

Administration et Tuning des BD
TP Optimisation par Index et Vues Matérialisées

RELARQUE IMPORTANTE

Pour l'ensemble des TP, on demande de réaliser un compte-rendu du travail effectué. Ce compte-rendu comprendra un rapport (manuscrit ou réalisé sur traitement de texte) détaillant les points intéressants abordés à chaque question :

- requêtes SQL correspondant aux questions posées
- résultat de l'exécution de la requête
- réponses aux questions soulevées dans le sujet,
- explication des parties non triviales des requêtes complexes,
- analyse et commentaire des jeux d'essais.

1. Index

Syntaxe de création d'un index binaire sur une Table

```
CREATE BITMAP INDEX Nom_Index_bix  
ON Table (Attribut-Indexé);  
[COMPUTE STATISTICS] //mise à jour des Statistiques après création
```

Syntaxe de création d'un index de jointure binaire sur une table selon les attributs d'autres tables.

```
CREATE BITMAP INDEX Nom_Index_bjix  
ON Table (Table1.Attribut1, Table2.Attribut2, ...)  
FROM Table, Table1, Table 2, ...  
WHERE Table.FK1 = Table1.PK1 and Table.FK2 = Table1.PK2 and ....// FK : clé étrangère, PK : Clé  
primaries.  
[COMPUTE STATISTICS] //mise à jour des Statistiques après création
```

2. Vues Matérialisées

Syntaxe de création d'une vue matérialisée

```
CREATE MATERIALIZED VIEW Nom_de_la_vue //nom de la vue  
BUILD IMMEDIATE //créer immédiatement la VM  
REFRESH COMPLETE //mise à jour de la vue.  
ENABLE QUERY REWRITE //activer la réécriture de la vue  
AS Requête_Select // requête permettant de calculer la vue.
```

Remarque :

- Pour activer l'affichage du plan d'exécution, utiliser la commande : **SET AUTOTRACE ON**
- Pour vider tous les buffers partagés (**ALTER SYSTEM FLUSH SHARED_POOL**)
- Pour vider le cache de données (**ALTER SYSTEM FLUSH BUFFER_CACHE**).
- Pour forcer l'utilisation d'un index, on utilise un HINT index dans la clause select. Un hint a la forme suivante : /*+ hint(argument-1 argument-2) */.
 - Par exemple **SELECT /*+ index(PRODLEVEL Family_idx) */ from Prodlevel where FAMILY_LEVEL= 'ABDCDEF'**. Forcera Oracle à utiliser l'index Family_idx pour exécuter cette requête.

Soit les requêtes Q1, Q2 et Q3 suivantes :

Q1: SELECT sum(dollarcost) From ACTVARS A, Timelevel T Where A.TIME_LEVEL=T.TID and T.MONTHLEVEL=3	Q2 : SELECT sum(dollarcost) From ACTVARS A, PRODLEVEL P ,TIMELEVEL T Where A.TIME_LEVEL=T.TID and A.PRODUCT_LEVEL=P.CODE_LEVEL and P.CLASS_LEVEL='CI493YZ9KZUJ' and T.MONTH_LEVEL=3
Q3 : SELECT year_level, sum(dollarcost) FROM ACTVARS A,CUSTLEVEL C,PRODLEVEL P ,TIMELEVEL T WHERE A.CUSTOMER_LEVEL=C.STORE_LEVEL AND A.TIME_LEVEL=T.TID AND A.CHANNEL_LEVEL=CH.BASE_LEVEL AND A.PRODUCT_LEVEL=P.CODE_LEVEL AND P.CLASS_LEVEL='CI493YZ9KZUJ' AND C.RETAILER_LEVEL='ROJNENOU PKMQ' AND T.MONTH_LEVEL=3 AND CH.ALL_LEVEL ='EFGHIJKLMNOP' group by year_level	

Remarque : il faut changer les valeurs soulignées par des valeurs existant dans la BD.

Questions

1. Exécuter chaque requête et donner son temps d'exécution.
2. Générer et expliquer le plan d'exécution de chaque requête.
3. **Pour Q1**
 - a. Créer un **index binaire** sur la colonne Timelevel
 - b. Quel est le temps de création de l'index
 - c. Exécuter Q1, quel est son nouveau temps d'exécution.
 - d. Afficher son plan d'exécution et vérifier qu'Oracle a réellement utilisé l'index créé.
 - e. Si l'index n'a pas été utilisé, forcer son utilisation avec un Hint Index.
 - f. Créer un **index de jointure binaire** sur l'attribut Timelevel.
 - g. Répondre aux mêmes questions que b,c,d et e.
4. **Pour Q2**
 - a. Reprendre les mêmes questions en créant un index sur les deux attributs Class_level et Month_level
5. **Pour Q3**
 - a. Reprendre les mêmes questions en créant un index sur les quatre attributs Class_level, Month_level, Reatiler_level et All_level.
 - b. Créer une vue matérialisée permettant de réduire le coût de Q3.
 - i. Exécuter la requête après la création de la vue
 - ii. Donner son nouveau temps et plan d'exécution
 - c. Comparer les différentes options d'optimisation pour Q3.

Remarque : Pour activer l'affichage du temps d'exécution des requêtes dans SQL Plus, il faut exécuter la commande suivante : **SET timing ON**