

Objectifs

- ⌘ Comprendre l'architecture d'un serveur de BD Oracle
- ⌘ Démarrage et arrêt d'une *instance* et d'une *base de données* Oracle
- ⌘ Création d'une base de données opérationnelle
- ⌘ Gestion des *fichiers* d'une base de données Oracle
- ⌘ Gestion de la *structure logique* (tablespaces, segments, extents et blocs)
- ⌘ Gestion des *utilisateurs*, des *privileges* et des *ressources*.
- ⌘ Sauvegarde/restauration/réorganisation des données

Administration des BD relationnelles

Plan détaillé

⌘ Introduction

- ☒ Le rôle de l'*administrateur base de données*
- ☒ Composants de l'architecture d'Oracle

⌘ Serveur Oracle

- ☒ Gestion d'une instance Oracle
- ☒ Création d'une base de données
- ☒ Création de vues du dictionnaire de données et Création des packages standards

⌘ Les fichiers

- ☒ Mise à jour du fichier de contrôle
- ☒ Mise à jour des fichiers de reprise
- ☒ Gestion des Tablespaces et des fichiers de données

Plan détaillé (2)

⌘ Structure logique de stockage

- ☒ Notions sur les segments
- ☒ Gestion des rollback segments
- ☒ Gestion des segments temporaires

⌘ Sécurité

- ☒ Gestion des utilisateurs
- ☒ Gestion des profils
- ☒ Gestion des privilèges
- ☒ Gestion des rôles

⌘ Sauvegarde/Restauration

Introduction à l'administration des BD

Rôles de l'administrateur de BD

- ⌘ son rôle est avant tout **technique**
- ⌘ il doit néanmoins bien comprendre l'organisation de tout le reste ...
- ⌘ ses grandes fonctions :
 - ☒ installer le SGBD et les applications clientes
 - ☒ Créer la base de données et la *maintenir*
 - ☒ Gérer les *utilisateurs*
 - ☒ Assurer la *cohérence* et la *sécurité* des données
 - ☒ Optimiser la *base de données*
 - ☒ Améliorer les performances
 - ☒ gestion des ressources mémoires
 - ☒ gestion des temps de réponses prohibitifs

II. Composants de l'architecture d'Oracle

1. Les clients et les serveurs d'Oracle
2. Les processus, les structures de la mémoire et les fichiers d'Oracle
3. Étapes du traitement d'une requête d'interrogation
4. Étapes du traitement d'un ordre de modification/ajout/suppression
5. Étapes du traitement d'un ordre de fin de transaction (commit)

1. Les clients et les serveurs d'Oracle

1 *serveur* <--> plusieurs *clients*
Oracle Oracle

⌘ Utilisateurs

- ☒ Connexion directe à l'hôte
 - ☒ e.g. avec sql*plus
 - ☒ Connexion client/serveur ou deux tiers
 - ☒ e.g. avec les outils OEM (e.g. SQL Worksheet)
- ou avec une application développée en Client/Serveur (e.g. avec Developer 2000)

Connexion à une base de données

- ⌘ via SQL*Plus ou application tierce => lancement d'un *processus utilisateur* sur le poste du client (qui peut être le serveur)
- ⌘ lors de la connexion (login/passwd/service BD) => lancement d'un *processus serveur* sur le serveur
- ⌘ communication
 - ☒ inter-processus si les 2 processus s'exécutent sur le même poste
 - ☒ via un logiciel de réseau sinon
- ⌘ Session : connexion spécifique entre un *utilisateur* et un serveur *prêt à l'emploi*, i.e. un serveur démarré.

Processus Utilisateur

- ⌘ Fonctionne sur la machine du client
- ⌘ Démarre lors de l'appel d'un outil ou d'une application
- ⌘ Exécute l'outil ou l'application (SQL*Plus, Server Manager, OEM, Developer/2000)
- ⌘ Inclut l'UPI (*User Program Interface*)
- ⌘ Appelle le serveur Oracle

Processus Serveur

- ⌘ Fonctionne sur la même machine que le serveur Oracle
- ⌘ En cas de *serveur dédié*, prend en charge un *unique* processus utilisateur
- ⌘ Utilise une PGA exclusive (*Program Global Area*)
- ⌘ Inclut l'OPI (*Oracle Program Interface*)
- ⌘ Traite les appels générés par le client
- ⌘ Retourne les résultats au client
- ⌘ En cas de *serveur partagé* (MTS), plusieurs processus utilisateurs peuvent partager un processus serveur

Architecture

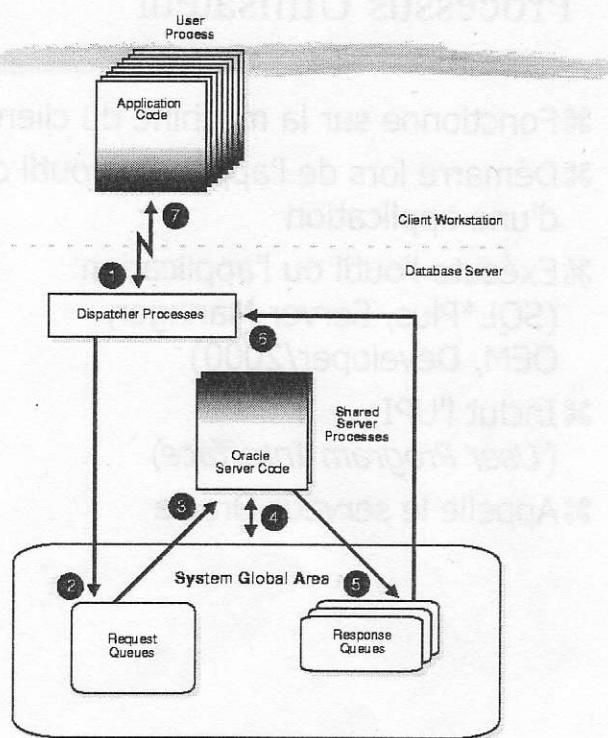
- ⌘ Architecture à *serveur dédié*

1 processus	<->	1 processus
client		serveur

 - ☒ Architecture
 - ☒ la plus simple
 - ☒ la plus répandue
 - ☒ Inadapté si beaucoup de clients connectés de façon simultanée
- ⌘ Architecture à *serveur partagé*

n processus	<->	1 processus
clients		serveur

 - ☒ Serveur partagé ou Multi-Thread Server (MTS)
 - ☒ Gestion d'une file de requêtes des processus clients et de réponses à retourner
 - ☒ Dispatcher + Listener



2. Serveur Oracle

⌘ Serveur Oracle =

⌘ Instance Oracle

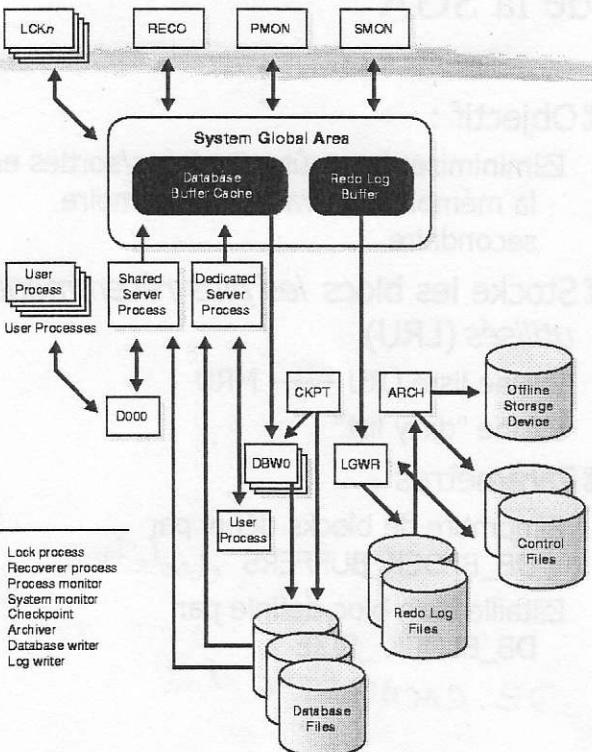
- ▢ C'est un moyen pour accéder à une base de données Oracle
- ▢ Ouvre une *unique* base de données
- ▢ Instance Oracle = SGA + des processus en arrière plan pour gérer la base

Instance Oracle : la SGA

- ⌘ SGA (Shared Global Area) ou Zone mémoire globale du système
- ⌘ contient les données et les informations de contrôle pour le serveur Oracle
 - ▢ Fait le lien avec le processus serveur
- ⌘ alloué en mémoire virtuelle par le système d'exploitation du serveur
- ⌘ Elle comprend
 - ▢ Une zone partagée (shared pool)
 - ▢ Le cache de données
 - ▢ Le cache de reprise (redo log)

Instance Oracle : Processus en arrière plan

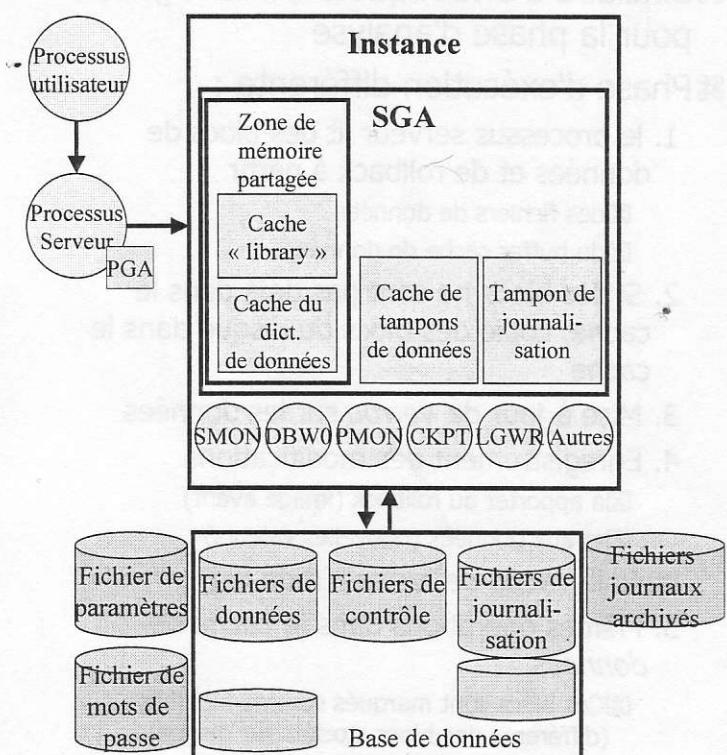
- ⌘ 5 principaux processus sont lancés pour une instance donnée
 - ▢ DBWR (processus d'écriture des blocs de données)
 - ▢ LGWR (processus d'écriture du fichier de reprise)
 - ▢ CKPT : point de synchronisation
 - ▢ SMON (processus System Monitor)
 - ▢ PMON (processus Process Monitor)
- ⌘ D'autres processus sont lancés
 - ▢ suivant le type de serveur
 - ▢ serveur dédié/partagé
 - ▢ suivant le mode d'archivage
 - ▢ archivelog ou noarchivelog
 - ▢ ...



BD Oracle

- ⌘ désignée par un nom (DB_NAME), souvent pris identique à celui de l'instance correspondante.
- ⌘ représente les structures physiques des données
- ⌘ se compose de *fichiers* du système d'exploitation
 - ☒ fichiers de données (>=1)
 - ☒ fichiers de reprise (redo log) (>=2)
 - ☒ fichiers de contrôle (>=1)
 - ☒ fichier de paramètres
 - ☒ fichier mot de passe
 - ☒ fichiers de reprise archivés

3. Traitement d'une requête d'interrogation



- ⌘ Le processus client envoie la requête au processus serveur
- ⌘ Le processus serveur effectue l'analyse syntaxique et sémantique de la requête
 - ☒ utilise la shared pool de la SGA pour compiler l'ordre
 - ☒ retourne l'état (analyse correcte ou incorrecte) au processus client.
- ⌘ Exécution de la requête
- ⌘ Récupération des résultats
 - ☒ le processus serveur envoie les lignes extraites par la requête (à partir du cache de données si les données y ont déjà été chargées)

La shared Pool de la SGA

⌘ taille définie par SHARED_POOL_SIZE

⌘ composée de :

- ☒ le « library cache » : contient le texte de la requête, le code analysé et un plan d'exécution déterminé par l'optimiseur
- ☒ le cache du dictionnaire de données : pour l'analyse sémantique de la requête

Le cache de données de la SGA

⌘ Objectif :

- ☒ minimiser les coûts d'entrées/sorties entre la mémoire centrale et la mémoire secondaire

⌘ Stocke les blocs *les plus récemment utilisés* (LRU)

- ☒ Une liste LRU ----- MRU
- ☒ Une "dirty list"

⌘ Paramètres

- ☒ nombre de blocks défini par DB_BLOCK_BUFFERS *obsolete*
- ☒ taille d'un bloc défini par DB_BLOCK_SIZE
DB_CACHE_SIZE *tail*

La zone mémoire du programme (PGA)

⌘ Zone mémoire utilisée par un seul processus serveur

- ☒ à la différence de la SGA qui est partagée par tous les processus serveurs

⌘ PGA contient :

- ☒ une zone de tri
- ☒ des informations sur la session
- ☒ l'état du curseur
- ☒ ...

4. Traitement d'un ordre de mise à jour (DML)

⌘ Similaire à une requête d'interrogation pour la phase d'analyse

⌘ Phase d'exécution différente :

1. le processus serveur lit des blocs de données et de rollback à partir
 - ☒ des fichiers de données
 - ☒ du buffer cache de données
2. Si des blocs ne sont pas déjà dans le cache, copie des blocs du disque dans le cache
3. Mise à jour de *verrou* sur les données
4. Enregistrement des modifications
 - ☒ à apporter au rollback (image avant)
 - ☒ et aux données (nouvelles valeurs)dans le *cache de reprise* (redo log)
5. Mêmes opérations dans le *cache de données*
 - ☒ Ces blocs sont marqués comme modifiés (différents des blocs stockés sur disque)

Rollback Segment

- ⌘ Permet la gestion des *transactions*
 - ☒ COMMIT : valide un ensemble de MAJ
 - ☒ ROLLBACK : annule un ensemble de MAJ
- ⌘ Avant d'effectuer une modification des données, le processus serveur enregistre l'ancienne valeur dans un *rollback segment*
- ⌘ Intérêts :
 - ☒ si la transaction échoue, les modifications "déjà faites" sont annulées
 - ☒ Assure une lecture cohérente des données par les autres transactions
 - ☒ en cas de panne, les données sont restaurées dans un état cohérent.

Cache de reprise

- ⌘ Redo log = reprise
- ⌘ Fait partie de la SGA
- ⌘ Taille définie par LOG_BUFFER
- ⌘ Stocke la trace des modifications
 - ☒ le bloc modifié
 - ☒ l'emplacement de la modification
 - ☒ la nouvelle valeur
 - ☒ les 3 constituent une "entrée redo"
- ⌘ Pas de distinction sur le type du bloc
 - ☒ bloc d'index, de données, ...
- ⌘ Cache circulaire
 - ☒ capacité infinie !

Database Writer (DBWR)

- ⌘ Objectifs :
 - ☒ Limiter les coûts d'entrées/sorties en "retardant" (de façon transparente à l'utilisateur) l'écriture des blocs modifiés
- ⌘ DBWR
 - ☒ un des processus serveur en arrière plan
 - ☒ enregistre les modifications du cache de données *dans les fichiers de données*
 - ☒ Quand ?
 - ☒ nb blocks modif > seuil
 - ☒ pas assez de blocks libres
 - ☒ programme se synchro (checkpoint program)
 - ☒

Log Writer (LGWR)

- ⌘ Objectifs
 - ☒ Enregistrer *toutes* les modifications apportées aux données afin d'assurer la *reprise après panne*
- ⌘ Enregistre les modifications du cache de reprise de la SGA *dans les fichiers de reprise*
- ⌘ Effectue des écritures séquentielles
 - cache de reprise plein
 - lorsque transaction validée (committé)
 - juste avant que DB writer ne fonctionne
- ⌘ Comme le cache de reprise, les fichiers de reprise sont organisés de façon circulaire
 - ☒ capacité infinie

Traitement des opérations COMMIT

⌘ SCN pour *System Change Number*

- ☒ à chaque transaction validée -> 1 identifiant de modification SCN
- ☒ sorte d'horodateur interne
 - ☒ permet de vérifier la cohérence indépendamment de la date/heure du SE

⌘ Lorsqu'un ordre COMMIT est effectué:

1. Le processus serveur enregistre dans le cache de reprise 1 SCN
2. LGWR effectue une lecture contiguë de tout le cache de reprise (SCN inclu) dans les fichiers de reprise
3. L'utilisateur reçoit "commit complete"
4. Le processus serveur libère les verrous

III. Les outils d'administration

⌘ Mode ligne de commandes

- ☒ svrmgr30 sous NT
 - ☒ remplace SQL*DBA
- ☒ SQL*Loader
- ☒ import/export
- ☒ oradim80, orapwd80, ...

⌘ Mode graphique

- ☒ Outils d'OEM
 - ☒ Instance Manager
 - ☒ Storage Manager
 - ☒ SQL Worksheet
 - ☒ Schema Manager
 - ☒ Security Manager
 - ☒ Backup Manager
 - ☒ Data Manager
- ☒ Autres outils Oracle :
 - ☒ Oracle Net8

Objectifs

Instance Oracle

- ⌘ Création du *fichier de paramètres*
- ⌘ Démarrage d'une instance et ouverture de la base de données
- ⌘ Fermeture de la base de données et arrêt de l'instance
- ⌘ Obtention et initialisation des paramètres
- ⌘ Gestion des *sessions*
- ⌘ Contrôle des fichiers d'alertes et de traces

Instance Oracle

⌘ Préalable obligatoire pour accéder à une base de données Oracle

⌘ Instance Oracle

☒ associée à une *unique* base de données

☒ SGA

☒ processus en arrière plan

⌘ Tâches courantes de l'administrateur

☒ de démarrer l'instance

☒ puis de monter la BD et de l'ouvrir

☒ de l'arrêter

☒ après avoir fermer puis démonter la BD

Utilisateurs administrateurs

⌘ Utilisateur SYS et SYSTEM

☒ créés automatiquement (rôle DBA)

⌘ SYS

☒ mot de passe : change_on_install

☒ propriétaire du dictionnaire

☒ relations + vues dans le schéma SYS

☒ possède tous les droits

⌘ SYSTEM

☒ mot de passe : manager

☒ moins de priviléges que SYS

SQL>connect sys/123456 as sysdba
SQL> connect / as sysdba.

Instance Oracle : Rappel (2)

⌘ SGA (Shared Global Area) ou Zone mémoire globale du système

⌘ contient les données et les informations de contrôle pour le serveur Oracle

⌘ alloué en mémoire virtuelle (gérée par le système d'exploitation du serveur)

⌘ Elle comprend

☒ la shared pool : stocke les informations les plus récemment utilisées (e.g. requêtes SQL)

☒ le cache de données : stocke les données les plus récemment utilisées

☒ le cache de reprise : utilisé pour enregistrer les modifications apportées à la base de données

5 principaux processus lancés pour une instance Oracle :

☒ **DBWR** (processus d'écriture des blocs de données modifiés) : *écrit les données modifiées dans la base de données*

☒ **LGWR** (processus d'écriture des modifications) : *enregistre les modifications enregistrées dans le cache de reprise dans les fichiers de reprise*

☒ **SMON** (processus System Monitor) : *vérifie la cohérence et redémarre la base de données après panne*

☒ **PMON** (processus Process Monitor) : *nettoie les ressources lors de l'échec d'un processus*

☒ **CKPT** : *Synchronise et assure la cohérence des données*

Création d'une instance

- ⌘ Pouvoir se connecter comme administrateur
- ⌘ Editer le *fichier de paramètres* init<sid>.ora
- ⌘ Sous svrmgr, commande STARTUP
 - ☒ Exemple :
 - ☒ connect sys/oracle as sysdba
 - ☒ startup nomount
pfile=d:\data\initU01.ora
- ⌘ Aucune vérification de cohérence à ce niveau
- ⌘ Allocation de la SGA

```
c:\ > oradim - NEW
      - SID ORCL
      - INTPWD oracle
      - STARTMODE AUTO
      - PFILE C:\ ... \ initisima.ora
```

c:\ met |start| oracle services
stop

Le fichier de paramètres

⌘ Fichier init<SID>.ora

☒ e.g. initORCL.ora

⌘ Paramètre obligatoire :

☒ db_name : idf de la bd, < à 8 caractères

⌘ Paramètres communément modifiés:

- ☒ control_files : nom des fichiers de contrôle
- ☒ compatible : version du serveur avec laquelle cette instance est compatible
- ☒ db_block_buffers : nombre de bloc mis en cache dans la SGA (min 50)
- ☒ shared_pool_size : taille en octet de la zone de partage (min 3.500.000)
- ☒ log_buffer : Nombre d'octets alloués au cache de reprise dans la SGA
- ☒ user_dump_dest : emplacement où les fichiers trace sont créés
- ☒ processes : nombre max de processus du SE pouvant se connecter simultanément à une instance
- ☒ sql_trace : active ou pas un espion pour chaque session utilisateur
- ☒ max_dump_file_size : taille max du fichier de trace
- ☒ timed_statistics : active ou pas le minutage dans les fichiers trace

Instance et Service NT

- ⌘ serveur Oracle = instance + BD
- ⌘ Sous NT, à chaque serveur est associé un *service*
- ⌘ Pour créer un service, il faut utiliser l'outil ORADIM80
 - ☒ permet d'exécuter plusieurs serveurs Oracle sur la même machine
 - ☒ Attention au nombre de caractères
☒ < à 4 pour les noms de service sous 8.0
 - ☒ Créer une instance
 - ☒ ORADIM80 -NEW -SID ORCL
[-INTPWD INTERNAL_PWD]
([-SRVC SRVCNAME])
[-MAXUSERS NUMBER]
[-STARTMODE AUTO, MANUAL]
[-PFILE FILENAME]
 - ☒ Supprimer une instance
 - ☒ ORADIM80 -DELETE -SID ORCL

Exemple de fichier de paramètres

```
db_name = u01
db_files = 1020
control_files = C:\orant\database\ctl1u01.ora
db_file_multiblock_read_count = 8
db_block_buffers = 200
shared_pool_size = 6500000
log_checkpoint_interval = 8000
processes = 50
dml_locks = 100
log_buffer = 8192
sequence_cache_entries = 10
sequence_cache_hash_buckets = 10
#audit_trail = true
#timed_statistics = true
background_dump_dest = C:\orant\rdbms80\trace
user_dump_dest = C:\orant\rdbms80\trace
db_block_size = 2048
compatible = 8.0.4.0.0
sort_area_size = 65536
log_checkpoint_timeout = 0
remote_login_passwordfile = shared
max_dump_file_size = 10240
...
```

Démarrage par étapes

⌘ 3 options lors du démarrage

- ☒ **nomount** : seul l'instance est démarrée
- ☒ **mount** : fichier de contrôle ouvert pour cette instance
- ☒ **open** : tous les fichiers définis pour cette instance dans le fichier de contrôle sont ouverts

⌘ Certaines commandes nécessitent que la base soit dans une de ces options

- ☒ étape NOMOUNT pour la création de la base de données
- ☒ étape MOUNT pour renommer des fichiers de reprise, ...

La commande STARTUP

⌘ Syntaxe :

```
STARTUP [FORCE] [RESTRICT] [PFILE=filename]
[EXCLUSIVE | PARALLEL | SHARED]
[ OPEN [RECOVER][DATABASE]
| MOUNT | NOMOUNT]
```

- ☒ **open** : permet aux utilisateurs d'accéder à la base
- ☒ **mount** : monte la base mais accès limités
- ☒ **nomount** : sga + processus arrière plan. Pas d'accès à la base
- ☒ **exclusive** : seule cette instance accède la base
- ☒ **parallel** ou **shared** : avec server // Oracle
- ☒ **force** : annule l'instance courante avant de démarrer normalement
- ☒ **restrict** : accès réservés aux utilisateurs ayant le privilège RESTRICTED SESSION
- ☒ **recover** : démarre la restauration quand la base démarre

La commande SHUTDOWN

⌘ Syntaxe :

```
SHUTDOWN [NORMAL | TRANSACTIONNAL
| IMMEDIAT | ABORT]
```

Mode d'arrêt	A	I	T	N
nouvelles connexions permises	0	0	0	0
attend la fin des sessions en cours	0	0	0	1
attend la fin des transactions en cours	0	0	1	1
force un point de synchronisation et ferme	0	1	1	1

⌘ Temps d'arrêt : N -> T -> I -> A

Vues dynamiques sur les performances

⌘ Entretenuées par le serveur Oracle

- ☒ mises à jour automatiquement
- ☒ image fidèle du serveur

⌘ Décrivent les structures du disque et de la mémoire

⌘ Utiles pour le tuning des performances

⌘ Préfixées par V\$

- ☒ définies à partir de tables normalisées préfixées par X\$

⌘ Accessible par l'administrateur à l'étape NOMOUNT

⌘ V\$FIXED_TABLE : les décrivent toutes

Exemples

⌘ SGA

☒ V\$PARAMETER, V\$SGA, V\$OPTION,
V\$PROCESS, V\$SESSION, V\$VERSION,
V\$INSTANCE

⌘ Fichiers :

☒ V\$THREAD, V\$CONTROLFILE,
V\$DATABASE, V\$DATAFILE,
V\$DATAFILE_HEADER, V\$LOGFILE

⌘ Afficher les valeurs courantes des paramètres

☒ SHOW PARAMETER control
OU

☒ select name from V\$PARAMETER
where name like '%control%';

	V\$SQL V\$SQLAREA V\$SQLTEXT V\$SQLTEXT_WITH_NEWLINES V\$SQL_BIND_DATA (V8) V\$SQL_BIND_METADATA (V8) V\$SQL_CURSOR V\$SQL_SHARED_MEMORY
BACKUPS, ARCHIVE, RESTAURATION	
V\$ARCHIVE V\$ARCHIVED_LOG (V8) V\$ARCHIVE_DEST (V8) V\$BACKUP V\$BACKUP_CORRUPTION (V8) V\$BACKUP_DATAFILE (V8) V\$BACKUP_DEVICE (V8) V\$BACKUP_PIECE (V8) V\$BACKUP_REDOLOG (V8) V\$BACKUP_SET (V8) V\$DELETED_OBJECT (V8) V\$RECOVERY_FILE_STATUS V\$RECOVERY_LOG V\$RECOVERY_STATUS V\$RECOVER_FILE	
DATABASES et INSTANCES	V\$ACTIVE_INSTANCES V\$BGPROCESS V\$BH (V8) V\$COMPATIBILITY V\$COMPATSEG V\$COPY_CORRUPTION (V8) V\$DATABASE V\$DATAFILE V\$DATAFILE_COPY (V8) V\$DATAFILE_HEADER (V8) V\$DBFILE V\$DBLINK V\$DB_PIPES V\$INSTANCE V\$LICENSE V\$OFFLINE_RANGE (V8) V\$OPTION V\$SGA V\$SGASTAT V\$TABLESPACE (V8) V\$VERSION
CACHE MANAGEMENT	V\$CACHE (V8) V\$DB_OBJECT_CACHE V\$LIBRARYCACHE V\$ROWCACHE V\$SUBCACHE (V8)
CONTROL FILES	V\$CONTROLFILE V\$CONTROLFILE_RECORD_SECTION (V8)
CURSORS et SQL	V\$OPEN_CURSOR

SQL*LOADER (Chargement direct)

V\$LOADCSTAT
V\$LOADPSTAT (V8)
V\$LOADTSTAT
V\$LOCKED_OBJECT

FIXED Vues

V\$FIXED_TABLE
V\$FIXED_VIEW_DEFINITION
V\$INDEXED_FIXED_COLUMN

I/O

V\$FILESTAT
V\$WAITSTAT

Verrous

V\$BUFFER_POOL (V8)
V\$CACHE_LOCK
V\$CLASS_PING (V8)
V\$DLM_CONVERT_LOCAL (V8)
V\$DLM_CONVERT_REMOTE (V8)
V\$DLM_LATCH (V8)
V\$DLM_MISC (V8)
V\$ENQUEUE_LOCK (V8)
V\$EVENT_NAME
V\$FALSE_PING
V\$FILE_PING (V8)
V\$LATCH
V\$LATCHHOLDER
V\$LATCHNAME
V\$LATCH_CHILDREN

V\$LATCH_MISSES
V\$LATCH_PARENT
V\$LOCK
V\$LOCK_ACTIVITY
V\$LOCK_ELEMENT

V\$LOCKS_WITH_COLLISIONS
V\$PING
V\$RESOURCE
V\$RESOURCE_LIMIT (V8)
V\$TRANSACTION_ENQUEUE (V8)

V\$_LOCK
V\$_LOCK1

V\$TIMER
V\$TYPE_SIZE
V\$_SEQUENCES (V8)

Serveur MULTI-THREADED et PARALLEL

V\$CIRCUIT
V\$DISPATCHER
V\$DISPATCHER_RATE (V8)
V\$MTS
V\$QUEUE
V\$REQDIST
V\$SHARED_SERVER
V\$THREAD

PERFORMANCE

V\$GLOBAL_TRANSACTION (V8)
V\$OBJECT_DEPENDENCY
V\$SHARED_POOL_RESERVED
V\$SORT_SEGMENT
V\$SORT_USAGE (V8)
V\$STATNAME
V\$SYSSTAT
V\$SYSTEM_CURSOR_CACHE
V\$SYSTEM_EVENT
V\$TRANSACTION

OPTION REQUETE PARALLELE

V\$EXECUTION
V\$EXECUTION_LOCATION (V8)
V\$PQ_SESSTAT
V\$PQ_SLAVE
V\$PQ_SYSSTAT
V\$PQ_TQSTAT

PARAMETRES ORACLE

V\$NLS_PARAMETERS
V\$NLS_VALID_VALUES
V\$PARAMETER
V\$SYSTEM_PARAMETER

REDO LOGS

V\$LOG
V\$LOGFILE
V\$LOGHIST
V\$LOG_HISTORY

ROLLBACK SEGMENTS

V\$ROLLNAME
V\$ROLLSTAT

SECURITE et PRIVILEGES

V\$ENABLEDPRIVS
V\$PWFILE_USERS

SESSIONS

V\$ACCESS
V\$MYSTAT
V\$PROCESS
V\$SESSION
V\$SESSION_CONNECT_INFO
V\$SESSION_CURSOR_CACHE
V\$SESSION_EVENT
V\$SESSION_LONGOPS (V8)
V\$SESSION_OBJECT_CACHE (V8)
V\$SESSION_WAIT
V\$SESSTAT
V\$SESS_IO

Paramètres d'initialisation dynamiques

⌘ Modification de certains paramètres en cours d'instance

☒ Ordre ALTER SYSTEM ou ALTER SESSION

⌘ Exemples :

☒ ALTER SESSION SET SQL_TRACE=true;

☒ modifie seulement la valeur pour la session courante

☒ ALTER SYSTEM SET TIMED_STATISTICS=true;

☒ modification globale

☒ ALTER SYSTEM SET SORT_AREA_SIZE=131072 DEFERRED;

☒ modification pour les autres sessions

Mode "restricted session"

⌘ Utiliser l'option RESTRICT au démarrage de l'instance

☒ l'accès est ainsi limité dès le début

⌘ Utiliser ALTER SYSTEM

☒ pour rendre l'accès à une instance limitée

☒ exemples :

☒ ALTER SYSTEM ENABLE RESTRICTED SESSION

☒ ALTER SYSTEM DISABLE RESTRICTED SESSION

Fermeture de sessions

⌘ Identifier la session à fermer

☒ interroger la vue V\$SESSION

☒ Exemple :

☒ select sid, serial#

from V\$SESSION

where username='SCOTT';

⌘ Supprimer la avec un ordre ALTER SYSTEM

☒ Exemple :

☒ ALTER SYSTEM KILL SESSION '7,117';

Fichiers trace

⌘ Remplis par le serveur et les processus en arrière plan

⌘ Tout y est consigné

☒ le fichier ALERT constitue un journal chronologique des messages et des erreurs

⌘ Possible de l'activer

☒ avec ALTER SESSION

☒ le paramètre SQL_TRACE

⌘ Intérêts :

☒ permet de détecter les erreurs internes et les corruptions de blocs

☒ surveiller (espionner) les opérations sur la base de données

Informations sur l'instance

⌘ Interroger la vue V\$INSTANCE

- ☒ utiliser la commande describe
- ☒ faire des requêtes

⌘ commande SHOW

Base de données Oracle

Prérequis

⌘ Un compte autorisé DBA

- ☒ tous les priviléges sur le système d'exploitation

⌘ Une mémoire suffisante pour le démarrage de l'instance

- ☒ SGA
- ☒ exécutable Oracle
- ☒ les processus

⌘ Un espace disque suffisant pour stocker la base de données

- ☒ au moins 1 fichier de données
- ☒ au moins 2 fichiers de reprise
- ☒ au moins 1 fichier de contrôle

⌘ Pour le disque, il faut *estimer* ...

BD Oracle : Rappel

⌘ désignée par un nom DB_NAME

- ☒ souvent le même que celui de l'instance
- ☒ représente les structures physiques des données

⌘ se compose de *fichiers* du SE

- ☒ fichiers de données ($>=1$)
 - ☒ stocke le dictionnaire des données, les objets utilisateurs et les « images avant » des données
- ☒ fichiers de reprise (redo log) ($>=2$)
 - ☒ stocke toutes les modifications apportées à la base de données (pour la reconstruire en cas de panne)
- ☒ fichiers de contrôle ($>=1$)
 - ☒ contient les informations nécessaires à la mise à jour et à la vérification de l'intégrité des données.
- ☒ Autres fichiers :
 - ☒ fichier de paramètres, fichier mot de passe, fichiers de reprise archivés

Création d'une base : Remarques

⌘ Sous Unix :

- ☒ Crédit automatique au cours d'une installation
- ☒ Crédit manuelle après installation

⌘ Sous NT :

- ☒ Crédit automatique au cours d'une installation
- ☒ Crédit via Oracle Database Assistant ou création manuelle

⌘ En cas de création manuelle, utiliser des *scripts de création*

Création Manuelle d'une base de données

- ⌘ Choisir un nom unique pour l'instance et la base de données
- ⌘ Choisir un jeu de caractères
- ⌘ Définir les variables du SE
- ⌘ Préparer le fichier des paramètres
 - (⌘ Crédit un fichier mot de passe)
- ⌘ Démarrer l'instance
- ⌘ Crédit la base de données
- ⌘ Exécuter les scripts pour générer le dictionnaire de données

```
→ C:\> set oracle_sid = isima
→ fichier de param
→ oraclim - ...
→ SQLPLUS /nolog
SQL> connect / as sysdba
SQL> startup nomount pfile= ...
SQL> create database ...
```

Démarrage de l'instance et création de la BD

Fichier de paramètres

- ⌘ Crédit un fichier init<sid>.ora à partir d'un fichier existant
- ⌘ Editer le pour modifier les paramètres

```
db_name = u01
db_files = 1020
control_files = 'C:\data\ctl1u01.ora'
db_file_multiblock_read_count = 8
db_block_buffers = 200
shared_pool_size = 6500000
log_checkpoint_interval = 8000
processes = 50
dml_locks = 100
log_buffer = 8192
sequence_cache_entries = 10
sequence_cache_hash_buckets = 10
#audit_trail = true
#timed_statistics = true
background_dump_dest = C:\orant\rdbms80\trace
user_dump_dest = C:\orant\rdbms80\trace
db_block_size = 2048
compatible = 8.0.4.0.0
sort_area_size = 65536
log_checkpoint_timeout = 0
remote_login_passwordfile = shared
max_dump_file_size = 10240
```

⌘ Se connecter en tant que SYSDBA

⌘ Démarrer l'instance à l'étape NOMOUNT

```
STARTUP NOMOUNT PFILE=d:\data\initU01.ora
```

⌘ Crédit de la base de données

⌘ Ordre CREATE DATABASE

```
create database U01
  maxfiles 32
  maxlogmembers 5
  maxdatafiles 30
  logfile
    group 1
      'd:\data\disk3\log1a.rdo' size 150 K,
    group 2
      'd:\data\disk3\log2a.rdo' size 150 K
  datafile
    'd:\data\disk1\system01.dbf' size 25 M
  character set WE8ISO8859P1;
```

```
SQL> ALTER database mount;
SQL> ALTER database open;
```

Après la création de la base de données

⌘ La base contient :

- ☒ les fichiers de données constituant le tablespace SYSTEM
- ☒ les fichiers de contrôle et de reprise
- ☒ l'utilisateur SYS et SYSTEM
- ☒ le rollback segment SYSTEM
- ☒ Tables X\$ + vues V\$

⌘ Pour modifier la base de données

- ☒ ordre ALTER DATABASE
- ☒ Exemple pour ouvrir une base de données qui est à l'état NOMOUNT :
 - ☒ ALTER DATABASE [MOUNT | OPEN]

⌘ En particulier, elle ne contient pas encore le *dictionnaire de données*

⌘ NB : pas de vue V\$ (ni sur le dictionnaire) pour la BD oracle

Dictionnaire de données

enrichi avec les supématrices normales

normalisées

composées de tables et de vues

Objectifs

⌘ Sur le dictionnaire :

- ☒ Construction des vues du dictionnaire de données
 - ☒ script catalog.sql
- ☒ Utilisation du dictionnaire
 - ☒ par exemple, pour l'évaluation sémantique des requêtes

Dictionnaire de données

⌘ Fournit des informations sur

- ☒ la structure logique et physique d'une base de données
- ☒ les noms, définitions, et allocation d'espace des objets des schémas
- ☒ les contraintes d'intégrité
- ☒ les utilisateurs, les priviléges de la base
- ☒ l'audit

⌘ Composé de tables et de vues :

☒ Les tables :

☒ tables normalisées

- exemple : IND\$, OBJ\$...

☒ créées à partir du script sql.bsq

☒ Les vues :

☒ simplifient l'accès aux tables

☒ créées à partir du script catalog.sql

Vues du dictionnaire

⌘ Le préfixe USER

✉ USER_XXX : objets qui *appartiennent* à l'utilisateur

✉ Exemple :

✉ USER_TABLES : contient toutes les tables d'un utilisateur

⌘ Le préfixe ALL

✉ ALL_XXX : objets qui sont *accessibles* aux utilisateurs (grâce colonne OWNER)

⌘ Le préfixe DBA

✉ DBA_XXX : objets de *toute la base*

✉ Accédées par

✉ l'administrateur ou

✉ un utilisateur ayant le privilège système
SELECT ANY TABLE

⌘ Ordre d'inclusion :

✉ USER <= ALL <= DBA

Exemples de vues

Vues	Description
dictionary	Généralités
dict_columns	
dba_tables	Informations sur les objets utilisateurs, e.g. tables, colonnes ...
dba_objects	
dba_lobs	
dba_tab_constraints	
dba_users	Informations sur les priviléges et les rôles utilisateurs
dba_sys_privs	
dba_roles	
dba_extents	Allocation d'espace pour les objets de la base
dba_free_space	
dba_segments	
dba_rollback_segs	Structure générale de la base
dba_data_files	
dba tablespaces	
dba_audit_trail	Information sur l'audit
dba_audit_objects	
dba_audit_obj_opts	

Scripts administratifs

⌘ Information catalogue et dictionnaire de données

✉ cat*.sql

✉ exemple : catrman.sql pour l'utilitaire Recovery Manager

⌘ Spécification des packages de la base

✉ dbms*.sql

⌘ Code compressé d'un package base de données

✉ prvt*.plb

⌘ Vues et tables pour les utilitaires de la base

✉ utl*.sql

✉ exemple : utlsampl.sql pour tables de démonstration