

Solution

ExoN 1

1-Trois types :aucune matérialisation , matérialisation partielle, matérialisation complète(Multi way Array Aggregation), La plus faisable est la partielle **1Pt**

2- Select type , store sum(Number)
From Pets Group by Rollup(type,store) **1.5 Pt**

Select type , store sum(Number) From Pets
Group by Cube(type,store) **1.5 Pt**

Type	Store	Number
Cat	Miami	18
Cat	Naples	9
Cat	NULL	27
Dog	Miami	12
Dog	Naples	5
Dog	Tampa	14
Dog	NULL	31
Turtle	Naples	1
Turtle	Tampa	4
Turtle	NULL	5
NULL	NULL	63

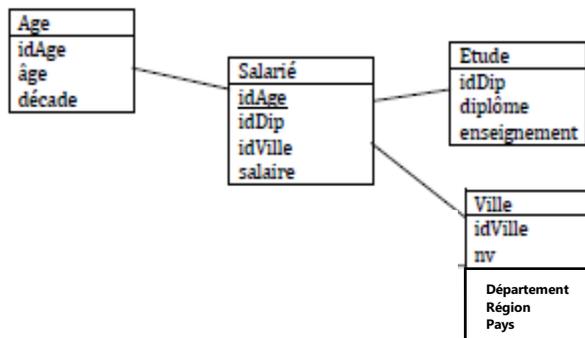
Cat	Miami	18
Cat	Naples	9
Cat	NULL	27
Dog	Miami	12
Dog	Naples	5
Dog	Tampa	14
Dog	NULL	31
Turtle	Naples	1
Turtle	Tampa	4
Turtle	NULL	5
NULL	NULL	63
NULL	Miami	30
NULL	Naples	15
NULL	Tampa	18

3- 1Pt

ROWID	C	B0	B1	B2	B3
0	2	0	0	1	0
1	1	0	1	0	0
2	3	0	0	0	1
3	0	1	0	0	0
4	3	0	0	0	1
5	1	0	1	0	0
6	0	1	0	0	0
7	0	1	0	0	0
8	2	0	0	1	0

Exo N 2

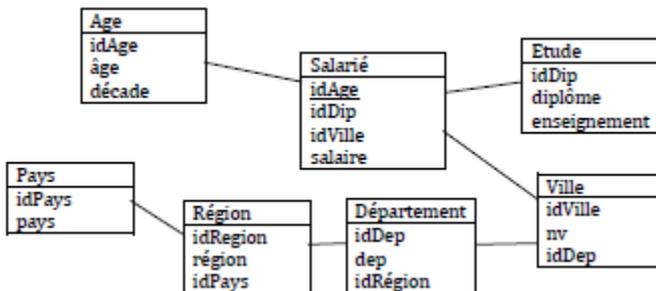
1- 1Pt



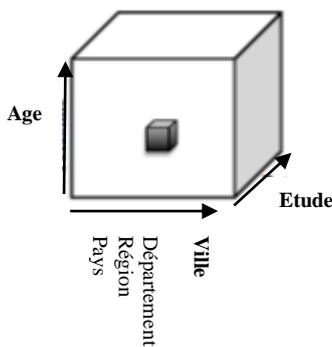
2- Salarié, **0.5Pt** 3-Un seul fait salaire, **0.5Pt** 4-Trois dimensions: Ville, Etude et Age **0.75Pt**

5- ville → Département → Région → Pay **0.75Pt**

6-1Pt



7- **0.5Pt**



Exo N° 3

1- a-sup(X,Y) <-SupDirect(X,Y)

sup(X,Y) <-SupDirect(X,Z) AND sup(Z,Y) **1Pt**

b-supCommun(X,Y,Z) <-sup(X,Y) AND sup(X,Z) **0.5Pt**

c-memeNiveau(X,Y) <-supDirect(Z,X) AND supDirect(Z,Y)

memeNiveau(X,Y) <-supDirect(Z1,X) AND supDirect(Z2,Y) AND memeNiveau(Z1,Z2)

memeNiveau(X,Y) <-X = Y AND Employe(X) AND Employe(Y) **1.5Pt**

2-	<u>SupDirect</u>		<u>Sup</u>	1Pt	<u>memeNiveau</u>	1Pt
	Chafik	Labib	Chafik	Labib	Chafik	Chafik
	Chafik	Chemsseddine	Chafik	Chemsseddine	Chemsseddine	Chemsseddine
	Labib	Amine	Labib	Amine	Labib	Labib
			Chafik	Amine	Amine	Amine
					Labib	Chemsseddine
					Chemsseddine	Labib

Exo N 4

1^{er} choix : D'après l'algorithme glouton, la vue sommet doit être choisie en premier.

Ensuite, on choisit la vue c qui a un bénéfice égal à $4141 = (200 - 99) * 41$. **2Pt**

Avec : 200 coût de a. 99 de c.

41 est le nombre de descendants de c plus la vue c elle-même.

2^{ème} choix : On peut prendre b ou d, toutes les deux ont un bénéfice égal à $2100 = (200 - 100) * 21$.

Avec : 200 coût de a. 100 coût de b.

21 est le nombre de descendants de b plus la vue b elle-même.

Alors avec $k = 2$, l'algorithme glouton produit une solution avec un bénéfice égal à $B_c + B_b = 4141 + 2100 = 6241$. **2Pt**

Cependant, la solution optimale est celle qui prend b et d. toutes les deux réduisent le coût de 100 ainsi que les 80 vues des quatre chaînes. $B(b, S) = (200-100) * 41 = 4100$. Même bénéfice pour d. Donc la solution optimale a un bénéfice de $8200 = 4100 * 2$.

Le rapport du bénéfice de l'algorithme glouton / l'algorithme optimal est $6241/8200$, qui est $3/4$. **1Pt**

Des preuves théoriques dans [Har 96] démontrent que le bénéfice de l'algorithme glouton est estimé à au moins 63% du bénéfice de l'algorithme optimal et pas moins.

C'est un mauvais exemple de l'algorithme glouton