

TP4 – Plans d'exécution

L'objectif de ces travaux pratiques est de réaliser quelques manipulations visant à illustrer le choix de plans d'exécution de requêtes par l'optimiseur par défaut d'Oracle, à savoir l'outil EXPLAIN PLAN.

Dans Madoc vous pouvez voir de l'information sur l'outil explain :

<http://madoc.univ-nantes.fr/mod/resource/view.php?id=142034>

Et bien sur, sur la documentation Oracle : http://docs.oracle.com/cd/B28359_01/server.111/b28274/ex_plan.htm

Les chemins d'accès fréquents :

- Parcours séquentiel (FULL TABLE sSCAN).
- Par adresse (ACCESS BY ROWID).
- Parcours de regroupement (CLUSTER SCAN). On récupère alors dans une même lecture les
- n-uplets des 2 tables du cluster.
- Recherche par hachage (HASH SCAN).
- Parcours d'index (INDEX SCAN).

Les principales opérations physiques :

- INTERSECTION : intersection de deux ensembles de n-uplet.
- CONCATENATION : union de deux ensembles.
- FILTER : élimination de n-uplets (sélection).
- PROJECTION : opération de l'algèbre relationnelle.

Opérations liées aux algorithmes de jointures d'ORACLE :

1. Boucles imbriquées quand il y a au moins un index. Opération NESTED LOOP.
2. Tri/fusion quand il n'y a pas d'index. Opération SORT et MERGE.
3. Jointure par hachage quand il n'y a pas d'index. Opération HASH JOIN

L'outil EXPLAIN PLAN donne le plan d'exécution d'une requête. La description comprend :

- Le chemin d'accès utilisé.
- Les opérations physiques (tri, fusion, intersection,...).
- L'ordre des opérations.

Il est représentable par un arbre.

La base considérée concerne un extrait de la base de IMDB, la plus grosse base de données mondiale sur les films, séries et animations. Nous avons choisi 15 années aléatoirement afin de réduire la BD pour avoir environ 25% du volume initial de la base (1874, 1890, 1910, 1918, 1936, 1951, 1968, 1973, 1985, 1992, 2002, 2008, 2015, 2020, 2029). Le schéma de la base de données se trouve à la fin de ce sujet. Les tables ont été créées avec l'utilisateur IMDBUSER. Vous avez le droit uniquement de lecture sur ces tables.

Pour afficher les stats de la base, on peut passer par les vues DBA_TAB_STATISTICS, DBA_IND_STATISTICS, DBA_TAB_COL_STATISTICS, etc.

Voici quelques requêtes exemple :

```
SELECT table_name, num_rows, blocks
FROM dba_tab_statistics
WHERE owner = 'IMDBUSER' order by table_name;
```

```
SELECT index_name, table_name, blevel, leaf_blocks, distinct_keys, num_rows,
last_analyzed
FROM dba_ind_statistics
WHERE owner = 'IMDBUSER' order by table_name, index_name;
```

```
SELECT table_name, column_name, num_distinct, density, last_analyzed,
avg_col_len
FROM DBA_TAB_COL_STATISTICS
WHERE owner = 'IMDBUSER' ORDER BY table_name, column_name;
```

Vous pouvez analyser les statistiques de vos tables (celles que vous avez crée) en remplaçant owner='IMDBUSER' par owner='L3INFOvotrenumero'.

Exercice 1. Prise en main de la base

Consultez le schéma de chacune des tables, identifiez les clés primaires, les clés étrangères et les possibilités de jointures, la volumétrie des tables, etc.

Visualisation du plan d'exécution d'une requête

Pour visualiser le plan d'exécution d'une requête, on peut utiliser le code SQL suivant (aussi bien sous SQLPlus que sous l'interface SQLDeveloper) :

- EXPLAIN PLAN FOR <ecrire ici la requête>;
- SELECT * FROM table(dbms_xplan.display);

A NOTER QUE LES REQUÊTES NE SONT PAS EXÉCUTÉES.

Sous SQLDeveloper, on peut aussi afficher le plan d'exécution avec la touche F10 ou encore avec le menu contextuel affiché avec le bouton droit de la souris.

Exercice 2. Analyse de l'évaluation de requêtes.

Q1. Exécuter et visualiser le plan des requêtes suivantes. Pour chacune noter les différentes informations fournies et comparer le plan pour R2 et celui pour R3.

- R1 : SELECT * FROM imdbuser.work_types WHERE type ='tv';
- R2 : SELECT title, type, region, language FROM imdbuser.work_types
NATURAL JOIN imdbuser.work_akas;

Les 3 requêtes suivantes sont différentes manières d'exprimer une jointure :

- R3.1 : SELECT count(*) FROM imdbuser.work_types NATURAL JOIN
imdbuser.work_akas;
- R3.2 : SELECT count(*) FROM imdbuser.work_types where (id_work,
ordering) IN (SELECT id_work, ordering FROM imdbuser.work_akas);

- R3.3 : `SELECT count(*) FROM imdbuser.work_types wt where EXISTS (SELECT * FROM imdbuser.work_akas WHERE id_work = wt.id_work and ordering = wt.ordering);`

Considérez maintenant `work_basics` qui est la table la plus volumineuse :

- R4 : `SELECT worktype, primaryTitle FROM imdbuser.work_basics where startYear > 2000;`
- R5 : `SELECT worktype, primaryTitle FROM imdbuser.work_basics NATURAL JOIN imdbuser.work_akas WHERE startYear > 2000 and language=UPPER('FR');`

On considère maintenant la table `imdbuser.work_basics1` au lieu de `imdbuser.work_basics`. Que constatez-vous dans les plans utilisés ?

Q2. Faire le bilan des index définis sur les différentes tables de la base (y compris `work_basics1`) avec leurs caractéristiques. Proposez une explication pour les résultats des requêtes R4 et R5.

Vous pouvez avoir de l'information sur les indexes en interrogeant le dictionnaire `ALL_INDEXES`.

Q3. Visualisez le plan d'exécution d'autres requêtes de votre choix.

Q4. Il est possible d'utiliser l'interface de SQL Developer pour définir un index. On peut aussi définir un index par la commande SQL simplifiée :

```
CREATE INDEX <nom_index> on <relation> (<nom_index>) ;
```

Tous les détails liés au stockage de l'index sont alors définis par défaut par Oracle. Voir : http://download.oracle.com/docs/cd/B28359_01/server.111/b28310/indexes003.htm#i1006525)

En utilisant les tables que vous avez définies dans les premiers TPs,

Visualiser le plan d'exécution d'une requête du type :

1. `SELECT * FROM avis WHERE note > 10;`
2. `SELECT * FROM avis WHERE round(note,2) > 10;`
3. Créer un index sur l'attribut `note`, puis visualiser de nouveau le plan d'exécution des requêtes précédents. Conclure.

La description des tables de ce schéma se trouve [ici](#).

