

Requêtes d'agrégation en SQL

Dr N. BAME

Introduction

- Jusqu'à présent on a vu des requêtes comme **suite d'opérations tuple à tuple**
- Les *fonctions d'agrégation* permettent d'exprimer des conditions et de faire des opérations sur des **groupes de tuples**
- Avec SQL nous pouvons:
 - **Partitionner** une relation **en groupes**
 - Exprimer des relations sur des **groupes**
 - **Agréger** des valeurs **sur les groupes**

SQL : Fonctions d'agrégation

Elles peuvent servir à calculer une valeur numérique à partir d'une relation : **Agg(A)** où A est un d'attribut (ou *) et Agg est une fonction parmi :

- **COUNT(A)** ou **COUNT(*)** : nombre de valeurs ou n-uplets,
- **SUM(A)** : somme des valeurs,
- **MAX(A)** : valeur maximale,
- **MIN(A)** : une valeur minimale,
- **AVG(A)** : moyenne des valeurs

dans *l'ensemble de valeurs* désignées par **A**

Agrégation sur l'agrégat par défaut

On applique des fonctions de calcul sur le résultat d'une requête

```
SELECT Agg1(Ai), ..., AggN(Aj)  
FROM R1, ..., Rm  
WHERE Condition ;
```

- On exécute la requête (clause from et where).
- On applique les différentes **fonctions Aggk sur l'ensemble du résultat.**

Agrégation sur l'agrégat par défaut

quelques remarques

La fonction d'agrégation **Agg** peut s'appliquer

- Sur **une ligne**, uniquement pour le **COUNT** : **COUNT(*)**, pour compter le nombre de ligne du résultat d'une requête
- Sur **un attribut** : **Agg(att)** on applique la fonction aux **valeurs NON NULLES** de l'attribut att
- Sur **un attribut avec DISTINCT** : **Agg(DISTINCT att)** pour appliquer la fonction aux **valeurs distinctes** de l'attribut *att*

Exemples d'agrégation

Emp(Eno, Ename, Title, City)

Pay(Title, Salary)

Project(Pno, Pname, Budget, City)

Works(Eno, Pno, Resp, Dur)

- Nombre d'employés Lougatois ?
- Plus grand salaire et plus petit salaire de toutes les professions?

Exemples d'agrégation

Emp(Eno, Ename, Title, City)

Pay(Title, Salary)

Project(Pno, Pname, Budget, City)

Works(Eno, Pno, Resp, Dur)

- Nombre d'employés Lougatois ?

```
SELECT COUNT(*)  
FROM Emp  
WHERE City='Louga'
```

- Plus grand salaire et plus petit salaire de toutes les professions?

```
SELECT MAX(Salary), MIN(Salary)  
FROM Pay
```

Exemples d'agrégation

Emp(Eno, Ename, Title, City)

Pay(Title, Salary)

Project(Pno, Pname, Budget, City)

Works(Eno, Pno, Resp, Dur)

- Budgets totaux des projets de Thies?
- Nombre de villes où il y a un projet avec l'employé E4?

Exemples d'agrégation

Emp(Eno, Ename, Title, City)

Pay(Title, Salary)

Project(Pno, Pname, Budget, City)

Works(Eno, Pno, Resp, Dur)

- Budgets totaux des projets de Thies?

```
SELECT SUM (Budget)
```

```
FROM Project
```

```
WHERE City = 'Thies'
```

- Nombre de villes où il y a un projet avec l'employé E4?

```
SELECT COUNT(DISTINCT City)
```

```
FROM Project, Works
```

```
WHERE Project.Pno = Works.Pno
```

```
AND Works.Eno = 'E4'
```

Agrégation sur l'agrégat par défaut quelques remarques

Attention aux agrégations dans le WHERE :

```
SELECT TITLE
```

```
FROM Pay
```

```
WHERE Salary = MAX(Salary)
```

=> cette formulation est INTERDITE !

- Pas d'attribut dans le SELECT si agrégation (sauf si GROUP BY voir après)

```
SELECT TITLE, MAX(Salary)
```

```
FROM Pay
```

=> cette formulation est INTERDITE !

Comment faire pour connaître alors la profession payant le plus?

Fonctions d'agrégation et imbrication

Emp(Eno, Ename, Title, City)

Project(Pno, Pname, Budget, City)

Pay(Title, Salary)

Works(Eno, Pno, Resp, Dur)

Pour utiliser des attributs de tables du FROM et une agrégation, il faut souvent passer par une sous-requête :

- Noms des professions qui payent le plus (et les salaires correspondant) ?

SELECT Title, Salary

FROM Pay

WHERE Salary = (**SELECT MAX**(Salary)
FROM Pay)

Fonctions d'agrégation et imbrication

Emp(Eno, Ename, Title, City)

Pay(Title, Salary)

Project(Pno, Pname, Budget, City)

Works(Eno, Pno, Resp, Dur)

- Noms des projets dont le budget est supérieur au budget moyen?

SELECT Pname

FROM Project

WHERE Budget > (**SELECT AVG(Budget)**
FROM Project)

Remarque : dans le cas où la requête imbriquée retourne le résultat d'un agrégat (donc un seul résultat), pas besoin d'utiliser un IN, ANY ou ALL (mais pas faux)

Requêtes de groupement

- Pour partitionner les n-uplets résultats en fonction des valeurs de certains attributs :

```
SELECT Ai, ..., An agg1, agg2, ...  
FROM R1, ..., Rm  
WHERE P  
GROUP BY Aj, ..., Ak
```

Principes :

- On exécute la requête **FROM WHERE**
- On **regroupe le résultat en paquets** d'enregistrements en plaçant dans un paquet tous les enregistrements **ayant la même valeur** pour A_j, ..., A_k
- On fait le SELECT en appliquant les agrégations à chaque paquet

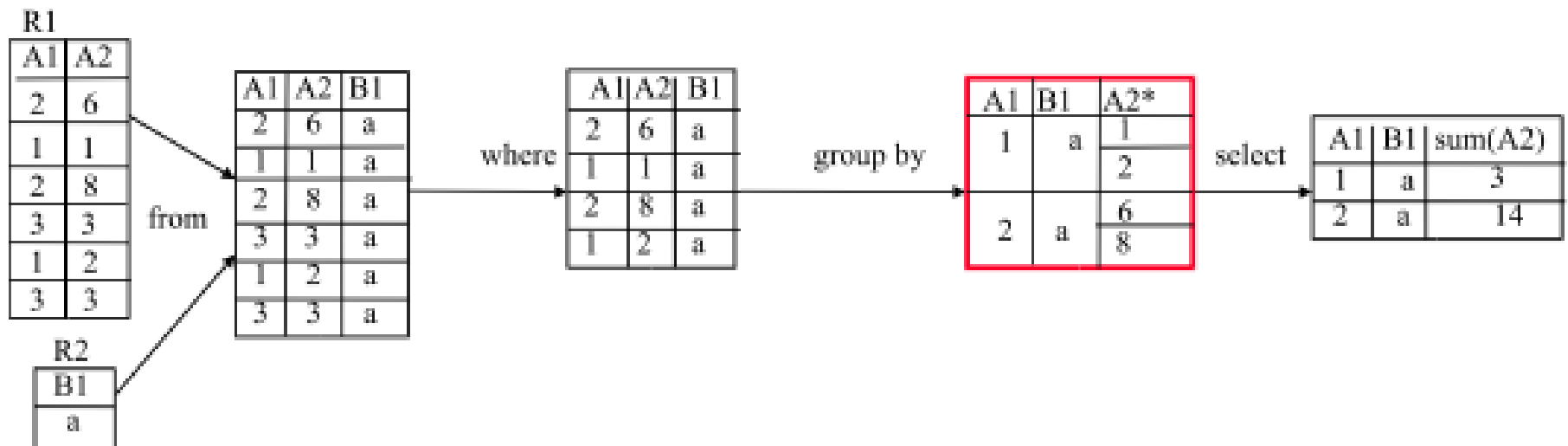
GROUP BY

SELECT A1, B1, sum(A2)

FROM R1, R2

WHERE A1 < 3

GROUP BY A1, B1



GROUP BY : remarques

Tous les **attributs** (A_i, \dots, A_n) dans la clause SELECT :

- **Ne doivent pas apparaître** dans une opération d'agrégation et
- **doivent être inclus dans** l'ensemble des **attributs** (A_j, \dots, A_k) **de la clause GROUP BY** (qui peut avoir d'autres attributs en plus)

Le SELECT ne retourne qu'**une seule ligne par agrégat** (paquet)

=> Impossibilité de mettre un attribut non présent dans le GROUP BY car prend plusieurs valeurs dans un même paquet

Exemples de groupement

Emp(Eno, Ename, Title, City)

Pay(Title, Salary)

Project(Pno, Pname, Budget, City)

Works(Eno, Pno, Resp, Dur)

- Numéros des projets avec leurs effectifs ?

```
SELECT Pno, Count(Eno)
```

```
FROM Works
```

```
GROUP BY Pno
```

- Pour chaque ville, nombre d'employés par profession?

```
SELECT City, Title, Count(*)
```

```
FROM Emp
```

```
GROUP BY City, Title
```


Exemples de groupement

Emp(Eno, Ename, Title, City)

Pay(Title, Salary)

Project(Pno, Pname, Budget, City)

Works(Eno, Pno, Resp, Dur)

- Numéros des projets avec leurs effectifs Numéros et noms des projets avec la durée moyenne et la durée maximale de participation des employés

```
SELECT Pno, Pname, AVG(Dur), MAX(Dur)
FROM Works, Project
WHERE Works.Pno=Project.Pno
GROUP BY Pno, Pname
```

=> OK pour ajouter Pname,
la clé de Projet est dans
GROUP BY

On a ajouté Pname dans le GROUP BY (alors que Pno suffisait pour faire les agrégats) afin de le mettre dans le SELECT

- Numéro,nom, et responsable des projets et la durée maximale de participation d'un employé à ce projet?

```
SELECT Pno, Pname, resp, MAX(Dur)
FROM Works, Project
WHERE Works.Pno=Project.Pno
GROUP BY Pno, Pname
```

=>INTERDIT,pas de clé de
Works dans GROUP BY

Prédicats sur des groupes

Pour garder (éliminer) les groupes (partitions) qui satisfont (ne satisfont) pas une certaine condition :

SELECT $A_i, \dots, A_n, \text{agg1}, \text{agg2}, \dots$

FROM R_1, \dots, R_m

WHERE P

GROUP BY A_j, \dots, A_k

HAVING Q

Principe : une fois la requête FROM-WHERE exécutée, on regroupe en agrégat avec le GROUP BY et on ne garde ensuite que les agrégats satisfaisants le HAVING

Remarques

Ordre d'exécution des clauses :

- 5 SELECT
- 1 FROM
- 2 WHERE
- 3 GROUP BY
- 4 HAVING
- 6 ORDER BY

Conséquence : à partir de 3 on manipule des agrégats donc pour 4, 5 et 6 on ne manipule que des *fonctions d'agrégations ou des attributs apparaissant dans le GROUP BY*

Exemples de groupement (1)

Emp(Eno, Ename, Title, City)

Pay(Title, Salary)

Project(Pno, Pname, Budget, City)

Works(Eno, Pno, Resp, Dur)

- Villes dans lesquelles habitent plus de 2 employés?

```
SELECT City
FROM Emp
GROUP BY City
HAVING COUNT(ENO) > 2
```

- Projets demandant plus de 1000 jours/homme?

```
SELECT Pno, Pname
FROM Project, Works
WHERE Projet.Pno=Work.Pno
GROUP BY Pno, Pname
HAVING SUM(Dur) > 1000
```

Exemples de groupement (2)

Emp(Eno, Ename, Title, City)
Pay(Title, Salary)

Project(Pno, Pname, Budget, City)
Works(Eno, Pno, Resp, Dur)

- Numéros et noms des projets ayant des employés venant de plus de 5 villes différentes ?

```
SELECT Pno, Pname
FROM Project, Works, Emp
WHERE Projet.Pno=Work.Pno AND Work.Eno=Emp.Eno
GROUP BY Pno,Pname
HAVING COUNT(DISTINCT Emp.City) > 5
```

- Liste des employés triés par temps total de travail décroissants?

```
SELECT Eno, Ename, SUM(Dur)
FROM Works, Emp
WHERE Work.Eno=Emp.Eno
GROUP BY Eno,Ename
ORDER BY SUM(Dur) DESC
```