

Base de données distribuées



Babacar Diop

Dpt. d'Informatique

UFR des Sciences Appliquées et de Technologies

Université Gaston Berger de Saint-Louis

2018/2019

Gestion de requêtes dans une BD centralisée

Requêtes

Concept

- Lorsqu'une requête est écrite, elle est d'abord analysée syntaxiquement, sémantiquement et validée
- Une représentation interne de la requête est ensuite créée, telle qu'un arbre ou un graphe de requête
- Des stratégies d'exécution alternatives sont ensuite conçues pour extraire les résultats des tables de la base de données
- Le processus de choix de la stratégie d'exécution la plus appropriée pour le traitement des requêtes est appelé **optimisation des requêtes**

Requêtes

Optimisation

- Dans un SGBDD, l'optimisation des requêtes est cruciale
- La complexité est élevée car le nombre de stratégies alternatives peut augmenter de manière exponentielle en raison des facteurs suivants:
 - La présence d'un certain nombre de fragments,
 - La répartition des fragments ou des tableaux sur différents sites
 - La vitesse des liens de communication
 - La disparité dans les capacités de traitement local

Requêtes

Optimisation

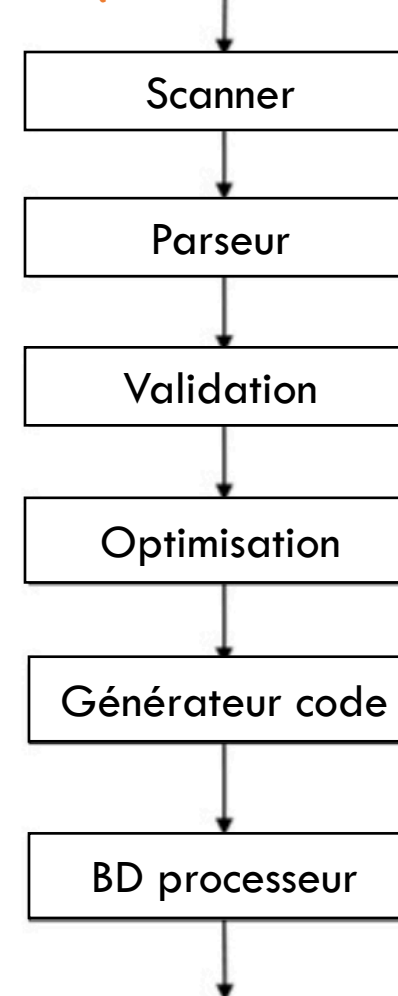
- Par conséquent, dans un système distribué, l'objectif est souvent de trouver une **bonne stratégie d'exécution** pour le traitement des requêtes plutôt que la **meilleure**
- Le temps d'exécution d'une requête est la somme des éléments suivants:
 - temps pour communiquer des requêtes aux bases de données
 - temps pour exécuter les fragments de requête locaux
 - temps pour rassembler les données provenant de différents sites
 - temps pour afficher les résultats dans l'application

Requêtes

Traitement

- Le traitement des requêtes est l'ensemble de toutes les activités allant du placement de la requête à l'affichage des résultats de la requête
- Les étapes sont comme indiquées dans le diagramme suivant (à droite)

Requête sur BDD



Résultat requête

Requêtes

Optimisation dans une BD centralisée

- Le chemin d'accès optimal est déterminé à partir des chemins d'accès produits par calcul algébrique relationnel
- Dans un système centralisé, le traitement des requêtes est effectué dans le but suivant:
 - minimiser le temps de réponse de la requête
 - maximiser le débit du système
 - réduire la quantité de mémoire et de stockage requise pour le traitement
 - augmenter le parallélisme

Requêtes

Optimisation dans une BD centralisée

- Initialement, la requête SQL est analysée pour rechercher les erreurs syntaxiques et l'exactitude des types de données. Si la requête réussit cette étape, la requête est décomposée en blocs de requête plus petits
- Chaque bloc est ensuite traduit en une expression d'algèbre relationnelle équivalente

Requêtes

Optimisation dans une BD centralisée

Étapes pour l'optimisation de la requête

- L'optimisation des requêtes implique trois étapes, à savoir :
 - la génération d'arborescence de requêtes
 - la génération de plans
 - la génération de codes de plans de requêtes

Optimisation dans une BD centralisée

- Une arborescence de requête est une structure de données arborescente représentant une expression d'algèbre relationnelle
- Les tables de la requête sont représentées sous forme de nœuds feuille
- Les opérations d'algèbre relationnelle sont représentées en tant que nœuds internes
- La racine représente la requête dans son ensemble

Optimisation dans une BD centralisée

- Pendant l'exécution, un nœud interne est exécuté chaque fois que ses tables d'opérandes sont disponibles
- Le nœud est ensuite remplacé par la table de résultats
- Ce processus se poursuit pour tous les nœuds internes jusqu'à ce que le nœud racine soit exécuté et remplacé par la table de résultats

Optimisation dans une BD centralisée

- Par exemple, considérons les schémas suivants et la requête ci-dessous -

EMPLOYEE

EmpID	ENAME	SALARY	DEPTNO	DATEOFJOINING
-------	-------	--------	--------	---------------

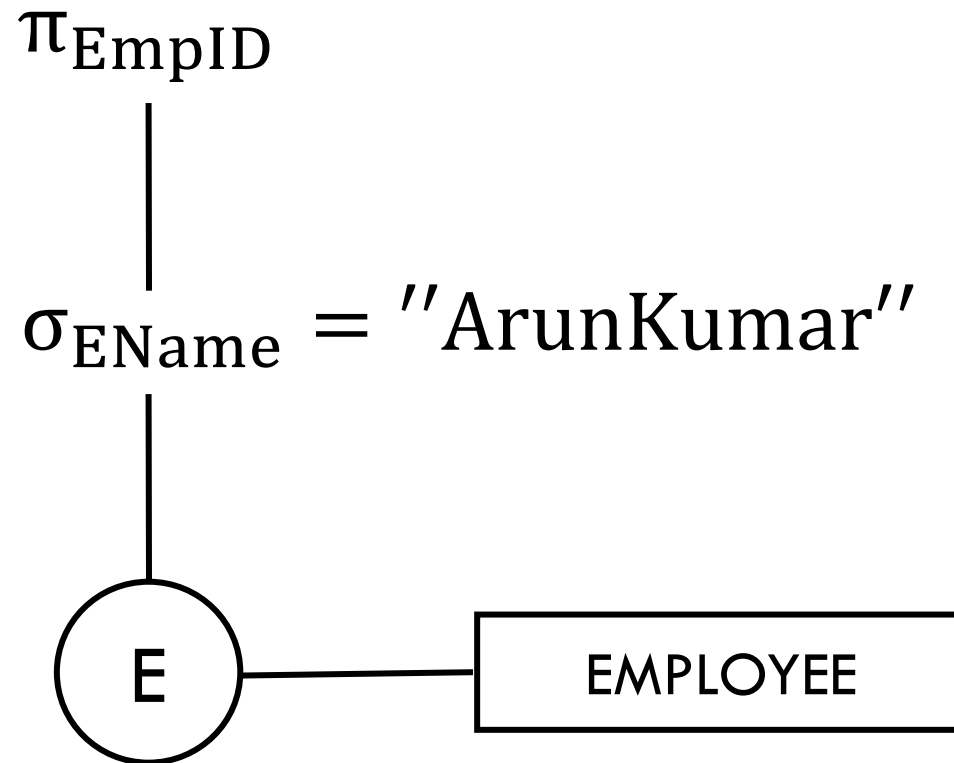
DEPARTMENT

DNO	DNAME	LOCATION
-----	-------	----------

$$\pi_{EmpID}(\sigma_{ENAME="ArunKumar"}(EMPLOYEE))$$

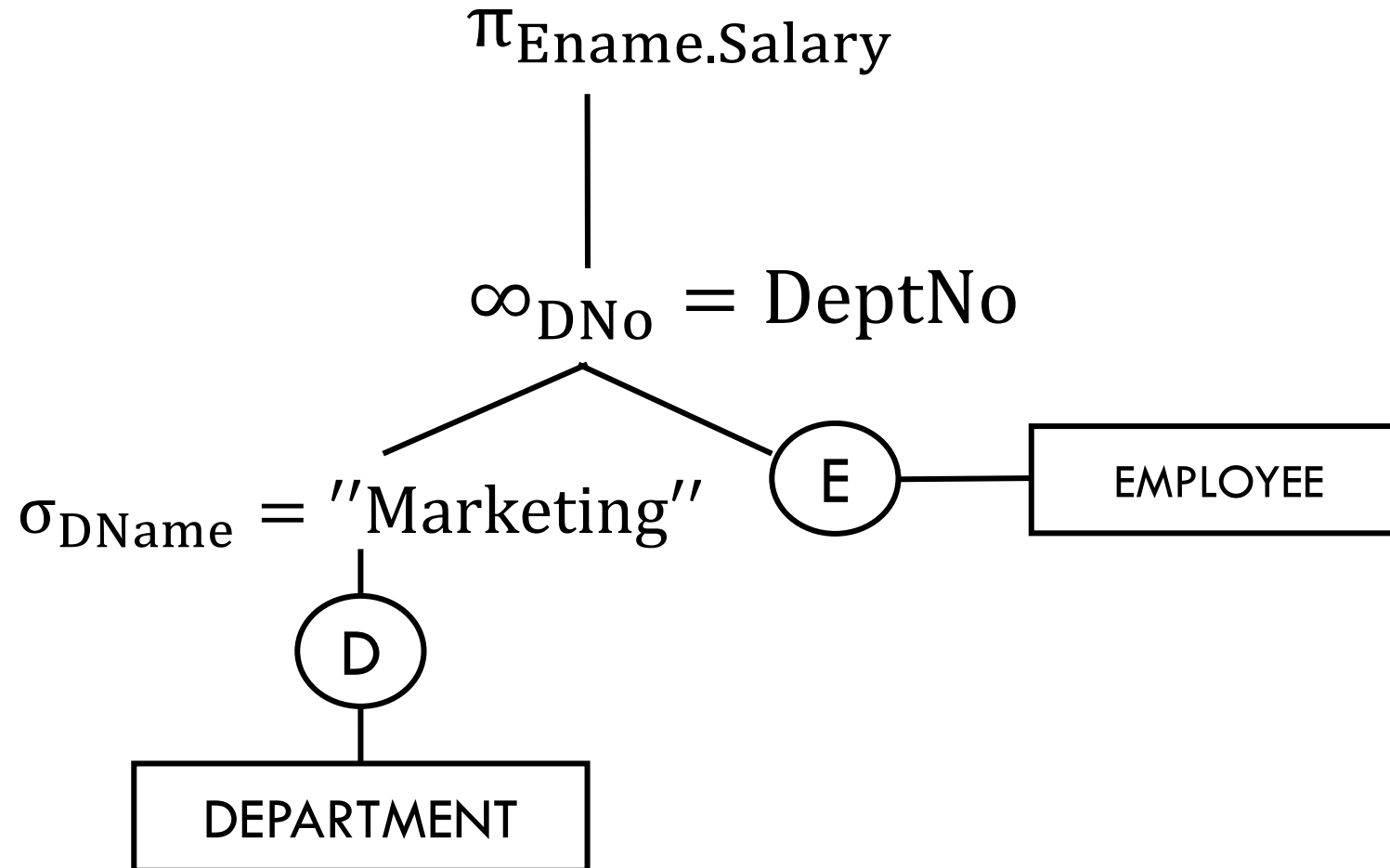
Optimisation dans une BD centralisée

$\pi_{EmpID}(\sigma_{ENAME="ArunKumar"}(EMPLOYEE))$



Optimisation dans une BD centralisée

$\pi_{ENAME, SALARY}(\sigma_{DNAME='Marketing'}(DEPARTMENT)) \bowtie_{DNO=DEPTNO} (EMPLOYEE)$



Optimisation dans une BD centralisée

- Une fois l'arborescence de requêtes générée, un plan de requête est créé
- Un plan de requête est une arborescence de requête étendue qui inclut des chemins d'accès pour toutes les opérations de l'arborescence de requête
- Les chemins d'accès spécifient comment les opérations relationnelles dans l'arborescence doivent être effectuées
- En outre, un plan de requête indique également comment les tables intermédiaires doivent être transmises d'un opérateur à l'autre, comment les tables temporaires doivent être utilisées et les opérations doivent être combinées

Optimisation dans une BD centralisée

- La génération de code est la dernière étape de l'optimisation de requêtes
- C'est la forme exécutable de la requête, dont la forme dépend du type de système d'exploitation sous-jacent
- Une fois le code de requête généré, le gestionnaire d'exécution l'exécute et génère les résultats

Requêtes

Optimisation dans une BD centralisée

- Différentes familles d'approches:
 - Recherche exhaustive
 - Recherche heuristique (approximative)

Optimisation dans une BD centralisée

- Pour une requête, tous les plans de requête possibles sont initialement générés, puis le meilleur plan est sélectionné
- Bien que cette technique constitue la meilleure solution, la complexité est exponentielle dans le temps et dans l'espace en raison de la grandeur de l'espace de solution

Optimisation dans une BD centralisée

- Les heuristiques ont une complexité temporelle et spatiale polynomiale, inférieure à la complexité exponentielle des algorithmes exhaustifs basés sur la recherche dans un espace de solutions
- Certaines des règles heuristiques communes sont -
 - Effectuez les opérations de sélection et de projection avant les opérations de jointure. Pour ce faire, déplacez les opérations `SELECT` et `PROJECT` dans l'arborescence de la requête. Cela réduit le nombre de tuples disponibles pour la jointure
 - Effectuez les opérations de sélection/projection les plus restrictives en premier avant les autres opérations
 - Évitez les opérations entre produits car elles génèrent des tables intermédiaires de très grande taille

Gestion de requêtes dans une BD distribuée