

ENDO

TP4 : Optimisation par vues matérialisées

NOM : ACHACHI

Prénom : El hadj ALI

MAT : 181832008538

1/ requête R1 qui donne le nombre d'appels vers le destinataire 'VFKPXLCD' :

```
SQL> select SUM(NBAppels) as nombrea from FAppel F,DDestinataire D
  2  where F.CodeOperateurDestinataire = D.CodeOperateurDestinataire
  3  and D.NomOperateurDestinataire LIKE 'VFKPXLCD' group by D.NomOperateurDestinataire;

  NOMBREA
-----
      4511

Ecoulu : 00 :01 :08.26
```

2/ Créer une vue matérialisée VMD :

```
SQL> CREATE MATERIALIZED VIEW VMD
  2  BUILD IMMEDIATE
  3  REFRESH COMPLETE
  4  ON DEMAND
  5  ENABLE QUERY REWRITE
  6  AS
  7  SELECT D.NomOperateurDestinataire,D.CodeOperateurDestinataire,
  8  SUM(F.NBAppels) ,SUM(F.Duree) from FAppel F,DDestinataire D
  9  where F.CodeOperateurDestinataire = D.CodeOperateurDestinataire
 10  group by D.CodeOperateurDestinataire ,D.NomOperateurDestinataire;

Vue matérialisée créée.
```

3/ Ré exécuter la requête R1 :

```
SQL> alter system flush shared_pool;

Système modifié.

Ecoulu : 00 :00 :00.09
SQL> alter system flush buffer_cache;

Système modifié.

Ecoulu : 00 :00 :00.53
SQL> select SUM(NBAppels) as nombrea from FAppel F,DDestinataire D
  2  where F.CodeOperateurDestinataire = D.CodeOperateurDestinataire
  3  and D.NomOperateurDestinataire LIKE 'VFKPXLCD' group by D.NomOperateurDestinataire;

  NOMBREA
-----
      4511

Ecoulu : 00 :00 :00.98

Plan d'exécution
-----
Plan hash value: 1889684103

-----
| Id | Operation | Name | Rows | Bytes | Cost (%CPU)| Time |
|----|-----|-----|-----|-----|-----|-----|
| 0 | SELECT STATEMENT | | 1 | 30 | 2 (0)| 00:00:01 |
| 1 | HASH GROUP BY | | 1 | 30 | 2 (0)| 00:00:01 |
|* 2 | MAT_VIEW REWRITE ACCESS FULL | VMD | 1 | 30 | 2 (0)| 00:00:01 |
```

Conclusion : la vue est exploitée

4/ Créer une vue matérialisée VMNBMensuel :

```
SQL> CREATE MATERIALIZED VIEW VMNBMensuel
  2  BUILD IMMEDIATE
  3  REFRESH COMPLETE
  4  ON DEMAND
  5  ENABLE QUERY REWRITE
  6  AS
  7  select T.Mois,SUM(F.NBAppels) FROM DTemp T , Fappel F
  8  where F.CodeTemps = T.CodeTemps
  9  group by F.NBAppels , T.Mois;

Vue matérialisée créée.
```

5/ Requête R2 qui donne les nombres d'appels annuels :

```
SQL> select T.Annee,SUM(F.NBAppels) FROM DTemp T , Fappel F
  2  where F.CodeTemps = T.CodeTemps
  3  group by T.Annee;

ANNE  SUM(F.NBAPPELS)
-----
2023          673411
2021          825418
2020          829415

Ecoulé : 00 :00 :01.24
```

Conclusion : la vue n'est pas exploitée

6/ Créer les méta données de toutes les dimensions :

```
SQL> CREATE DIMENSION CLDim
  2  LEVEL CL IS (DPCClient.CodeClient)
  3  LEVEL Sexe IS (DPCClient.SexeClient)
  4  LEVEL Ville IS (DPCClient.CodeVille)
  5  LEVEL Wilaya IS (DPCClient.CodeWilaya)
  6  HIERARCHY H1 (CL CHILD OF Sexe)
  7  HIERARCHY H2 (CL CHILD OF Ville CHILD OF Wilaya)
  8  ATTRIBUTE CL DETERMINES (DPCClient.NomClient)
  9  ATTRIBUTE Ville DETERMINES (DPCClient.NomVille)
 10  ATTRIBUTE Wilaya DETERMINES (DPCClient.NomWilaya);

Dimension créée.
```

```

SQL> CREATE DIMENSION TLDim
  2  LEVEL TL IS (DTypeLigne.CodeTypeLigne)
  3  ATTRIBUTE TL DETERMINES (DTypeLigne.TypeLigne);

Dimension créée.

Ecoulé : 00 :00 :00.03
SQL> CREATE DIMENSION TADim
  2  LEVEL TA IS (DTypeAppel.CodeTypeAppel)
  3  ATTRIBUTE TA DETERMINES (DTypeAppel.TypeAppel);

Dimension créée.

Ecoulé : 00 :00 :00.01
SQL>
SQL> CREATE DIMENSION DESDim
  2  LEVEL DES IS (DDestinataire.CodeOperateurDestinataire)
  3  ATTRIBUTE DES DETERMINES (DDestinataire.NomOperateurDestinataire);

Dimension créée.

```

```

SQL> CREATE DIMENSION TimeDim
  2  LEVEL Jour IS (DTemps.CodeTemps)
  3  LEVEL Mois IS (DTemps.Mois)
  4  LEVEL Annee IS (DTemps.Annee)
  5  HIERARCHY H1 (Jour CHILD OF Mois CHILD OF Annee)
  6  ATTRIBUTE Jour DETERMINES (DTemps.LibJour);

Dimension créée.

```

7/ modifier la session à travers la commande :

```

SQL> Alter session set query_rewrite_integrity=trusted
  2  ;

Session modifiée.

```

8/ Exécuter R2 et examiner temps et plan d'exécution :

```
SQL> select T.Annee,SUM(F.NBAppels) FROM DTemp T , Fappel F
2  where F.CodeTemp = T.CodeTemp
3  group by T.Annee;
```

```
ANNE SUM(F.NBAPPELS)
-----
2023      673411
2021      825418
2020      829415
```

Ecoulé : 00 :02 :05.50

Plan d'exécution

Plan hash value: 2283652925

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		3	48	2644 (21)	00:00:32
1	HASH GROUP BY		3	48	2644 (21)	00:00:32
* 2	HASH JOIN		2330K	35M	2249 (7)	00:00:27
3	TABLE ACCESS FULL	DTEMP	731	6579	3 (0)	00:00:01
4	TABLE ACCESS FULL	FAPPEL	2330K	15M	2204 (6)	00:00:27

9/ Créer une vue matérialisée VMNBVille :

```
SQL> CREATE MATERIALIZED VIEW VMNBVille
2  BUILD IMMEDIATE
3  REFRESH COMPLETE
4  ON DEMAND
5  ENABLE QUERY REWRITE
6  AS
7  SELECT CodeVille ,NomVille, sum(NBAppels) as Nombre_Appels
8  FROM Fappel A, DPClient C
9  WHERE A.CodeClient = C.CodeClient
10 GROUP BY CodeVille,NomVille;
```

Vue matérialisée créée.

10/ Une requête R3 qui donne le nombre d'appels de chaque wilaya :

```
SQL> SELECT CodeWilaya ,NomWilaya, sum(NBAppels) as Nombre_Appels
2 FROM FAppel A, DPCCClient C
3 WHERE A.CodeClient = C.CodeClient
4 GROUP BY CodeWilaya,NomWilaya;
```

CODEWILAYA	NOMWILAYA	NOMBRE_APPELS
1	CUWCCQSG	60028
19	VJFVHMVB	43243
51	UXXOOKVG	38260
11	FIMXNZJS	29565
47	WWKIFYOW	59382
40	EEHPRCCC	42280
30	WYRTLUD	38201
57	OTKJSFZI	41834
34	VSZLHJNC	42689
24	OBOURYDD	42711
49	UMUHFVJL	25050

CODEWILAYA	NOMWILAYA	NOMBRE_APPELS
15	GWQRBWCR	42148
35	DQFUOEWQ	50957
56	DRCAOMRX	21129
4	ZXBHQNJG	63867
13	EEDOVNLP	28957
8	RYJSECNK	55749
9	RXDBAOTT	33545
21	VDAVWJLO	21558
31	YOARLYNE	42300
3	IMARPRXG	34221
50	BRQYIPQB	59325

CODEWILAYA	NOMWILAYA	NOMBRE_APPELS
46	LJCVGFTE	29756
55	RQNHBFNA	38905
36	JWWJEOKR	45699
29	XUYWTSJ	42640
14	JZJGRNYW	46371
48	PUSYSRKH	30461
41	SQIEPBWL	25583
58	YEC DOTQY	46328
12	ZZTOTKJD	50842

44	EYMOIQVZ	29981
38	HADTIAGX	38121

CODEWILAYA	NOMWILAYA	NOMBRE_APPELS
------------	-----------	---------------

6	LHABOHR	51091
54	NBHNOEXK	46796
32	TPXAQLGA	30223
10	TBSEYRXV	50913
52	GRWVUSCG	25665
45	RLYXCCIV	64739
26	DAJLESUH	42890
53	VXDMDPX	50409
42	QIOWFDNO	25527
25	ZEZJRBUO	33800
17	FVTUEILP	34647

CODEWILAYA	NOMWILAYA	NOMBRE_APPELS
------------	-----------	---------------

20	BTIUQINO	47664
18	HWDENHPG	42999
23	GVOGPSZG	21165
5	BBQAKRXZ	29826
22	STVOERXT	33546
33	VIHXLCK	38283
37	HZRWXGOT	39362
43	ZRPURCWL	42163
28	EPPNIPRS	47052
27	OWAFAMVK	46925
39	FDVFQDRE	29875

CODEWILAYA	NOMWILAYA	NOMBRE_APPELS
------------	-----------	---------------

16	XXPTYDXX	51490
7	WIGFHZPC	34283
2	JQHCRIQU	25225

58 ligne(s) sélectionné(s).

11/Le plan d'exécution du requête R3 :

```

Ecoulu : 00 :00 :01.58

Plan d'exécution
-----
Plan hash value: 345497618

-----
-----
| Id | Operation                               | Name          | Rows  | Bytes | Cost (%CP
U)| Time          |
-----
-----
|  0 | SELECT STATEMENT                       |               |  1683 | 45441 | 1921  (1
4)| 00:00:24 |
|  1 | HASH GROUP BY                           |               |  1683 | 45441 | 1921  (1
4)| 00:00:24 |
|  2 | VIEW                                    |               |  1683 | 45441 | 1921  (1
4)| 00:00:24 |
|  3 | HASH UNIQUE                             |               |  1683 | 94248 | 1921  (1
4)| 00:00:24 |
|*  4 | HASH JOIN                               |               | 1060K |  56M | 1751  (
5)| 00:00:22 |
|  5 | MAT_VIEW REWRITE ACCESS FULL| VMNBVILLE   |   547 | 20786 | 2     (
0)| 00:00:01 |
|  6 | TABLE ACCESS FULL                    | DPCCLIENT    | 1060K |  18M | 1729  (
4)| 00:00:21 |
-----
-----
```


12 / Supprimer les méta données de la dimension DClient et réexécuter la requête R3:

```
SQL> DROP DIMENSION CLDim;

Dimension supprimée.

Ecoulé : 00 :00 :02.06
SQL> SELECT CodeWilaya ,NomWilaya, sum(NBAppels) as Nombre_Appels
  2  FROM FAppel A, DPCClient C
  3  WHERE A.CodeClient = C.CodeClient
  4  GROUP BY CodeWilaya,NomWilaya;
```

CODEWILAYA	NOMWILAYA	NOMBRE_APPELS
40	EEHPRCCC	42280
30	WYRTLUD	38201
47	WWKIFYOW	59382
19	VJFVHMB	43243
51	UXXOOKVG	38260
1	CUWCCQSG	60028
11	FIMXNZJS	29565
57	OTKJSFZI	41834
49	UMUHFVJL	25050
24	OBOURYDD	42711
34	VSZLHJNC	42689

CODEWILAYA	NOMWILAYA	NOMBRE_APPELS
8	RYJSECNK	55749
21	VDAVWJLO	21558
9	RXDBAOTT	33545
56	DRCAOMRX	21129
4	ZXBHQNJG	63867

58 ligne(s) sélectionnée(s).

Ecoulé : 00 :00 :03.40

Plan d'exécution

Plan hash value: 3020736707

Id	Operation	Name	Rows	Bytes	TempSpc	Cost (%CPU)	T
0	SELECT STATEMENT		2379	68991		8364 (9)	0
1	HASH GROUP BY		2379	68991		8364 (9)	0
* 2	HASH JOIN		2330K	64M	32M	7969 (4)	0
3	TABLE ACCESS FULL	DPCCLIENT	1060K	20M		1729 (4)	0
4	TABLE ACCESS FULL	FAPPEL	2330K	20M		2204 (6)	0

Comparaison : le temps d'exécution avant la suppression du dimensionnement est moins qu'après.

13/ Conclusions :

ENABLE QUERY REWRITE est important

Les Vues matérialisées permettent d'optimiser le temps d'exécution à condition de leur "faire confiance"

14/ L'intérêt de la commande : `Alter session set query_rewrite_integrity=trusted :`

In TRUSTED mode, the optimizer trusts that the relationships declared in dimensions and RELY constraints are correct.

In this mode, the optimizer also uses prebuilt materialized views or materialized views based on views,

and it uses relationships that are not enforced as well as those that are enforced. It also trusts declared but not ENABLED VALIDATED primary

or unique key constraints and data relationships specified using dimensions.

This mode offers greater query rewrite capabilities but also creates the risk

of incorrect results if any of the trusted relationships you have declared are incorrect.