



Université des sciences et de la Technologie  
Houari Boumediene  
USTHB – Alger

Département d'Informatique

**ADMINISTRATION ET TUNING DE BASES  
DE DONNÉES**

**RESPONSABLE  
DR K. BOUKHALFA**

**TUNING DES BASES  
DE DONNÉES**

## OPTIMISATION ET TUNING

L'optimisation ou « tuning » consiste à établir le meilleur niveau de performances d'un système en tenant compte de contraintes souvent contradictoires

### Optimisation

- Grande étendue
- Remise en cause de l'optimisation existante
- Nécessite la maîtrise de plusieurs paramètres
- Deux options
  - Utilisation d'algorithmes heuristiques ou méta-heuristiques pour trouver la meilleure configuration
  - Utilisation des Advisor comme Oracle Database SQL Access Advisor

## OPTIMISATION ET TUNING

### Tuning

- Petite étendue (un paramètre, une requête, un seul index)
- Modifications légères de la configuration courante.
- Touche un minimum de paramètres
- Deux options
  - Manuellement : selon les connaissances de l'administrateur
  - Utilisation des Advisor comme Oracle Database Tuning Advisor

## TUNING

Le tuning doit faire partie intégrante de la conception d'une application.

Trop souvent, on ne se soucie d'optimisation que lorsque

- L'application ne fonctionne pas
- Les utilisateurs sont mécontents
- La situation est explosive !

Pourtant, plus de 90 % de l'optimisation possible provient de l'application et du design de la base de données.

Malheureusement, il n'existe pas le paramètre **PERFORMANCE=MAXI** dans les paramètres d'initialisation d'une base.

## TUNING

### Deux niveaux de tuning

#### 1. Système

#### 2. Base de données

## TUNING DU SYSTEME

**Le tuning du système a une incidence sur la performance du serveur de BD**

**Il ne faut pas surcharger le système par des tâches Windows annexes :**

- ❑ Serveur de fichiers et d'imprimantes ;
- ❑ Serveur Active Directory ou contrôleur de domaine ;
- ❑ Passerelle d'accès distant ;
- ❑ Serveur DNS, etc.

## LE SUPPORT DES LARGES PAGES

L'utilisation par Oracle (depuis la version 10g) des *larges pages sur un serveur Windows économise le CPU* et tire le meilleur parti des ressources mémoires :

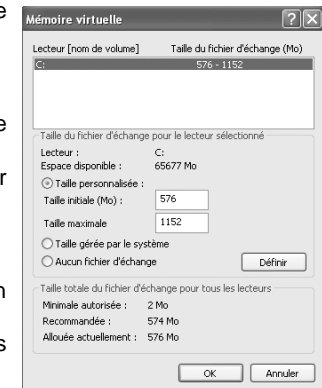
- ❑ Au lieu de manipuler des portions mémoire de **4 Ko**, l'option large page permet de les passer à **4 Mo** par portion.
- ❑ Cette option est particulièrement utile si la taille des données en SGA est de plusieurs giga-octets.
- ❑ Pour de petites configurations, le gain sera négligeable, voire négatif.

## **RÉDUIRE LA PRIORITÉ DES APPLICATIONS DE PREMIER PLAN**

- ❑ Donner un maximum de temps de traitement de Windows pour les tâches de fond, c'est-à-dire la base de données.
- ❑ Démarrer > Paramètres > Panneau de configuration > Système.
- ❑ Affecter un maximum de temps processeur et de ressource mémoire aux applications fonctionnant en arrière-plan.
- ❑ Définir la taille du fichier d'échange du système (Swap).

## **FICHER DE PAGINATION (SWAP)**

- ❑ On peut créer un ou plusieurs fichiers de pagination.
- ❑ La taille de l'ensemble des fichiers de pagination doit être suffisante pour contenir tous les « débordements » du système.
- ❑ Une taille trop grande ne présente aucun inconvénient : prévoyez large, de 4 à 6 fois la taille de la mémoire vive.



## ***SUPPRIMER LES SERVICES NON INDISPENSABLES***

De nombreux services sont lancés par Windows et Oracle.

Annulez tous ceux qui ne sont pas indispensables à votre configuration.

Certains services peuvent être stoppés :

- Plug and Play ;
- Gestionnaire de connexions d'accès distant ;
- Gestionnaire de numérotation automatique d'accès distant ;
- Serveur d'accès distant ;
- Service Téléphonie ;
- ... et tous ceux qui ne sont ni utiles, ni utilisés dans votre contexte.

## ***SUPPRIMER LES SERVICES NON INDISPENSABLES***

D'autres ne doivent pas être stoppés :

- Enregistrement d'événements ;
- Messagerie ;
- Service d'appels RPC ;
- Serveur ;
- Planning ;
- Aide TCP/IP NetBIOS ;
- Système d'exploitation ;
- Station de travail ;
- OracleServiceSID ;
- OracleOraHome92TNSListener80.

## ***SUPPRIMER LES PROTOCOLES INUTILES***

**Différents protocoles sont peut-être installés sur le serveur et certains peuvent être supprimés.**

- ❑ Laisser TCP/IP
- ❑ Certains protocoles, comme SMB (, peuvent être supprimés d'un serveur Oracle. SMB (Server Message Block2) est un protocole permettant le partage de ressources (fichiers et imprimantes) sur des réseaux locaux avec des PC sous Windows.
- ❑ Si plusieurs protocoles sont utilisés sur le serveur, il faut leur donner un ordre qui met en tête le protocole le plus utilisé par votre base Oracle.
- ❑ Ce protocole est majoritairement TCP/IP.

## ***UTILISEZ LE DERNIER SERVICE PACK DE WINDOWS***

Microsoft améliore Windows en sortant régulièrement des correctifs ou Service Pack. Ils comprennent des améliorations touchant les performances, la sécurité, des corrections de bugs, etc.

Ils sont numérotés et doivent être lancés dans l'ordre de leur parution : ServicePack 1, puis 2, puis 3...

## LES OUTILS DISPONIBLES SOUS WINDOWS

### *Le gestionnaire des tâches Windows*

Le gestionnaire des tâches visualise la quantité de mémoire, physique et logique, consommée par le système.

Lorsque l'utilisation de la mémoire s'approche de sa limite maximale, les performances sont dégradées et on risque un blocage du système : il est urgent d'agir !

On peut:

- arrêter des programmes consommateurs ;
- augmenter la taille du fichier de pagination ;
- augmenter la mémoire du système.

## L'ANALYSEUR DE PERFORMANCES WINDOWS

Le suivi des performances entrées/sorties disque nécessite un paramétrage de Windows.

Les compteurs de performance ne sont pas disponibles et il faut les installer.

L'installation des compteurs s'effectue en lançant diskperf.exe :

- `C:\diskperf.exe -Y`

Les indicateurs disponibles pour les disques logique et physique sont :

- Longueur moyenne de la file d'attente (pourcentage moyen de l'attente en E/S) ;
- Longueur actuelle de la file d'attente (vue instantanée de la file d'attente en E/S) ;
- Nombre de bytes par transfert (volume de données échangées par E/S).



## DISKPERF

`diskperf [-Y[D|V] | -N[D|V]] [\computername]`

Option	Description
-?	Displays context sensitive help.
-Y	Start all disk performance counters when the computer restarts.
-YD	Enable disk performance counters for physical drives when the computer restarts.
-YV	Enable disk performance counters for logical drives or storage volumes when the computer restarts.
-N	Disable all disk performance counters when the computer restarts.
-ND	Disable disk performance counters for physical drives when the computer restarts.
-NV	Disable disk performance counters for logical drives or storage volumes when the computer restarts.
\\<computername>	Specify the name of the computer where you want to enable or disable disk performance counters.

## L'ANALYSEUR DE PERFORMANCES WINDOWS POUR ORACLE

Par défaut, tous les compteurs proposés ne concernent que Windows.

Oracle a donc ajouté ses propres compteurs relatifs à la surveillance d'une base.

Une fois installés, il faut préciser quelle base il convient d'étudier.

L'utilitaire **operfcfg.exe** inscrit les informations conservées en :

HKEY\_LOCAL\_MACHINE\SYSTEM\Current-ControlSet\  
SERVICES\ORACLE9\Performance.

Usage: `OPerfCfg [-U <username>] [-P <password>] [-D [database name]]`

`C:\> operfcfg -U system -P mot_de_passe -D PROD`

## **ACCÉDER AUX COMPTEURS DE PRRFORMANCE ORACLE**

Start > Programs > Oracle - HOME\_NAME > Configuration and  
Migration Tools > Oracle Counters for Windows Performance Monitor

Trois visualisations possibles

1. Graphiques
2. Ensembles d'alerts
3. Fichiers logs
4. Rapports de performance

## ***FERMER TOUTES LES APPLICATIONS INUTILES***

**Comme pour les services, il faut fermer toutes les applications  
inutiles qui fonctionnent sur le serveur :**

- Les applications démarrées au lancement de la machine  
(comme les barres d'outil de Microsoft Office).
  - Ce sont des applications très consommatrices qui n'ont pas leur  
place sur un serveur base de données.
- Les économiseurs d'écran : On les oublie souvent, mais ce sont  
de véritables catastrophes en terme de consommation  
processeur et mémoire.

# LES ÉTAPES DU TUNING DE LA BD

## TUNING DE LA BD

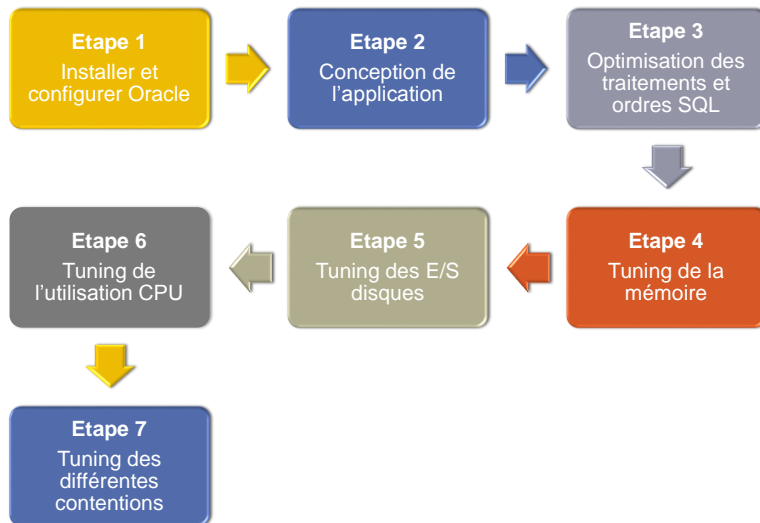
### Souvent, le tuning est oublié durant la conception de l'application

- Les applications sont testées sur de petites configurations avec peu de données.
- Elles sont ensuite déployées et le volume de données n'a plus rien en commun avec ce qu'il était initialement.
- Des problèmes, jusque-là masqués, apparaissent.

### → Nécessité des actions de tuning.

- On peut améliorer de façon spectaculaire les performances en corrigeant des erreurs courantes provenant de l'application ou du paramétrage.
- Le Tuning se fait en plusieurs étapes.

## ETAPES DU TUNING



## ETAPES DU TUNING

### ❑ Étape 1 : Installer et configurer Oracle 10g

- ❑ installer et paramétrer Oracle 10g en utilisant les règles *Optimal Flexible Architecture* (OFA) : nommage des fichiers, répertoires, emplacements, etc.

### ❑ Étape 2 : Design de l'application

- ❑ choix des paramètres initiaux de la base ;
- ❑ schéma physique de la base ;
- ❑ techniques de développement des applications

### ❑ Étape 3 : Optimisation des traitements et ordres SQL

- ❑ mise à jour des statistiques de la base ;
- ❑ tuning des ordres SQL avec TKPROF, EXPLAIN PLAN et des outils d'Oracle Enterprise Manager.

## ETAPES DU TUNING

### Étape 4 : Tuning de la mémoire

- ❑ mémoire Oracle : tuning du nombre de buffers de données, buffers redo-log et d'autres paramètres de la System Global Area (SGA) ;
- ❑ tuning de la mémoire Windows en réduisant la pagination.

### Étape 5 : Tuning des entrées/sorties disques

- ❑ répartir les entrées/sorties et les applications.

### Étape 6 : Tuning de l'utilisation CPU

- ❑ répartition des applications ;
- ❑ réorganisation des processus.

### Étape 7 : Tuning des différentes contentions

- ❑ vision très fine des paramètres d'optimisation d'Oracle.

## ORDRE DES ETAPES

- ❑ Il est important de suivre le déroulement logique de ces étapes car des changements effectués sur l'une d'elles influent sur les suivantes.
  - ❑ Par exemple, la modification d'un ordre SQL dans l'étape 3 influera sur les ressources mémoire, CPU et disque consommées, c'est-à-dire les étapes 4, 5 et 6.
- ❑ Face à une situation où l'on constate de très mauvaises performances, la bonne approche consiste à rechercher le point de contention pour le supprimer.
- ❑ On peut commencer un tuning par étape, une fois ce point corrigé, car il masque toutes les autres caractéristiques.
- ❑ Certaines actions peuvent dégrader des points visualisés dans les étapes précédentes.
- ❑ C'est pourquoi le tuning doit être effectué en « boucle », en contrôlant la stabilité du résultat obtenu dans les précédentes étapes.

## IDENTIFICATION DES CONTENTIONS

Avant d'effectuer le tuning d'un système, il est important de connaître les différentes

**contentions** qui peuvent survenir.

- ❑ Un système est lent à cause d'un goulot d'étranglement qui limite les performances de l'ensemble.
- ❑ Les trois ressources systèmes où peut apparaître une contention sont :
  - ❑ la mémoire, les systèmes d'entrée/sortie disque et réseau et enfin le ou les processeurs.
- ❑ **La principale cause de contention : Manque de mémoire**
  - ❑ → La pagination (le transfert sur disque de pages mémoire).
  - ❑ Il est important de surveiller ce point qui peut, à terme, fortement dégrader les performances du système.
  - ❑ Des ordres SQL permettent de surveiller la contention mémoire côté Oracle 10g (*Enterprise Manager* et des utilitaires graphiques côté Windows (*l'analyseur de performance Windows*)).

## ÉTAPE 1 : INSTALLER ET CONFIGURER ORACLE 10G

Une installation Windows soignée est bénéfique car elle limite l'apparition de goulots d'étranglement.

- ❑ **Choisir des disques durs**
  - ❑ La base Oracle 10g est conçue pour utiliser massivement la mémoire, beaucoup plus rapide d'accès que les disques
- ❑ **Préparer une installation respectant les règles OFA**
  - ❑ L'Optimal Flexible Architecture (OFA) définit un ensemble de règles à respecter pour faciliter le travail d'administrateur.
- ❑ **L'installation d'Oracle 10g sous Windows.**
  - ❑ Le `DB_BLOCK_SIZE` d'Oracle (taille de block)
    - ❑ Il y a de gros gains de performance à attendre de ce paramètre : 2 K, 4 K, 8 K ou 16 K
    - ❑ `Select name, value from v$parameter where name='DB_BLOCK_SIZE' ;`

## ***DB\_BLOCK\_SIZE***

- ❑ **2 Ko et 4 Ko** : pour les applications réalisant beaucoup de transactions, qui accèdent et manipulent peu de données à la fois ;
- ❑ **8 Ko** : valeur intermédiaire, correcte pour les applications transactionnelles et des requêtes balayant beaucoup de blocs de données sur disque ;
- ❑ **16 Ko** : pour les applications décisionnelles : infocentre, data warehouse, qui accèdent toutes à de gros volumes de données.

## **ÉTAPE 2 : DESIGN DE L'APPLICATION**

**Le design de l'application est primordial pour bénéficier de bonnes performances.**

### ***Identifier les points sensibles***

- ❑ Tant que le volume de données est faible, il y a peu de risques que des problèmes apparaissent.
- ❑ Des tables plus volumineuses révèlent des problèmes jusque-là masqués.
- ❑ Les points sensibles concernent principalement un schéma conceptuel mal élaboré : nombre très important de jointures, des index oubliés et des ordres SQL mal construits.

### ***Des index oubliés***

- ❑ Il faut s'assurer que les clés primaires des tables possèdent toutes un index.
- ❑ Si l'application interroge les tables sur des colonnes non indexées, il faut étudier la possibilité de créer des index sur ces colonnes.

## ÉTAPE 3 : OPTIMISATION DES TRAITEMENTS ET ORDRES SQL

### Comment détecter les problèmes ?

- ❑ mauvaises performances ou une dégradation progressive des temps de réponse au fur et à mesure de l'évolution de la base de données.
- ❑ Pour chaque application, il est assez facile de sélectionner quelques requêtes à surveiller.
- ❑ Il faut construire un jeu de test lancé régulièrement pour lequel vous enregistrerez les résultats mesurés. Cela vous permettra de disposer de courbes de tendance et d'effectuer des actions préventives.

### L'organisation physique de la base de données

- ❑ Quelques points importants à vérifier : la taille adéquate du DB\_BLOCK\_SIZE, l'absence de fragmentation et de chaînage pour les tables et les index et enfin la pertinence des statistiques mises à la disposition de l'optimiseur d'Oracle .

## DISPOSER DE STATISTIQUES À JOUR

- ❑ Lorsque Oracle reçoit une requête SQL, il décide, en interne, comment la traiter de manière optimale.
- ❑ C'est le rôle de la collecte de statistiques qui est effectuée automatiquement par Oracle pour tous les schémas.
- ❑ Elle s'effectue pendant la construction ou la reconstruction d'un index ou manuellement à l'aide du programme GATHER\_STATS\_PROG ou du job GATHER\_STATS\_JOB.
- ❑ Le job GATHER\_STATS\_JOB exécute la procédure DBMS\_STATS.GATHER\_DATABASE\_STATS\_JOB\_PROC qui collecte toutes les statistiques.



## **LES FICHIERS REDO-LOG**

- ❑ De nombreux sites fonctionnent avec des fichiers redo-log sous-dimensionnés.
- ❑ Cela crée un point de contention et la base se met en attente d'écriture dans les redo-log.
- ❑ L'utilisation des vues V\$LOGHIST et V\$LOG\_HISTORY indique quand un fichier redolog est plein et qu'il bascule vers un autre. La durée conseillée est de 20 minutes minimum.
- ❑ La taille optimale peut être consultée en interrogeant la colonne OPTIMAL\_LOGFILE\_SIZE de la vue V\$INSTANCE\_RECOVERY.
- ❑ Si la base auditée montre une fréquence plus rapprochée, il faut augmenter la taille des fichiers redo-log.

## **L'OPTIMISATION DES ORDRES SQL : UTILISATION DE SQL\*PLUS**

- ❑ Face à une requête SQL longue, que faire ?
- ❑ Le plus simple est de vérifier la manière dont le moteur d'Oracle *la traite*.
- ❑ *Ce plan d'exécution vous renseigne totalement.*
- ❑ // permet d'observer les index utilisés (ou non utilisés), l'ordre de balayage des tables, etc.
- ❑ C'est le point de départ de tout tuning applicatif sérieux.
- ❑ Le plan d'exécution des ordres SQL peut être visualisé graphiquement à l'aide d'Oracle Enterprise Manager ou d'une façon plus simple avec SQL\*Plus.

## **SQL TUNING ADVISOR**

- ❑ Cette nouveauté d'Oracle 10g *automatise les opérations de tuning d'ordres SQL.*
- ❑ *Elle est accessible en ligne, par des procédures et des fonctions ou une procédure PL/SQL.*
- ❑ On peut y accéder à partir d'Oracle Enterprise Manager pour examiner un ordre SQL.

## **LE PLAN D'EXÉCUTION DES ORDRES SQL AVEC SQL \*PLUS**

- ❑ Bien que vieillissant et offrant moins de souplesse que l'utilitaire SQL\*Plus Worksheet d'Oracle Enterprise Manager, SQL\*Plus dispose d'une option très intéressante : vous pouvez visualiser le plan d'exécution d'une requête SQL en interactif.
- ❑ Ce sont les mêmes statistiques que nous pouvons observer dans Oracle Enterprise Manager.
- ❑ Auparavant, ceci se faisait par l'utilitaire TKPROF, qui existe toujours, mais qui est d'un usage plus contraignant.

## PRÉPARATION DE L'ENVIRONNEMENT

1. Se connecter sous l'utilisateur SYS, lancer le fichier suivant qui crée le rôle PLUSTRACE
  1. SQL> start c:\oracle\ora10\sqlplus\admin\plustrce.sql
2. Toujours sous le même utilisateur SYS, affecter le rôle aux utilisateurs qui analyseront les plans d'exécution des requêtes.
  - ❑ Par exemple, pour l'utilisateur TABD : SQL> grant PLUSTRACE to TABD
3. Enfin, créer la table PLAN\_TABLE qui sert à stocker temporairement le plan d'exécution des requêtes (*utlxplan.sql livré avec Oracle*).
4. SQL> connect TABD
5. SQL> start C:\oracle\product\10.1.0\db\_1\RDBMS\ADMIN\utlxplan.sql

## EXEMPLE

L'exemple suivant met en œuvre l'utilisation de l'autotrace :

```
SQL> spool fichier_spool.lis
SQL> connect scott
SQL> set AUTOTRACE TRACEONLY EXPLAIN STATISTICS
SQL> select ename, job, dname from emp, dept
where emp.deptno = dept.deptno order by sal;
14 ligne(s) sélectionnée(s).
Plan d'exécution
-----
0 SELECT STATEMENT Optimizer=CHOOSE
1 0 SORT (ORDER BY)
2 1 NESTED LOOPS
3 2 TABLE ACCESS (FULL) OF 'EMP'
4 2 TABLE ACCESS (BY INDEX ROWID) OF 'DEPT'
5 4 INDEX (UNIQUE SCAN) OF 'PK_DEPT' (UNIQUE)
Statistiques
```

```
-----
349 recursive calls
5 db block gets
94 consistent gets
22 physical reads
0 redo size
1169 bytes sent via SQL*Net to client
709 bytes received via SQL*Net from client
4 SQL*Net roundtrips to/from client
8 sorts (memory)
0 sorts (disk)
14 rows processed
```

## EXPLAIN PLAN - EXEMPLES

Rows Execution Plan

```

-----
      12 SORT AGGREGATE
      2 SORT GROUP BY
 76563 NESTED LOOPS
 76575 NESTED LOOPS
      19 TABLE ACCESS FULL CN_PAYRUNS_ALL
 76570 TABLE ACCESS BY INDEX ROWID CN_POSTING_DETAILS_ALL
 76570 INDEX RANGE SCAN (object id 178321)
 76563 TABLE ACCESS BY INDEX ROWID CN_PAYMENT_WORKSHEETS_ALL
11432983 INDEX RANGE SCAN (object id 186024)

```

## EXEMPLE AVEC FRAGMENTATION

```

CREATE TABLE emp_range
PARTITION BY RANGE(hire_date)
(
PARTITION emp_p1 VALUES LESS THAN (TO_DATE('1-JAN-1991','DD-MON-YYYY')),
PARTITION emp_p2 VALUES LESS THAN (TO_DATE('1-JAN-1993','DD-MON-YYYY')),
PARTITION emp_p3 VALUES LESS THAN (TO_DATE('1-JAN-1995','DD-MON-YYYY')),
PARTITION emp_p4 VALUES LESS THAN (TO_DATE('1-JAN-1997','DD-MON-YYYY')),
PARTITION emp_p5 VALUES LESS THAN (TO_DATE('1-JAN-1999','DD-MON-YYYY'))
)

```

```
SELECT * FROM emp_range;
```

Operation	Name	Rows	Bytes	Cost	Pstart	Pstop
SELECT STATEMENT		105	8K	1		
PARTITION RANGE ALL					1	5
TABLE ACCESS FULL	EMP_RANGE	105	8K	1	1	5

## EXEMPLE AVEC FRAGMENTATION

```
CREATE TABLE emp_range
PARTITION BY RANGE(hire_date)
(
PARTITION emp_p1 VALUES LESS THAN (TO_DATE('1-JAN-1991','DD-MON-YYYY')),
PARTITION emp_p2 VALUES LESS THAN (TO_DATE('1-JAN-1993','DD-MON-YYYY')),
PARTITION emp_p3 VALUES LESS THAN (TO_DATE('1-JAN-1995','DD-MON-YYYY')),
PARTITION emp_p4 VALUES LESS THAN (TO_DATE('1-JAN-1997','DD-MON-YYYY')),
PARTITION emp_p5 VALUES LESS THAN (TO_DATE('1-JAN-1999','DD-MON-YYYY'))
)
```

```
SELECT * FROM emp_range
WHERE hire_date >= TO_DATE('1-JAN-1995','DD-MON-YYYY');
```

Operation	Name	Rows	Bytes	Cost	Pstart	Pstop
SELECT STATEMENT		3	54	1		
PARTITION RANGE ITERATOR					4	5
TABLE ACCESS FULL	EMP_RANGE	3	54	1	4	5

## ÉTAPE 4 : TUNING DE LA MÉMOIRE

- ❑ L'utilisation de mémoire est synonyme de performance.
- ❑ Les données accédées et manipulées en mémoire le sont beaucoup plus rapidement que sur disque.
- ❑ La zone mémoire qu'Oracle *utilise pour résoudre les requêtes SQL est paramétrable*.
- ❑ Toute l'astuce consiste à lui allouer autant de mémoire que possible, sans solliciter la zone de pagination qui intervient lorsque la capacité mémoire est dépassée et force Windows à paginer.
- ❑ Il faudra donc ajuster les paramètres en observant constamment l'évolution globale du système, au moyen de commandes Windows comme pfmom.exe ou d'utilitaires graphiques comme le gestionnaire de tâches ou l'analyseur de performances.

## **L'OPTIMISATION DE LA MÉMOIRE RÉSERVÉE POUR ORACLE**

- ❑ Les paramètres d'allocation mémoire pour Oracle 10g sont *modifiables dynamiquement*.
- ❑ Lors du démarrage d'une instance, ils sont lus dans un fichier d'initialisation.

La commande suivante, permet de connaître la taille mémoire allouée à l'instance :

- ❑ SQL> connect system
- ❑ SQL> select \* from v\$sga ;

## **PARAMÈTRES MÉMOIRES**

Les paramètres les plus importants, influant sur la zone mémoire réservée par chaque instance Oracle, sont :

- ❑ **SGA\_TARGET** spécifie la taille totale devant être allouée à la SGA. Si sa valeur est à 0, le tuning automatique des zones de la SGA est désactivé. Si une valeur lui est affectée, la taille des zones mémoire suivantes est automatiquement adaptée pour se répartir dynamiquement la taille définie par SGA\_TARGET :
- ❑ **DB\_CACHE\_SIZE** : Ce paramètre définit la zone mémoire utilisée pour conserver les données des tables et des index accédés. Plus cette zone est grande, plus petite est la probabilité d'accéder au disque à la suite d'un ordre SELECT. C'est lui qui a le plus d'impact sur la taille totale de la SGA.
- ❑ **SHARED\_POOL\_SIZE** : Cette zone mémoire est utilisée pour conserver les plans d'exécution des requêtes ainsi que les procédures PL/SQL traitées par le serveur. Elle peut être diminuée si vous sollicitez peu le PL/SQL, car c'est lui qui occupe le plus d'espace.

## PARAMÈTRES MÉMOIRES

- ❑ **JAVA\_POOL\_SIZE** : Cette zone mémoire est utilisée pour exécuter les applications Java au coeur de la base Oracle. Si vous n'utilisez pas cette option, ne lui allouez aucune mémoire.
- ❑ **DB\_BLOCK\_SIZE** : taille d'un bloc de données.
- ❑ **SORT\_AREA\_SIZE** : Lorsque le volume des données à trier est faible, Oracle 10g réalise l'opération dans une zone mémoire allouée au processus qui a demandé le tri.
- ❑ **LOG\_BUFFER** : Alloue une zone qui conserve les données à insérer dans les fichiers redo-log. Augmenter cette zone permet de réduire les entrées/sorties des fichiers log.

## ÉTAPE 5 : TUNING DES ENTRÉES/SORTIES DISQUE

**Le temps de lecture moyen d'un bloc de données se situe généralement entre 5 et 20 millisecondes suivant le système et la charge.**

**Si la durée est plus longue, on est en présence d'une saturation disque.**

**Deux options :**

- ❑ diminuer la charge en travaillant sur la base, les applications, etc. ;
- ❑ renvoyer la configuration des disques et la répartition des données sur différents axes.

## ÉTAPE 6 : TUNING DE L'UTILISATION CPU

- ❑ Il y a peu de possibilités pour optimiser la puissance processeur
- ❑ Le changer pour en utiliser un autre plus puissant
- ❑ Augmenter le nombre si la carte mère l'accepte
- ❑ Ou tout simplement diminuer les ressources consommées, c'est-à-dire la charge de la machine.

## ÉTAPE 6 : TUNING DE L'UTILISATION CPU

### ***Fermer toutes les applications inutiles***

- ❑ Fermer toutes les applications inutiles qui fonctionnent sur le serveur.

### ***Les applications démarrées au lancement de la machine***

- ❑ Faire le tri et ne conserver que celles qui demeurent nécessaires.

### **Les économiseurs d'écran**

- ❑ On les oublie souvent, mais ce sont de véritables catastrophes en terme de consommation processeur et mémoire.
- ❑ Choisir l'économiseur le plus simple possible



## ÉTAPE 7 : TUNING DES DIFFÉRENTES CONTENTIONS

- ❑ Oracle 10g propose de nouveaux outils pour réaliser un tuning fin de la base de données.
- ❑ Le plus récent, basé sur **Automatic Workload Repository (AWR)**, est l'**Automatic Database Diagnostic Monitor (ADDM)**, les plus anciens sont **tkprof** et **utlbstat/utlestat**.
- ❑ ADDM exploite les statistiques de tous ordres collectés par AWR.
- ❑ Comme AWR conserve l'historique des événements, ADDM peut être utilisé *a posteriori* pour éviter qu'une baisse de performance constatée ne se reproduise.

## ***AUTOMATIC WORKLOAD REPOSITORY (AWR)***

- ❑ AWR collecte des informations à une fréquence déterminée.
- ❑ Par défaut, la collecte s'effectue une fois par heure et la période de rétention des informations collectées est de 7 jours.
- ❑ Les statistiques collectées lors de chaque cliché concernent l'accès aux objets, l'activité et la durée des sessions, les ordres SQL les plus consommateurs de ressources (CPU, mémoire disque).
- ❑ Ces informations nécessitent de l'espace de stockage. Avec une moyenne de 10 sessions concurrentes, une mesure par heure sur une période de 7 jours (durée par défaut de la rétention), l'espace consommé se situe entre 200 et 300 Mo.
- ❑ La réduction de la fréquence de mesure ou de la durée de rétention diminuent la finesse de l'échantillonnage et peuvent affecter les recommandations faites

## ***AUTOMATIC WORKLOAD REPOSITORY (AWR)***

- ❑ Durée de rétention conseillée : période pendant laquelle les données doivent être disponibles pour récupération.
- ❑ Si le cycle de fonctionnement de l'application est hebdomadaire avec des traitements chaque fin semaine, une période de rétention d'une semaine est conseillée.
- ❑ Si le cycle est mensuel, avec des traitements importants répartis sur tout le mois, la période de rétention devra être adaptée à cette fréquence.
- ❑ On peut paramétrer AWR à partir d'Oracle Enterprise Manager

## ***AUTOMATIC DATABASE DIAGNOSTIC MONITOR (ADDM)***

L'analyse d'ADDM est exécutée chaque fois qu'un cliché de performances est pris par AWR. Il convient donc que les clichés soient fréquents pour bénéficier d'un échantillonnage correct mais pas trop de façon à ce qu'AWR et ADDM ne viennent pas perturber les performances de la base.

ADDM permet :

- ❑ d'analyser la base heure par heure ;
- ❑ de bénéficier du croisement d'une multitude d'éléments mesurés ;
- ❑ de bénéficier d'années d'expériences d'Oracle en ce domaine ;
- ❑ d'identifier la cause d'une baisse de performance et pas uniquement les symptômes ;
- ❑ de profiter des recommandations d'Oracle pour améliorer les performances ;
- ❑ d'appliquer immédiatement ces recommandations sur la base.

**La simplicité de l'outil permet d'effectuer un tuning en « boucle » : mesure de performances par AWR, recommandations par ADDM, mise en place de ces recommandations et ainsi de suite.**

## EXEMPLE

Base de données : PROD\_gilles.fr > Centre de conseil Connecté en tant que SYSTEM

Centre de conseil Page régénérée 2 juil. 2005 15:57:51 [Régénérer](#)

**Fonctions de conseil**

ADDM	Fonction de conseil sur la mémoire	Fonction de conseil sur les segments
SQL Tuning Advisor	Fonction de conseil sur la durée moyenne de récupération	Gestion de l'annulation (undo)
SQL Access Advisor		

**Tâches de la fonction de conseil** [Change Default Expiration](#)

Rechercher 989x623

Sélectionnez un type de conseil et entrez éventuellement un nom de tâche pour filtrer les données affichées dans l'ensemble de résultats (ResultSet).

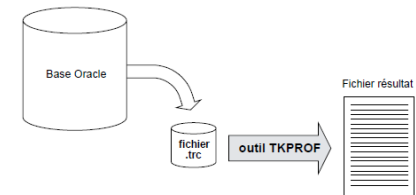
Type de conseil: Tous les types | Nom de tâche: 989x623 | Exécutions de la fonction de conseil: Dernière exécution | [Exécuter](#)

**Résultats** [Visualiser le résultat](#) [Supprimer](#) [Actions](#) [Reprogrammer](#) [Exécuter](#)

Sélectionner	Type de conseil	Nom	Description	Utilisateur	Statut	Heure de début	Heure de fin	Expire dans (jours)
<input type="radio"/>	ADDM	ADDM_4262180207_1_151	ADDM auto run: snapshots [150, 151], instance 1, database id 4262180207	SYS	TERMINE	26 juin 2005 23:00:42	26 juin 2005 23:00:42	24
<input type="radio"/>	SQL Tuning Advisor	SQL_TUNING_11118969249731		SYSTEM	TERMINE	15 juin 2005 22:44:24	15 juin 2005 22:44:29	13

## LE SQL ACCESS ADVISOR

Le SQL Access Advisor d'Oracle Enterprise Manager permet d'obtenir des recommandations concernant les index, les vues, les vues matérialisées pour améliorer les performances des requêtes.



## **TKPROF**

- ❑ Le principe de *tkprof* consiste à forcer la base de données à créer un fichier trace, puis à exploiter le contenu de ce fichier pour y accéder de façon lisible.
- ❑ L'avantage principal de *tkprof* est de ne pas être intrusif : vous pouvez ainsi étudier une session utilisateur, récupérer l'ensemble des ordres SQL transmis à la base de données, visualiser leur plan d'exécution, etc., sans disposer des sources de cette application.
- ❑ On peut ainsi visualiser les ordres SQL défaillants pour agir sur la base (ajout d'index, réorganisation des données...) ou faire pression sur un prestataire pour qu'il améliore son application.
- ❑ La création du fichier trace s'effectue soit pour une session (cas de déclenchement à partir de SQL\*Plus), soit pour toutes les sessions de la base de données. Dans ce cas, on doit contrôler le volume des fichiers trace créés.