre-fetching)
- No benefit from consecutive random access of the same block (i.e., no caching)

a) We receive `UPDATE T SET C=C+1 WHERE Id%10=0;`

b) We receive `UPDATE T SET C=C+1 WHERE Id%1000=0;`

8. (Solució disponible) Supposeu que tenim una taula CentMilResp(ref, pobl, edat, cand, val), que costa 3 minuts llegir-la sencera. La freqüència d'operacions sobre aquesta taula és:

- 20%: (C1) `SELECT cand, AVG(val) FROM CentMilResp GROUP BY cand;`
- 15%: (C2) `SELECT cand, edat, COUNT(*), AVG(val), MAX(val), FROM CentMilResp GROUP BY cand, edat;`
- 20%: (C3) `SELECT edat, pobl, MAX(val), AVG(val), FROM CentMilResp GROUP BY edat, pobl;`
- 15%: (I) `INSERT INTO CentMilResp...;`
- 15%: (U) `UPDATE CentMilResp SET val = val * 0.9 WHERE ...;`
- 15%: (D) `DELETE FROM CentMilResp WHERE ...;`

Suposant que no tenim problemes d'espai, que podem llegir qualsevol de les vistes que es materialitzin amb un cost zero i que el cost de les operacions d'actualització també és zero, quines de les tres consultes guardaríeu en forma de vista materialitzada si el SGBD només oferís la possibilitat d'actualitzar cada vista "ON COMMIT" i "FAST" amb un cost de 2 minuts? Per decidir-ho, ompliu la taula següent amb els cost en minuts de cada operació en funció de les vistes materialitzades. Poseu el cost mitjà a la darrera columna.

Vistes materialitzades/Consultes	C1	C2	C3	I	U	D	Mitjana
Cap							
C1							
C2							
C3							
C1 i C2							
C1 i C3							
C2 i C3							
C1, C2 i C3							

Vistes a materialitzar:

9. Suppose that we have a table  $T(A,B,C)$  and the list of the frequencies of the operations on this table (a.k.a. workload). For the stated frequencies, which queries would you store as a materialized view if the DBMS only offered the possibility of updating each view "ON COMMIT" and "FAST"? Numerically justify your answer.

- a)
- 15%: `SELECT A, SUM(C) FROM T GROUP BY A;`
  - 9%: `SELECT B, SUM(C) FROM T GROUP BY B;`
  - 36%: `SELECT SUM(C) FROM T;`
  - 40%: `INSERT`, `UPDATE` (of all the attributes) and `DELETE` over T

Do the calculations assuming that:

- It takes 4 minutes to read the table.
- Updating each view "ON COMMIT" and "FAST" has a cost of 2 minutes.
- You don't have problems of space.
- You can read any of the views you materialize with cost zero.
- The cost of modifying the table is also zero.
- The **rewriting mechanism is very limited** and it just uses a materialized view if the query coincides exactly with the view (that is, for example, the view corresponding to the first query cannot be used to answer the third query).

- b)
- 15%: `SELECT A, SUM(C) FROM T GROUP BY A;`
  - 9%: `SELECT B, SUM(C) FROM T GROUP BY B;`
  - 36%: `SELECT SUM(C) FROM T;`
  - 10%: `INSERT` into T
  - 10%: `UPDATE` of A
  - 10%: `UPDATE` of B
  - 10%: `DELETE` from T

Do the calculations assuming that:

- It takes 4 minutes to read the table.
- Updating each view "ON COMMIT" and "FAST" has a cost of 2 minutes.
- You don't have problems of space.
- You can read any of the views you materialize with cost zero.

- The cost of modifying the table is also zero.
- The **rewriting mechanism is good enough** to use a materialized view even if it doesn't exactly coincide with the query (that is, for example, the view corresponding to the first query can be used to answer the third).

- c)
- 20%: `SELECT A, SUM(C) FROM T GROUP BY A;`
  - 20%: `SELECT B, SUM(C) FROM T GROUP BY B;`
  - 20%: `SELECT SUM(C) FROM T;`
  - 10%: `INSERT` into T
  - 15%: `UPDATE` of A
  - 5%: `UPDATE` of B
  - 10%: `UPDATE` of C

Do the calculations assuming that:

- You don't have space problems.
- You can read either the table or any of the views you materialize with a cost equal to the number of attributes it has.
- The cost of modifying either the table or a materialized view is always one.
- The **rewriting mechanism is good enough** to use a materialized view even if the query does not exactly match the view.

10. (*Solució disponible*) Give an example (as simple as possible) of database and associated workload where a greedy algorithm picking first the materialized view generating the highest improvement of performance does **not** generate the optimum set of views to materialize in the end. Briefly explain why this happens in this case.

## 6 Physical Design (Access structures)

### 6.1 Theoretical questions

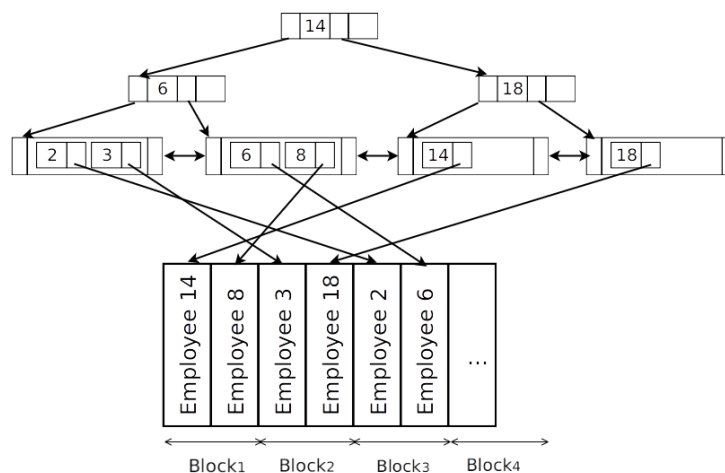
1. We estimate the cost of a table scan in a clustered index like  $\lceil 1.5B \rceil \cdot D$ . However, we estimate the cost of searching one tuple without index like  $0.5B \cdot D$ . Briefly explain why we round it up in the first case and not in the second.
2. Donada una consulta per igualtat d'un atribut, fora bona idea posar un índex Hash per a una taula amb 10 milions de tuples i 100 valors diferents a l'atribut indexat? Explica per què.

### 6.2 Problems

1. Let's suppose you have a dimension table with
  - a. 1,024,000 tuples
  - b. Keys of 10 bytes
  - c. Record Identifiers (i.e., RIDs) of 10 bytes
  - d. A block size of 1,000 bytes
  - e. A B-tree index loaded exactly at 80% for all nodes

Estimate the size needed to store the index following the assumptions done in the course.

2. Given the figure below, answer the following questions in order to analyze the difference between sequential and random access.



- a) If we need to **read the whole table**. How many blocks do we read if using a table scan (ignoring the index)? How many blocks do we read using the B+ index to read the whole table?
- b) If we need to retrieve the **employee number 6**. How many blocks do we read if using a table scan? How many blocks do we read if using the B+ index to find it?



3. Com bé sabeu, quan hem estudiat els índexs hem suposat que a les entrades s'hi guardaven parelles (clau, adreça). Pensem com afectaria els costos d'un B+ que cada entrada de l'índex contingues una fila sencera de manera que ja no distingiríem entre índex i taula sinó que les dues coses s'integren en una. Suposant que la mida d'un bloc són 12 KB, que les files ocupen 1 KB i que una entrada (clau, adreça) ocupa 0.05 KB, ompliu les taules següents.

ENTRADES (CLAU, ADREÇA)	Ordre arbre (d)	Cost una tupla	Cost diverses tuples (FS 0.1)	Cost diverses tuples (FS 0.9)
$ T  = 10$				
$ T  = 1000$				
$ T  = 10000$				

ENTRADES FILA SENCERA	Ordre arbre (d)	Cost una tupla	Cost diverses tuples (FS 0.1)	Cost diverses tuples (FS 0.9)
$ T  = 10$				
$ T  = 1000$				
$ T  = 10000$				

4. Siguin les dades següents:

nau(matr, nom, prop, eslora, pes)

$d = 150$ ,<sup>2</sup>  $B = 1000$ ,  $M = 10$ ,  $|nau| = 2000$

i les consultes

(Q1) SELECT nom, prop FROM nau;

(Q2) SELECT DISTINCT nom, prop FROM nau;

Indiqueu el cost de resoldre la consulta utilitzant l'estructura donada en cada cas, sense considerar el cost d'escriure el resultat final:

	Hash(nom)	B+(nom, prop)	No índex
Q1			
Q2			

<sup>2</sup>Feu servir el mateix valor de  $d$  com a referència de capacitat tant per al B+ de dos atributs com per al *hash* d'un atribut.

## 7 Query Optimization

### 7.1 Theoretical questions

1. Name the two heuristic rules of syntactic optimization, and briefly explain the rationale behind them (i.e., why they, in general, work).
2. Per a cada parell d'operacions en un arbre sintàctic, indiqueu si es poden intercanviar:
  - a) Projectió i unió:
  - b) Selecció i unió:
  - c) Projectió i diferència:
  - d) Selecció i diferència:

Per als casos en què no sigui possible, poseu un exemple que ho demostrï:

3. Give an example of SQL query whose syntactic optimization would require the merge of two projections.
4. Name the two join algorithms that allow pipelining and briefly explain what is the benefit of using it, and where it comes from.

### 7.2 Problems

1. Given the following tables, briefly explain what the semantic optimizer could do with the corresponding queries.

Tables:

```
CREATE TABLE students (  
  id CHAR(8) PRIMARY KEY,  
  mark FLOAT CHECK (mark>3),  
  school CHAR(8) REFERENCES schools(id)  
);  
  
CREATE TABLE schools (  
  id CHAR(8) PRIMARY KEY,  
  name CHAR(50)  
);
```

a) Query:

```
SELECT t.id  
FROM students t, schools c  
WHERE t.school=c.id;
```

b) Query:

```
SELECT t.id, c.name  
FROM students t, schools c  
WHERE t.school=c.id  
      AND t.school='FIB';
```

2. Given the following tables, briefly explain what the semantic optimizer could do with the corresponding queries:

(a) Query 1

```
CREATE TABLE employees (id CHAR(8) PRIMARY KEY, salary FLOAT CHECK (salary > 10) );
```

```
SELECT *  
FROM employees  
WHERE salary < 8;
```

(b) Query 2

```
CREATE TABLE employees (id CHAR(8) PRIMARY KEY, salary FLOAT CHECK (salary > 10) );
```

```
SELECT *  
FROM employees  
WHERE salary > 8;
```

(c) Query 3

```
CREATE TABLE employees (id CHAR(8) PRIMARY KEY, salary FLOAT CHECK (salary > 10) );
```

```
SELECT *  
FROM employees  
WHERE (salary = 11 AND id > '8')  
OR (salary = 12 AND id > '8');
```

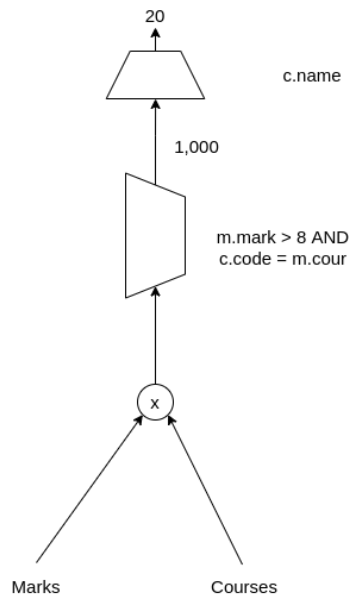
(d) Query 4

```
CREATE TABLE employees (id CHAR(8) PRIMARY KEY, salary FLOAT, dpt CHAR(8) REFERENCES  
departments(id));  
CREATE TABLE departments (id CHAR(8) PRIMARY KEY, name CHAR(50));
```

```
SELECT e.id  
FROM employees e  
WHERE e.dpt NOT IN (SELECT id FROM departments);
```

3. Optimize the following syntactic trees:

- a) (*Solució disponible*) Tree 1 (knowing that the query returns 20 tuples and 1,000 tuples result from the selection, indicate the tuples resulting from the cartesian product and those resulting from any operation after the syntactic optimization)



Courses(code, name, ...)  
 Students(id, ...)  
 Marks(cour, stu, mark)

- {stu} FK to Students
- {cour} FK to Courses

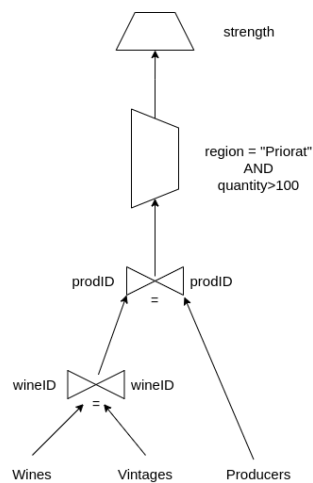
Table cardinalities:

- $|Courses| = 200$
- $|Students| = 3000$
- $|Marks| = 15000$

Query:

```
SELECT DISTINCT c.name
FROM courses c, marks m
WHERE c.code = m.cour
      AND m.mark > 8;
```

b) (Solució disponible) Tree 2



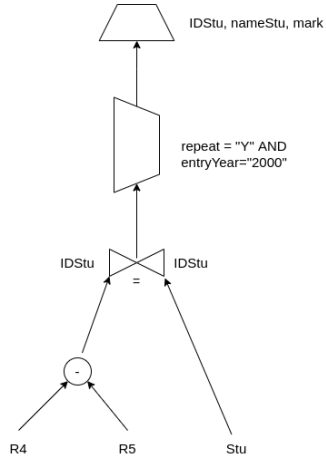
Wines(wineID, wineName, strength)  
 Producers(prodID, prodName, region)  
 Vintages(wineID, prodID, quantity)

- {wineID} FK to Wines
- {prodID} FK to Producers

Query:

```
SELECT DISTINCT w.strength
FROM wines w, producers p, vintages v
WHERE v.wineID = w.wineID
      AND p.prodID = v.prodID
      AND p.region = "Priorat"
      AND v.quantity > 100;
```

c) (Solució disponible) Tree 3



Subjects(IDSub, nameSub, credits)  
 Students(IDStu, nameStu, degree, entryYear)  
 Registration2005(IDStu, IDSub, mark, repeat)

- {IDStu} FK to Students
- {IDSub} FK to Subjects

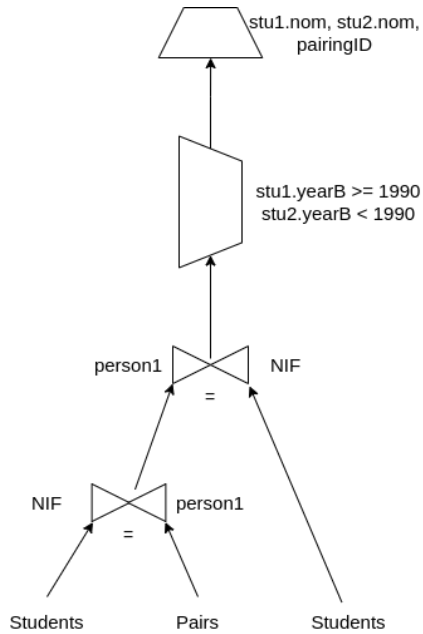
Registration2004(IDStu, IDSub, mark, repeat)

- {IDStu} FK to Students
- {IDSub} FK to Subjects

Query:

```
SELECT s.IDStu, s.nameStu, r.mark
FROM Students s,
( SELECT * FROM Registration2004
  EXCEPT
  SELECT * FROM Registration2005 ) r
WHERE r.repeat='Y' AND s.entryYear=2000 AND s.IDStu=r.IDStu;
```

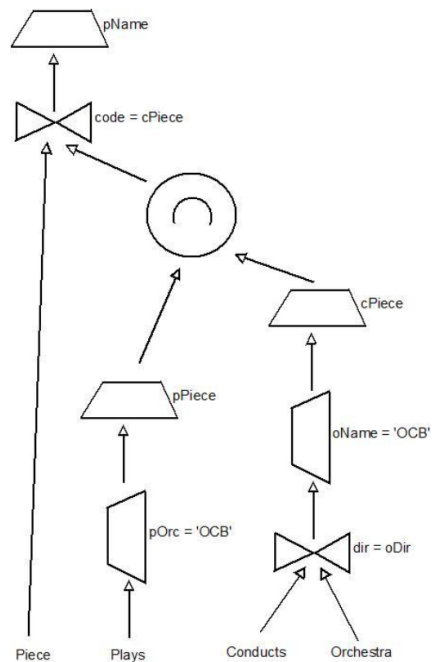
d) Tree 4



Students(NIF, name, surname, yearB, phone, email)  
 Pairs(pairingID, name, person1, person2)

- {person1} foreign key to Students
- {person2} foreign key to Students

e) Tree 5



Piece (code, pName, year)  
 Orchestra(oName, city, nMusicians, oDir)  
 Conducts(dir, cPiece)  
 Plays(pOrc, pPiece)

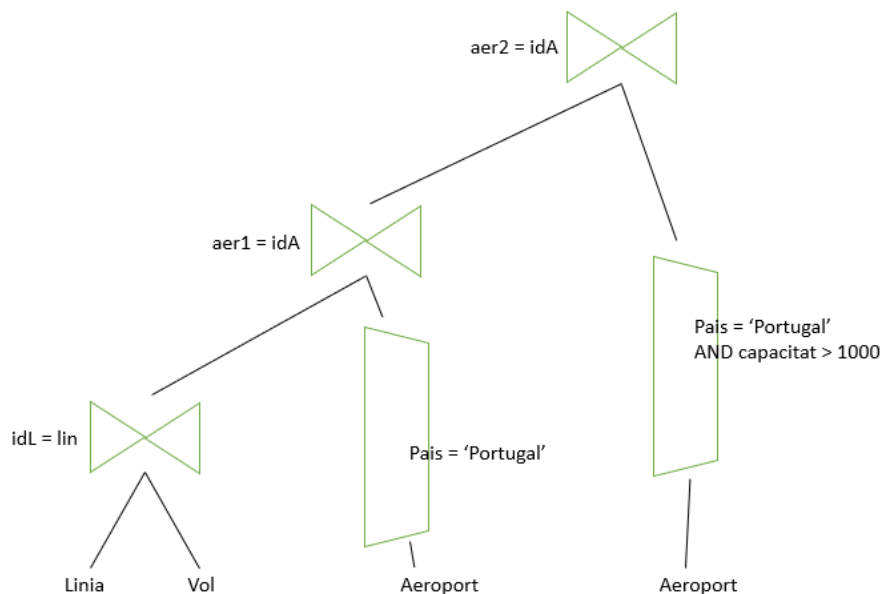
- {cPiece} foreign key to Piece
- {pPiece} foreign key to Piece

4. Considereu aquests objectes:

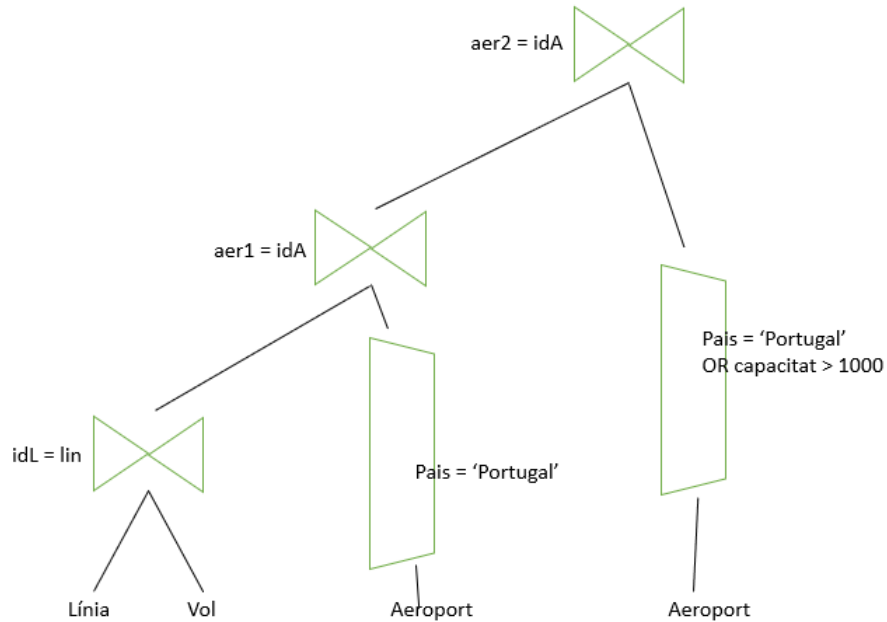
```
CREATE TABLE Linia(idL, pais);
CREATE TABLE Aeroport(idA, pais, capacitat);
CREATE TABLE Vol(idV, aer1, aer2, lin);

CREATE VIEW LinPort (idL, Lpais, idV, aer1, aer2, lin, ida1, APais1, capacitat1, ida2,
APais2, capacitat2) AS
SELECT *
FROM Linia L
JOIN Vol V ON (idL = lin)
JOIN Aeroport A1 ON (aer1 = A1.idA)
JOIN Aeroport A2 on (aer2 = A2.idA)
WHERE A1.pais = 'Portugal' AND A2.pais = 'Portugal';
```

a) Doneu una consulta SQL sobre la vista que doni el següent arbre:



b) Doneu una consulta SQL sobre la vista que doni el següent arbre:



5. (Solució disponible) Considereu la taula

`T(citId, name, address, status, children, birthDate, debt, jobKind)`

i suposeu que hi ha un índex B+ definit per a cada atribut de la taula. Indiqueu com s'efectuàrien les seleccions següents fent servir l'estil del material de classe per indicar les condicions resoltes amb índex i com es combinen els resultats fent unions i interseccions de llistes d'adreces.

a)  $(\text{status} = \text{'married1'} \text{ OR } \text{birthDate} > 10000) \text{ AND NOT } (\text{name} = \text{'Mariano'} \text{ OR } \text{debt} > 1000)$

b)  $(\text{jobKind} = \text{'Clerk'} \text{ AND } \text{birthDate} < 25000) \text{ OR NOT } (\text{status} = \text{'divorced3'} \text{ OR } \text{name} < \text{'V'})$

6. Considereu la taula

`Vehicles(matricula, marca, model, any)`

i suposeu que hi ha un índex B+ definit per a cada atribut de la taula. Indiqueu com s'efectuàrien les seleccions següents fent servir l'estil del material de classe per indicar les condicions resoltes amb índex i com es combinen els resultats fent unions i interseccions de llistes d'adreces.

a)  $\text{marca} = \text{'ACME'} \text{ AND } \text{any} > 1990 \text{ AND } \text{matricula} \neq \text{'UI8723'} \text{ AND NOT } (\text{model} = \text{'DFA'} \text{ OR } \text{any} < 2000)$

b)  $(\text{marca} = \text{'ACME'} \text{ OR } \text{any} < 1990) \text{ AND } \text{matricula} = \text{'UI8723'} \text{ AND } (\text{model} = \text{'DFA'} \text{ OR } \text{any} < 2000)$

7. (Solució disponible) Considereu la taula

`Joguina(referencia, marca, model, descripcio)`

Suposeu que tenim un índex hash per marca i un índex cluster per referència.

Suposeu també que els RIDs no ocupen espai a memòria (de manera que en podem tenir tants com volguem i les operacions que fem amb ells no tenen cost).

Tenint en compte les estadístiques següents, indiqueu quants blocs d'índex i de dades es llegiran per resoldre la consulta segons cada estratègia:

- $D=1$ ;  $C=0$
- $|Joguina| = 50000$ ;  $R_{Joguina} = 5$ ;  $B_{Joguina} = 10000$
- L'ordre de l'índex és  $d = 60$  SELECT \* FROM Joguina
- $\min(\text{referencia}) = 1$ ;  $\max(\text{referencia}) = 50000$  WHERE referencia < 10000 AND
- $\text{ndist}(\text{marca}) = 1000$  marca IN ('M1', 'M2', 'M3', 'M4', 'M5')

Estratègia	Usar només i. hash	Usar només i. cluster	Usar-los tots dos	No usar cap índex
Blocs índex				
Blocs dades				

8. Let us consider a table R with three attributes, A, B and C. Suppose a multi-attribute index on B, C and A (in this order). Which of the following conditions could be answered with this index?<sup>3</sup> (Answer Y/N)

- |  |  |  |
|--|--|--|
| a) $A > 1$                                       | e) $B < 3$                                       | i) $C = 5$                                       |
| b) $A = 5 \text{ AND } B > 1$                    | f) $A = 4 \text{ AND } C < 4$                    | j) $A < 5 \text{ AND } B = 4$                    |
| c) $B = 3 \text{ AND } A > 2 \text{ AND } C = 4$ | g) $A = 5 \text{ AND } C > 1 \text{ AND } B = 1$ | k) $A = 4 \text{ AND } B < 4 \text{ AND } C > 4$ |
| d) $B = 4 \text{ AND } A = 3$                    | h) $B = 2 \text{ AND } C = 6$                    | l) $A = 7 \text{ AND } C = 3$                    |

9. (Solució disponible) Considereu la taula Torrons(..., tipus, qual, pes, ...), amb les estadístiques i consulta següents:

- $|Torrons| = 10000$ ,  $B_{Torrons} = 5000$
- Hi ha un B+ per pes, un índex cluster per tipus i un hash per qual (tots amb ordre 75).
- $\text{Ndist}(\text{tipus}) = 100$ ,  $\min(\text{qual}) = 0$ ,  $\max(\text{qual}) = 10$ ,  $\min(\text{pes}) = 0$ ,  $\max(\text{pes}) = 1000$

```
SELECT * FROM Torrons WHERE (tipus = 'Crema' OR qual > 4.9) AND pes >= 750
```

Quants accessos es faran a cada índex i a la taula?

	N. accessos
TAULA	
B+ pes	
Cluster tipus	
Hash qual	

10. Let us consider table T, which contains two attributes A and B. Suppose a hash index on A and a B+ on B. Suppose now that RIDs do not occupy any space in memory (so you can have as many as you want and you can operate them for free). Given the following database configuration, give the query cost in each scenario:

- $D=1$ ;  $C=0$
- $|T| = 100.000$ ;  $R_T = 10$ ;  $B_T = 10.000$

<sup>3</sup>Remember that an index is only considered useful if it exactly retrieves those addresses fulfilling a given condition. Thus, do not consider them to be useful by scanning (partially or fully) the index structure and skipping some RIDs.



- Tree order is 75
- ndist(A)=20.000; ndist(B)=100.000
- min(A)=1; min(B)=1
- max(A)=20; max(B)=100

```
SELECT * FROM T WHERE A IN (1,2,3,4,5,6,7) AND B>95;
```

Strategy	Cost
Use only the hash on A	
Use only the B+ on B	
Use both indexes	

### 7.2.1 Joins

1. Considereu les taules i la consulta següent:

```
CREATE TABLE A(id INT, x INT, ...);

CREATE TABLE B(id, ...);

SELECT * FROM A, B
WHERE A.id = B.id AND A.x <= 10
```

Les dues taules tenen un índex cluster per id i A un B+ sobre x i totes dues taules tenen moltes tuples (de l'ordre de 1000000) i ocupen molts blocs (de l'ordre de 500000). Hi ha unes 90000 tuples de A per cada valor de x entre 1 i 10 i la resta de tuples tenen valors que oscil·len entre 11 i 1000.

Ompliu la taula amb camins d'accés per a la selecció i algorismes per a la join.

	Selecció	Join
Millor algorisme assumint distribució uniforme de valors		
Millor algorisme amb la distribució real de valors		

2. Given the following tables, answer the following exercises:

#### Connections

URL	access_time	user	landing_time	ip	Location
edu.upc.www	01/01/2017:11:45:30	Oscar	45"	127.34.23.1	Barcelona
www/edu.essi.upc	01/01/2017:11:46:15	Oscar	10"	127.34.23.1	Barcelona
www/edu.upc	01/01/2017:11:46:16	Anna	67"	123.44.9.6	Lleida
www/edu.essi.upc	01/01/2017:11:46:18	Ona	240"	126.78.65.3	Girona
www/edu.essi.upc.dtim	01/01/2017:11:46:25	Oscar	560"	127.34.23.1	Barcelona
www/edu.essi.upc	01/01/2017:11:46:33	Marc	24"	126.55.34.2	Barcelona
www/edu.upc	01/01/2017:11:45:33	Nuria	90"	123.23.1.2	Girona
www/edu.essi.upc	01/01/2017:11:47:24	Anna	100"	123.44.9.6	Lleida

#### Users

user_id	account_created	age
Oscar	20/12/2015:16:30:01	34
Anna	03/03/2014:21:55:23	33
Ona	31/12/2016:09:00:18	28
Marc	12/05/2016:15:20:01	20
Nuria	26/11/2016:11:32:33	21

Note: The *user* attribute in the *Connections* table is a FK to *user\_id* in the *Users* table.

1) Create the syntactic tree for the following SQL query:

```
SELECT DISTINCT users.user_id , users.age
FROM connections , users
WHERE connections.user = users.user_id
      AND connections.location = 'Barcelona'
ORDER BY users.age;
```

2) Consider now the following index:

- Hash on connections (location)
- B+ on users (user\_id)

And knowing the following statistics extracted from the database catalog:

- There are four possible location values in the database (Barcelona, Lleida, Girona and Tarragona)
- There are 1,000 users in the users table
- Only 200 of the users appear in the connections table

Note: assume uniform distribution of values and *Row Nested Loop* is the only join algorithm available.

- a) How many alternative process trees can you think of? Clearly identify them.
  - b) Estimate the selectivity factor.
  - c) With the information from the two previous questions, what is the best process tree in terms of I/O operations?
3. Let's suppose that the only join algorithm in our DBMS is Row Nested Loops and we have a B-tree index of order 3 on S over attribute B (which is occupied at 2/3, as usual). Give the approximate I/O cost of "SELECT S.C FROM R,S WHERE R.A=S.B AND R.D=123", given the following statistics:

- $|R| = 1000, B_R = 10, R_R = 100$
- $|S| = 100, B_S = 10, R_S = 10$
- $SF_{Selection} = 0.5, SF_{Join} = 0.01$

Justify your answer numerically, and briefly explain it.

4. (Solució disponible) Potser recordeu que el cost de l'algoritme Row Nested Loops per a B+ i índex cluster és el mateix si no cal obtenir les tuples de la taula del bucle intern (semi-join) però, en canvi, és diferent quan sí que cal obtenir-les.
- a) Quin és el motiu de la diferència?
  - b) Imaginem que la taula  $T_e$  del bucle extern té 1000 tuples i ocupa 100 blocs, que la taula  $T_i$  del bucle intern té 10000 tuples i ocupa també 100 blocs i que fem l'equi-join amb la clau primària de  $T_e$  i una clau forana de  $T_i$ . Quina serà la diferència d'accessos a disc segons quina de les dues estructures existeixi?

5. (Solució disponible) **Estimate the cost** of the query below (including the writing of the final output, but not any intermediate results) under the following assumptions:<sup>4</sup>

---

<sup>4</sup>Being  $Dist(attribute)$  the number of different values of the attribute.

- Fact table schema: Sales(TimeID, CustomerID, LocationID, income, expenses)
- Execution strategy: Pipelining
- Memory available: Seven pages (two devoted exclusively to each join operator and one for the final output)
- Join algorithm available: Block Nested Loops
- No indexes, no caching, and no parallelism available
- All attributes are statistically independent and values uniformly distributed
- Any CPU or memory cost can be ignored, only disk access is relevant
- $|Sales| = 10^7$ ;  $B_{Sales} = 10^5$ ;  $R_{Sales} = 100$
- $|Customer| = 10^4$ ;  $B_{Customer} = 200$ ;  $R_{Customer} = 50$ ;  $Dist(Gender) = 2$
- $|Time| = 10^3$ ;  $B_{Time} = 10$ ;  $R_{Time} = 100$ ;  $Dist(Year) = 4$
- $|Location| = 10^3$ ;  $B_{Location} = 10$ ;  $R_{Location} = 100$ ;  $Dist(City) = 10$
- $Cost_{I/O} = 1$

```
SELECT s.TimeID, s.CustomerID, s.LocationID, s.income, s.expenses
FROM Sales s, Time t, Customer c, Location l
WHERE s.TimeID = t.ID AND s.CustomerID = c.ID AND s.LocationID = l.ID
AND t.Year = 2023 AND l.City = "Barcelona" AND c.Gender = "Female";
```

6. L'algoritme de *Two-pass Hash Join* funciona, com sabeu, en dos passos: en un primer pas reorganitza les files de les dues taules fent servir una mateixa funció de *hash* que assigna cada fila a una de les  $p$  parts en què es parteix cada taula i, en un segon pas, fa la *join* part a part amb l'algoritme de *One-pass Hash Join*. Si suposem que tenim  $M + 2$  pàgines de memòria, per garantir que el primer pas es pot fer amb una sola passada per taula cal que  $M + 2 \geq p + 1$  de manera que tenim un buffer d'entrada i  $p$  buffers de sortida. D'altra banda, per garantir que en el *One-pass Hash Join* del segon pas l'índex hash de la taula petita càpiga a memòria cal que  $\frac{B_{SMALL}}{p} \leq M$ , atès que cada part de la taula més petita ocuparà  $\frac{B_{SMALL}}{p}$  blocs. És satisfent simultàniament aquestes condicions que el cost de l'algoritme es correspon a llegir dos cops i escriure un cop cada taula i la conjunció d'aquestes condicions dóna lloc a la que ja coneixeu,  $B_{SMALL} \leq M^2 + M$ . Cas que aquesta condició no es compleixi, ens podem plantejar dues solucions:

- Triar  $p$  tan petit com es pugui, garantint que el primer pas es fa amb una sola passada. Si la partició de la taula petita no cap a memòria, podeu assumir un cost de *One-pass Hash Join* igual a BNL (es a dir,  $B_R + B_S \cdot \lceil B_R/M \rceil$ ).
- Triar  $p$  tan gran com calgui per a garantir que l'índex hash de la taula petita càpiga a memòria en segon pas.

Ompliu la taula següent amb els valors de  $p$  que s'haurien de triar i el cost de cada pas en funció de  $B_{BIG}$ ,  $B_{SMALL}$  i  $M$ . A la darrera columna indiqueu amb  $S$ ,  $N$  o  $?$  si el cost total és sempre millor que el de fer directament la *join* amb BNL de les taules originals, sempre pitjor o no es pot assegurar.

	p	Cost pas 1	Cost pas 2	Millora BNL?
A				
B				

### 7.2.2 Bitmap indexes

- The database of the U.S. government has 194 countries. For each country it keeps the continent (i.e., 5 different values). To store the information of the continents they use an uncompressed bitmap. So we have an array of 1's and 0's (bitmap size:  $194 \times 5 = 970$  bits). Today the lunar base in the United States has made a coup and was declared the Democratic Republic of the Moon. The Foreign Office has updated the database to add the new country and new continent 'Moon'. Give the number of bits that the bitmap grows, and briefly explain the calculations.

2. Considerem ara la possibilitat d'explotar els bitmaps de formes alternatives. Suposem que A és l'atribut indexat i que no admet valors nuls.

a) Penseu que és possible resoldre amb el bitmap condicions de l'estil " $A <> v$ ", on  $v$  és un valor constant? En cas afirmatiu, expliqueu com es faria.

b) Per resoldre una condició com ara  $A \text{ IN } (v_1, \dots, v_n)$ , es converteix en  $A = v_1 \text{ OR } \dots \text{ OR } A = v_n$  que es resol accedint al bitmap amb els  $n$  valors i fent operacions de bits. Es pot fer d'alguna altra manera que accedeixi a menys blocs de disc en alguna circumstància?

3. Donada la consulta següent:

```
SELECT * FROM books
WHERE tematica='INFORMATICA' AND pagines IN [12,1300]
```

quan seria millor usar bitmap i quan B+? Responeu en funció del nombre de llibres, temàtiques i pàgines.

**Millor bitmap:**

**Millor B+:**

4. (*Solució disponible*) Consider an RDBMS with a fact table `sales` with a boolean dimension `local` indicating whether the corresponding sale was "local" (true) or "global" (false). The table contains 10,000,000 rows of 80 bytes each in the average (including both data and metadata). The dimension has a **bitmap index** over its `local` attribute. Assuming a disk block of 8K,<sup>5</sup> give the **cost** of the following query and explicit any assumption you make:

```
SELECT SUM(income)
FROM sales
WHERE local;
```

5. Supposeu que teniu un índex tipus *bitmap* per cadascun dels atributs A i B de la taula T. Aquesta taula ocupa 200 blocs i cada bloc conté 40 tuples. Cada pàgina dels *bitmaps* disposa de 500 bytes. A té 20 valors diferents i B en té 5. Digueu quants segons costaria (considereu  $D = 1$  i  $C = 0$ ) utilitzar aquests *bitmaps* versus fer una lectura total de la taula per a les consultes següents:

(a)

```
SELECT * FROM T WHERE A IN (1,2) AND B=1;
```

(b)

```
SELECT * FROM T WHERE A IN (1,2,3) OR B=1;
```

6. Com bé sabeu, la fórmula del cost d'accedir a diverses tuples amb un B+ assumeix que per a cada adreça trobada a l'índex es fa un accés a la taula. En canvi, amb un bitmap s'assumeix que un bloc no és accedit més d'una vegada.

Oracle implementa un camí d'accés (Table Access by Rowid Batched) que en el millor dels casos equival a fer servir les adreces trobades en un B+ com en el cas dels bitmaps, llegint els blocs que calgui una sola vegada.

---

<sup>5</sup>Consider 1K=1000bytes.

Anem a comparar el funcionament d'aquestes tres estructures: B+, B+ Batched (B+B) i Bitmap (BM). Ompliu les taules següents amb aquests acrònims (B+, B+B o BM) de manera que quedin ordenats segons l'espai ocupat i el cost de la consulta. No cal que doneu el cost o espai, només que ordeneu les estructures; en cas que en algun cas no es pugui determinar l'ordre, indiqueu quina informació us falta per poder-ho fer. En cas d'empat, poseu dos o tres acrònims a la mateixa casella.

(Q) SELECT \* FROM T WHERE A = <valor>;

$|T| = 1,000,000$  bitsPerBlock = 10,000 d = 150

(a)  $\text{ndist}(A) = 20$

	ESPAI		COST Q
menys (-)		menys (-)	
més (+)		més (+)	

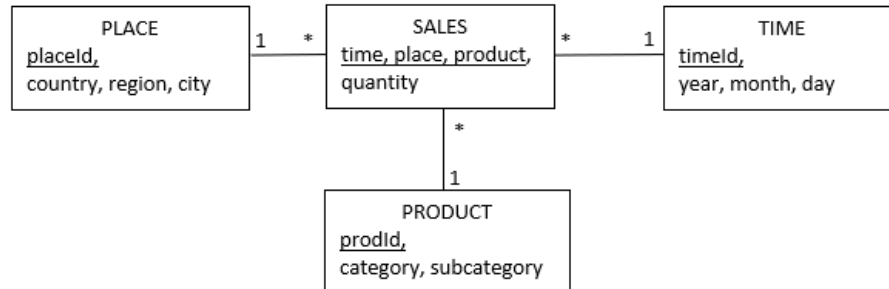
(b)  $\text{ndist}(A) = 300$

	ESPAI		COST Q
menys (-)		menys (-)	
més (+)		més (+)	

## 8 Parametrization and Tuning

### 8.1 Problems

1. Per resoldre consultes multidimensionals, es pot mirar d'aprofitar el fet que la clau primària d'un fet està formada per la unió de les claus foranes que referencien les dimensions associades al fet. Per exemple



Això permet resoldre totes les JOIN de cop si obtenim de cada dimensió involucrada a la consulta els identificadors de les tuples que compleixen la condició, fem el producte cartesià dels conjunts d'identificadors obtinguts i accedim a la taula del fet amb les combinacions d'identificadors generades. Donada aquesta consulta,

```

SELECT category FROM
sales S JOIN time T ON (S.time = T.timeId) JOIN place P on (P.placeId = S.place) JOIN
product R ON (S.product = R.prodId)
WHERE T.year = 2020 AND P.city IN ( 'Barcelona', 'Palma', 'Valencia' )
  
```

quins índexs definiries si treballassis amb un SGBD que tingués implementada aquesta estratègia i la volguessis fer servir?

Tipus índex	Taula	Atributs

I si només hi hagués els algorismes de JOIN que hem vist a classe?

Tipus índex	Taula	Atributs

2. Considereu la taula i estadístiques i informacions següents:

Vot(munic, col, mesa, cand)

$ Vot  = 10\,000\,000$	$B_{Vot} = 50\,000$	$d = 75$	$M = 100$	Bytes per pàg = 1000
$ndist(munic) = 10\,000$	$ndist(col) = 2000$	$ndist(mesa) = 5$		$ndist(cand) = 10$
Els atributs ocupen 10 bytes cadascun		bits = 50 000		

Apliqueu l'algoritme greedy de tria d'índexs per a les consultes següents:

(50%) `SELECT cand , COUNT(*) FROM Vot GROUP BY cand`  
 (30%) `SELECT munic , COUNT(*) FROM Vot WHERE cand = 'X' GROUP BY munic`  
 (20%) `SELECT COUNT(*) FROM Vot WHERE cand IN ( 'X' , 'Y' ) AND munic BETWEEN ( 'K1' and 'K2' )`

Ompliu la taula de la primera volta (suposeu que hi ha 100 municipis amb el nom entre K1 i K2):

Índex - tipus, atribut(s) -	Cost Q1	Cost Q2	Cost Q3

Quins índexs es trien al final (poseu-los en l'ordre en què són triats):

Quin és el cost final de cada consulta? Q1:

Q2:

Q3:

La solució obtinguda és òptima?

En cas negatiu, digues una solució millor:

## 9 Transactions

### 9.1 Theory

1. Tenemos un backup y un log de una base de datos. Desafortunadamente, el log ha sido destruido. ¿Podemos reconstruirlo a partir del backup y la BD? ¿Cómo/Por qué?

Ahora supongamos que perdemos la BD y el log. ¿Podemos reconstruirla? ¿Cómo/Por qué?

### 9.2 Problems

1. (*Solució disponible*) Donades aquestes dues transaccions que efectuen aquestes operacions de lectura i escriptura:

INSTANT	T1	T2
1	START	
2		START
3		
4	R(A)	
5		
6		
7		W(A)
8		
9		W(B)
10		
11		
12		COMMIT

INSTANT	T1	T2
13		
14	R(B)	
15		
16		
17	COMMIT	
18		
19		
20		
21		
22		
23		
24		

- Encerleu el nivell d'aïllament mínim que cal perquè no es produeixi cap interferència:

Cap      Read Uncommitted      Read Committed      Repeatable Read      Serializable

- Afegiu les operacions de bloqueig i desbloqueig que es faran segons el nivell d'aïllament triat (a cada instant, només una operació d'una transacció es pot efectuar). Desplaceu, si cal, les operacions de lectura/escriptura.
- Canvia alguna cosa si intercanviem d'instant les operacions d'escriptura de T2?

2. (*Solució disponible*) Una possible interferència entre transaccions que s'executen en paral·lel és la que s'anomena anàlisi inconsistent: "Consider a situation, where one transaction is applying the aggregate function on some records while another transaction is updating these records. The aggregate function may calculate some values before the values have been updated and others after they are updated. Example:



T1	T2
<code>read_item(X)</code> <code>X = X - N</code> <code>write_item(X)</code>	<code>sum = 0</code> <code>read_item(A)</code> <code>sum = sum + A</code>
<code>read_item(Y)</code> <code>Y = Y + N</code> <code>write_item(Y)</code>	<code>read_item(X)</code> <code>sum = sum + X</code> <code>read_item(Y)</code> <code>sum = sum + Y</code>

In the above example, transaction 2 is calculating the sum of some records while transaction 1 is updating them. Therefore the aggregate function may calculate some values before they have been updated and others after they have been updated.”<sup>6</sup>

- a) Digueu quin és el nivell mínim d’aïllament que s’ha d’establir per tal d’evitar aquesta interferència.
  - b) Reescriuiu la seqüència d’operacions anteriors afegint-hi, on calgui segons el nivell que heu dit a l’apartat a), operacions `lock_item(...)` i `unlock_item(...)`. Reescriuiu-la de manera que es vegi en quin ordre s’acabaran executant les operacions tenint en compte els bloquejos.
3. Per a cada situació, digueu si es pot recuperar la BD i tornar-la a posar en funcionament i, en cas afirmatiu, quina o quines de les operacions següents cal dur a terme: RESTAURAR, RECONSTRUIR, FER BACKUP.

	Recuperable?	Operacions necessàries
a) Pèrdua disc dades		
b) Pèrdua disc backup		
c) Pèrdua disc dietari		
a) i b)		
a) i c)		
b) i c)		
a), b) i c)		
Disc dietari ple		
Espai de dades ple		

4. Donat el problema corresponent a cada columna, indiqueu marcant amb un *tick* (anoteu també el total de *ticks* per fila i columna) les caselles corresponents als mecanismes/componentes que són necessaris per garantir en tot moment les propietats ACID:<sup>7</sup>

<sup>6</sup><https://www.geeksforgeeks.org/concurrency-problems-in-dbms-transactions>

<sup>7</sup>Podem identificar RW com a un entorn OLTP i RO com a un entorn DW.

<i>Hardware Failure</i>	No falla	No falla	No falla	No falla	Falla	Falla	Falla	Falla	
<i>Power Failure</i>	No falla	No falla	Falla	Falla	No falla	No falla	Falla	Falla	
<i>Concurrent Read/Write vs Read-Only User Tx</i>	RO	RW	RO	RW	RO	RW	RO	RW	<b>Total</b>
<i>Backup</i>									
<i>Recovery Manager (Log)</i>									
<i>Cache Manager</i>									
<i>Scheduler</i>									
<b>Total</b>									