

SAM cours 6: requêtes réparties

Exemples

Fragments:

$$\left. \begin{array}{l} \text{Client}_1 = \sigma_{\text{ville} = \text{'Paris'}} \text{Client} \\ \text{Client}_2 = \sigma_{\text{ville} \neq \text{'Paris'}} \text{Client} \\ \text{Cde1} = \pi_{\text{nclient}}(\text{Cde}) \\ \text{Cde2} = \pi_{\text{nclient, produit, qté}}(\text{Cde}) \end{array} \right\} \begin{array}{l} \text{Client} = \text{Client}_1 \cup \text{Client}_2 \\ \text{Cde} = \dots \end{array}$$

- Requêtes :
- a) **Select distinct nom From Client**
 - b) **$\sigma_{\text{ville} = \text{'Lyon'}} \text{Client}$**
 - c) **Select distinct nclient from Cde**

Méthode: substitution, éliminer l'accès aux fragments inutiles ?

Exemples

Fragments:

Site 1 : $\text{Client}_1 = \sigma_{\text{ville} = \text{'Paris'}} \text{Client}$

Site 2 : $\text{Client}_2 = \sigma_{\text{ville} \neq \text{'Paris'}} \text{Client}$

Site 3 : $\text{Cde1} = \text{Cde} \bowtie \text{Client}_1$

Site 4 : $\text{Cde2} = \text{Cde} \bowtie \text{Client}_2$

Requêtes :

Sur S1: Select * from Client, Cde

where Client.nclient = Cde.nclient and ville='Lyon'

Sur S5: Select * from Client, Cde

where Client.nclient = Cde.nclient and qté > 10

Coût des Solutions

Supposons

- taille (Cde1) = taille (Cde2) = 10 000
- taille (Client1) = taille (Client2) = 2 000
- coût de transfert de 1n-uplet = 1
- Sélectivité($qté > 10$) = 1%

Stratégie 1

- transfert de Cde1 + Cde2 = 20 000
- transfert de Client1 + Client2 = 4000

Stratégie 2

- transfert de Client1 + Client2 = 4000
- transfert de C1 + C2 = 200

Stratégie 3

- transfert de Client1 + Client2 = 4000
- Transfert de T1 + T2 = 200

Stratégie 4

- transfert de C1 + C2 = 200
- transfert de C3 + C4 = 200

Jointure

R sur S1, S sur S2, T sur S3.

Requête demandée sur le site S0 : $R \bowtie S \bowtie T$

Plusieurs possibilités :

a) Copier tout sur S0 et faire les jointures sur S0

b) Copier R sur S2, et joindre R et S sur S2

Copier le résultat sur S3, et faire la jointure avec T sur S3

Copier le résultat sur S0

Tenir compte des index, de la taille des relations à transférer, de la taille des relations intermédiaires, de la charge des sites, de la vitesse de transmission, etc.

On peut paralléliser les jointures : grand nombre de stratégies

Semi-jointure

Traiter la jointure entre deux relations réparties sur deux sites :

R1 sur S_1 et R2 sur S_2

Rappel : $R1 \bowtie R2 = \pi_{\text{Att de R1}} (R1 \bowtie R2)$

Requête $R1 \bowtie_A R2 = R1 \bowtie_A (R2 \bowtie_A R1)$ et le résultat doit être sur S_1

Sur S_1 : $T1 = \pi_A (R1)$, puis envoi de T1 sur S_2

Sur S_2 : $T2 = R2 \bowtie_A T1$, puis envoi de T2 sur S_1

Sur S_1 : Calcul de $R1 \bowtie_A T2$

Requête $R1 \bowtie R2 = (R1 \bowtie R2) \bowtie (R2 \bowtie R1)$ et résultat sur S_0

Sur S_1 : Transférer $T1 = \pi_A (R1)$ vers S_2

Sur S_2 : Transférer $T2 = \pi_A (R2)$ vers S_1

Sur S_1 : Transférer $T3 = R1 \bowtie R2$ vers S_0

Sur S_2 : Transférer $T4 = R2 \bowtie R1$ vers S_0

Sur S_0 : $T3 \bowtie T4$

ne transférer des nuplets
complets que s'ils joignent

Semi jointure en TME

- S1 : Club (cnum, division, ville, description)
- S2 : Joueur (prenom, age, cnum, profil)

Requête posée sur S1:

```
Select c.description, j.profil  
From Club, Joueur  
Where c.cnum = j.cnum  
And ville = 'Aix'
```

Méthode: sur S1: T1 = les cnum des clubs situés à Aix

sur S2: T2 = liste des joueurs dans un club de T1

sur S1: jointure T2 et Club pour avoir la description de chaque club