

Compte Rendu TP Base de Données

Licence 3

23/03/2020

**LE FEYER Aymeric
BAUDELET Conrad**



**Université
Polytechnique**

HAUTS-DE-FRANCE

Création des tables

Vous trouverez les ordres de creations dans le fichier creationTables.sql

Triggers

Affichage de message lors de la suppression de journee

On a un trigger after delete sur Journee qui affiche un message, rien de complexe.

Volume horaire dépassé

Le trigger est execute apres insertion dans la table Organisation

On a deux variables locales, P et V

P est le nombre d'activites de la journee ou est ajoutée la nouvelle organisation

V est la valeur a ne pas dépasser (le volume horaire quotidien)

S'il y a plus d'activites que necessaire, on affiche un message

Chaque activite dure 1 unite, c'est pour cela qu'on utilise pas le creneau horaire

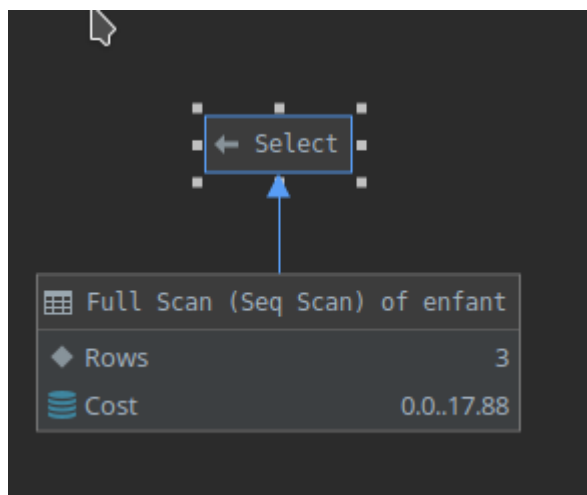
Plans d'exécution

Les requetes utilisees sont dans requetes.sql

PostgreSQL

1ere requete

| Operation | Params | Rows | Total Cost | Startup Cost | Raw desc |
|----------------------|----------------|------|------------|--------------|---|
| ↵ Select | | | | | |
| Full Scan (Seq Scan) | table: enfant; | 3 | 17.88 | 0.0 | Parallel Aware = false; Alias = enfa... |

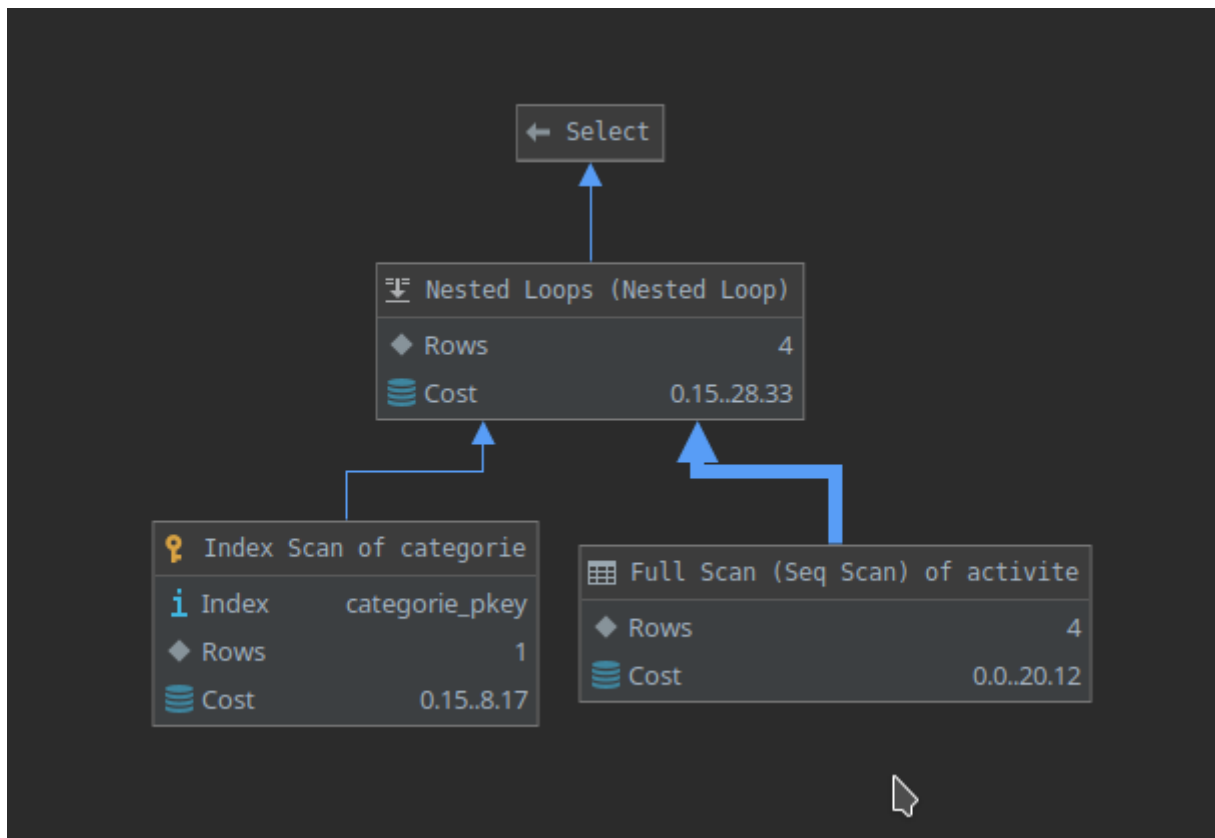


On trouve un cout de **17.88**

2eme requete

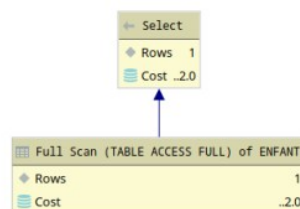
| Operation | Params | Rows | Total Cost | Startup Cost | Raw desc |
|----------------------------|--|------|------------|--------------|-----------------------------------|
| ↵ Select | | | | | |
| Nested Loops (Nested Loop) | | 4 | 28.33 | 0.15 | Parallel Aware = false; Join T... |
| Index Scan | table: categorie; index: categorie_pkey; | 1 | 8.17 | 0.15 | Parent Relationship = Oute... |
| Full Scan (Seq Scan) | table: activite; | 4 | 20.12 | 0.0 | Parent Relationship = Inner... |

Cout de **28.33**



Oracle

1ere requete

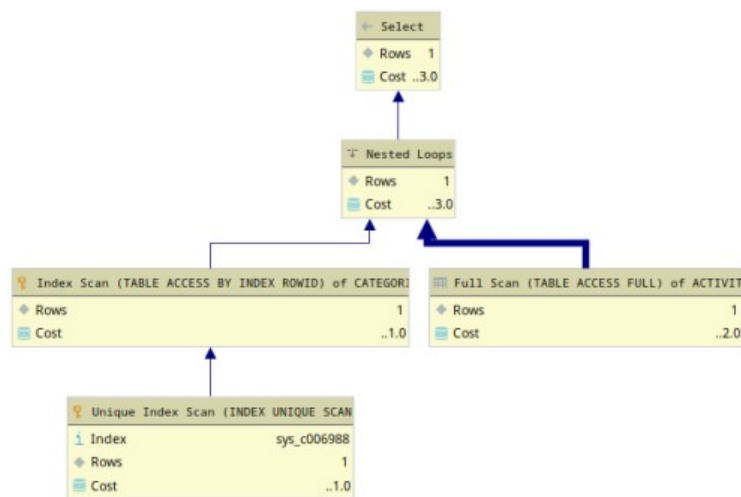


yfiles

| Operation | Params | Rows | Total Cost | Raw desc |
|-------------------------------|----------------|------|------------|------------------------------|
| Select | | 1 | 2.0 | cpu_cost = 7861, io_cost = 2 |
| Full Scan (TABLE ACCESS FULL) | table: ENFANT; | 1 | 2.0 | cpu_cost = 7861, io_cost = 2 |

Operations tree

Cout de **2.0**



yFiles

| Output Plan x SYSTEM.CATEGORIE x | | | | |
|--|---------------------|------|------------|------------------------------|
| Operation | Params | Rows | Total Cost | Raw desc |
| Select | | 1 | 3.0 | cpu_cost = 23124, io_cost... |
| Nested Loops | | 1 | 3.0 | cpu_cost = 23124, io_cost... |
| Index Scan (TABLE ACCESS BY INDEX ROWID) | table: CATEGORIE; | 1 | 1.0 | cpu_cost = 15483, io_cost... |
| Unique Index Scan (INDEX UNIQUE SCAN) | index: SYS_C006988; | 1 | 1.0 | cpu_cost = 8171, io_cost ... |
| Full Scan (TABLE ACCESS FULL) | table: ACTIVITE; | 1 | 2.0 | cpu_cost = 7641, io_cost ... |

Operations tree

On trouve un cout de **3.0**

a) PostGre et Oracle utilisent tout deux pour la premiere requete un FullScan et pour la seconde, un IndexScan en plus du FullScan dans une Nested Loop
Le fullscan est beaucoup plus lourd que l'indexscan puisque, comme son nom l'indique, le fullscan parcourt toute la table contrairement a l'index scan (qui trouve la valeur avec l'index directement)
Les couts sous Oracle ne veulent rien dire, on n'en tiendra pas compte

b) En tant qu'utilisateur on peut pour "changer" le plan 'execution': Ajouter ou supprimer un index sur la table, Mettre à jour la version d'oracle, changer les paramètres de la sessions oracle, Changer par exemple le nombre de ligne dans une table CHanger certaine pièce par exemple la vitesse d'entrée sortie , le temps de transfert entre les disques ou encore la vitesse du cpu ce qui engendre un changement des statistiques du système

On n'a malheureusement pas eu le temps de modifier les plans d'exécution comme envisagé dans le texte