Master d'Informatique
spécialité DAC
BDLE (Bases de Données Large Echelle)
-Seconde Partie-

Cours 2 : Requêtes relationnelles en M/R

Mohamed-Amine Baazizi – email: prénom.nom@lip6.fr
http://dac.lip6.fr/master/ues-2014-2015/bdle-2014-2015/

Objectifs

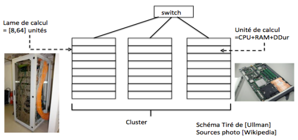
1. Traduction des requêtes relationnelles en MR
2. Estimation du coût d'un programme MR

Coût d'un programme MR

Rappel : architecture en grappe

Deux facteurs :

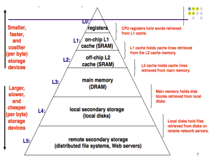
1. Communication : transfert des données entre noeuds
2. Calcul : exécution des *maps* et des *reduces* (accés disque)



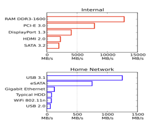
Différents modèles de coût

Facteur \ Modèle	*Orienté communication*	*Orienté calcul*
Communication	✓	✗
Calcul	✗	✓
	[Afrati et al.]	[Suciu et al.]
	Ignorer les calculs car considérés simples Considérer les transferts car hiérarchie des mémoires	Fixer les transferts ➔ coût de communication constant

Hiérarchie des mémoires



Source [Cours1]



Source [Mem]

Différents modèles de coût

Facteur \ Modèle	*Orienté communication*	*Orienté calcul*
Communication	✓	✗
Calcul	✗	✓
	[Afrati et al.]	[Suciu et al.][??]
	Ignorer les calculs car considérés simples Considérer les transferts car hiérarchie des mémoires	Fixer les transferts ➔ coût communication constant

Modèle orienté communication

Rappel

Communication = transfert données entre les noeuds

Transfert de données :

- *Maps* vers *reduces* : les paires clés valeurs
- *Reduces* vers *maps* : résultat d'une étape MR

Remarque

- Les *maps* s'exécutent localement, pas de transfert

Modèle orienté communication Illustration

Jointure naturelle R(A, B) et S(B, C)

Rappel :

Map pour chaque (a, b) de R produire (b, (R, a))
pour chaque (b, c) de S produire (b, (S, c))

Reduce
Produire (b, $\{a_1,..,a_k\} \times \{b\} \times \{c_1,..,c_m\}$)

Coût = |R|+|S| = coût pour mettre à disposition les données aux reduces

Modèle orienté communication opérateurs algébriques

- Opérateurs ensemblistes :
  - Union : ∪
  - Intersection : ∩
  - Différence : −
- Autres opérateurs :
  - Projection : π_X
  - Sélection : σ_C
  - Jointure naturelle : ⋈
  - Renommage : $\rho_{A\text{->}B}$
  - Produit cartésien : ×
  - Division : ÷

Coût : |R| si opérateur unaire
|R|+|S| si opérateur binaire

Question : opérateurs n-aires?

9

Modèle orienté communication opérateurs algébriques n-aires

Objectif : une seule passe MR

Hypothèse : $R_1 \ldots R_n$ n schémas de relations

Opérateurs ensemblistes : sch(R_i) identiques

- Union coût = $\Sigma\, |R_i|$
- Intersection coût = $\Sigma\, |R_i|$

Jointure : R(A,B) ⋈ S(B,C) ⋈ T(C,D)

- coût $=^?$ $\Sigma\, |R_i|$
- Généralisation de la jointure à 2 relations?

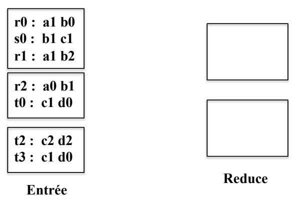
10

Jointure à n relations

Objectif : une seule passe MR

Hypothèse : n=3 et R(A,B), S(B,C) , T(C,D)

Exemple



Jointure à n relations

Objectif : une seule passe MR

Hypothèse : n=3 et R(A,B), S(B,C) , T(C,D)

Tentative 1 : généralisation 'naïve'

Map pour chaque (a, b) de R produire (b, (R, a))
pour chaque (b, c) de S produire (b, (S, c))
pour chaque (c, d) de T produire (c, (T, d))

(b, (R,a)) (b, (S,c)) <u>(b, [(R, a)..(S, c)])</u> (c, (T,d))

Constat ?

Jointure à n relations

Objectif : une seule passe MR

Hypothèse : n=3 et R(A,B), S(B,C) , T(C,D)

Tentative 2 : généralisation **moins** 'naïve'

Map pour chaque (a, b) de R produire (b, (R, a))
pour chaque (b, c) de S produire (b, (S, c))
pour chaque (c, d) de T produire (c, (T, d))
pour chaque (b, c) de S produire (c, (S, b))

(b, (R,a)) (b, (S,c)) (b, [(R, a)..(S, c)]) (c, (T,d)) (c,[(S,b)..(T,d)])

Constat ?

Jointure à n relations

Objectif : une seule passe MR

Hypothèse : n=3 et R(A,B), S(B,C) , T(C,D)

Tentative 3 : la bonne!

Question : que doit-on avoir du côté Reduce?

Réponse : les tuples de R, S et T correspondant à une combinaison de valeur de B et de C

Mise en œuvre : deux fonctions de hachage h et g

$h : \{b_0..b_n\} \rightarrow D_B$ et $g : \{c_0..c_m\} \rightarrow D_C$
avec $|D_B|\times|D_C| = |reduces| = k$

Jointure à n relations

Objectif : une seule passe MR

Hypothèse : n=3 et R(A,B), S(B,C) , T(C,D)

Tentative 3 : la bonne! *(suite)*

Map chaque (a, b) de R, $r_i \rightarrow$ les $|D_C|$ buckets **(h(b), y)**
chaque (b, c) de S, $s_j \rightarrow$ le seul bucket **(h(b), g(c))**
chaque (c, d) de T, $t_l \rightarrow$ les $|D_B|$ buckets **(x, g(c))**

Reduce considérer les buckets où il y a des tuples de S

Jointure à n relations

Objectif : une seule passe MR

Hypothèse : n=3 et R(A,B), S(B,C) , T(C,D)

Tentative 3 : la bonne! *(suite)*

Map chaque (a, b) de R, r_i → les $|D_C|$ buckets **(h(b), y)**
chaque (b, c) de S, s_j → le seul bucket **(h(b), g(c))**
chaque (c, d) de T, t_l → les $|D_B|$ buckets **(x, g(c))**

Reduce considérer les buckets où il y a des tuples de S

Coût du reduce $|S| + |D_C| \times |R| + |D_B| \times |T|$

Jointures circulaire

Objectif : une seule passe MR

Hypothèse : n=3 et R(A,B), S(B,C) , T(C,A)

Questions : généraliser la solution précédente.
Formule de coût?

Jointures à n relations

- Solution 'mono-passe' vs solution deux à deux
  - Comparaison de coûts
  - Coût jointure n-aire mono-passe
    $|S| + |D_B| \times |R| + |D_C| \times |T|$
    avec $|D_B| \times |D_C| = |reducers| = k$

Question : partitionner k pour avoir coût minimum?

Réponse: équations de Lagrange
$C = |S| + |D_B| \times |R| + |D_C| \times |T| - \lambda(|D_B| \times |D_C| - k) = 0$
Solutions à partir de $dC/d|D_B|$ et $dC/d|D_C|$
$|D_B| = (k|R|/|T|)^{1/2}$ et $|D_C| = (k|T|/|R|)^{1/2}$

Jointures à n relations

- Solution 'mono-passe' vs solution deux à deux
  - Comparaison de coûts
  - Coût jointure n-aire mono-passe
  - Coût jointure n-aire deux à deux

    p_{rs} (p_{st}) proba que tuples R et S (S et T) joignables
    selon ordre : si R,S puis T alors $|R|+|S|+p_{rs}|R||S|+|T|$

Jointures à n relations

Considérons R=S=T avec R(u,v) et $R \bowtie R \bowtie R$
(jointure pour trouver chemins de longueur 3)
$|\pi_u R|=3.10^8$ et à chaque u correspond 300 v
donc, $|R|=9.10^{10}$.

Solution 1. Jointure en mono-passe

$|R|(k)^{1/3}$ jointure circulaire mono-passe

Solution 2. deux jointures : $R \bowtie R = R_1$ puis $R_1 \bowtie R$

$|R|^2\, p$ *avec p* proba que les tuples de R joignables …

Jointure à n relations

Objectif : une seule passe MR

Hypothèse : R(A,B,C), S(A,B,D) , T(A,D,E), U(D,F)

Question: déterminer le partitionnement de *k*?

Réponse: résoudre l'équation de Lagrange

$|R|\, def + |S|\, cef + |T|\, bcf + |U|\, abce$
$- \lambda(abcdef - k)$

Problème : n'admet pas de solution non nulle

Solution : *Régle du Dominant*

Règle du dominant

Définition :
Etant donné un schéma de relations R, S, …
ayant des attributs X, Y, …
« X domine Y si chaque relation contenant Y alors elle contient aussi X »

Quels attributs sont dominants/dominés?
R(A,B,C), S(A,B,D) , T(A,D,E), U(D,F)

Règle du dominant

Utilisation :

- La taille des partition pour les attributs dominés peut être égale à 1.
- Il reste à partager k entre les attributs dominants

Exemple repris : R(A,B,C), S(A,B,D) , T(A,D,E), U(D,F)

$$|R|\, def + |S|\, cef + |T|\, bcf + |U|\, abce - \lambda(abcdef - k)$$

devient

$$|R|\, d + |S| + |T| + |U|\, a$$

Suite exemple

Exemple repris : R(A,B,C), S(A,B,D) , T(A,D,E), U(D,F)

$$|R|\, def + |S|\, cef + |T|\, bcf + |U|\, abce - \lambda(abcdef - k)$$

devient

$$|R|\, d + |S| + |T| + |U|\, a - \lambda(ad - k)$$

et admet comme solution

$$|R|\, d = \lambda k \text{ et } |U|\, a = \lambda k$$

comme $ad=k$

$$a=(k|R|/|U|)^{1/2} \text{ et } d=(k|U|/|R|)^{1/2}$$

Suite et fin exemple

Exemple repris : R(A,B,C), S(A,B,D) , T(A,D,E), U(D,F)

$a=(k|R|/|U|)^{1/2}$ et $d=(k|U|/|R|)^{1/2}$

- **Map** :
  - ❖chaque r va dans les d buckets (h(A), j) j=1..d
  - ❖chaque u va dans les a buckets (g(D), i) i=1..a
  - ❖chaque s et t vont dans le seul bucket (h(A),g(D))

Illustration

R(A,B,C), S(A,B,D) , T(A,D,E), U(D,F)

$a=(k|R|/|U|)^{1/2}$ et $d=(k|U|/|R|)^{1/2}$

R={(a0,b1,c2), (a1,b0,c2), (a0,b3,c3)}

S={(a1,b1,d0), (a0,b1,d1)}

T={(a1,d0,e1), (a0,d1,e3)}

U={(d1,f3), (d0,f3)}

Préalable : déterminer a et d (valeur entière si nécessaire)

Références

[Afrati et al.] *Optimizing joins in a map-reduce environment*, in Proceedings of EDBT 2010

[Cours1] https://www.cs.rutgers.edu/~badri/211dir/notes/w11-four.pdf

[Mem] en.wikipedia.org/wiki/Random-access_memory

[Suciu et al.] *Parallel evaluation of conjunctive queries*, in Proceedings of PODS 2011