

TD/TP n°3 : Administration de bases de données Oracle

1. Arrêt/démarrage de la base :

<ul style="list-style-type: none"> • Commande STARTUP pour démarrer l'instance <ul style="list-style-type: none"> ○ Utilise le SPFILE ou le PFILE à préciser ○ NOMOUNT : base fermée et non montée ○ MOUNT : base fermée et montée ○ FORCE : ouvre de force, en tuant une éventuelle instance démarrée • Commande ALTER DATABASE si l'instance est déjà ouverte <ul style="list-style-type: none"> ○ MOUNT pour monter la base ○ OPEN pour ouvrir la base ○ OPEN READ ONLY ouverte en lecture seule • ALTER SYSTEM ENABLE RESTRICTED SESSION : seuls les DBA peuvent ouvrir des sessions 	<pre> graph TD nomount((nomount)) --> mount((mount)) mount --> exclusive((exclusive)) exclusive --> open((open)) </pre> <p>Maintenance - création - modifications structurelles, - restaurations</p> <p>Maintenance - saisie massive</p> <p>Exploitation</p>
--	---

Note : l'ordre complet de démarrage d'une instance est :

startup pfile=\$ORACLE_BASE/admin /nom_fichier_init.ora,

et que par défaut le nom pris par le paramètre est :

pfile = \${ORACLE_HOME}/dbs/init\${ORACLE_SID}.ora.

Arrêt d'une base de données

- Commande SHUTDOWN
- Plusieurs modes d'arrêt :
 - NORMAL : attend la déconnexion de tous les utilisateurs
 - IMMEDIATE : annule toutes les transactions non validées et tue les sessions en cours
 - TRANSACTIONAL : attend la fin des transactions puis tue les sessions
 - ABORT : tue les sessions, mais n'annule pas les transactions non validées
- ABORT est à utiliser en dernier recours
- ABORT nécessite une restauration de la base pour retrouver sa cohérence
- On peut suspendre une base de données : ALTER SYSTEM SUSPEND/RESUME

Chaque base de données dans oracle est basée sur un fichier nommé *.ora contenant les différents paramètres de configuration de la base.

Atelier 1

1. Se connecter à oracle en tant qu'administrateur ;
2. Editer le fichier de paramètres c:\orant\Database\initiorcl.ora avec v\$parameter ou show parameter
 - a. Quelle est le nom de la base de données associée à ce fichier ?
 - b. Quelles sont les fichiers de contrôle associés à cette base ?
 - c. Quelle est la taille d'un bloc de données dans cette base ?
3. Mettre en place les requêtes SQL, dans un script SQLPLUS, qui vont permettre de fournir un rapport détaillé contenant les informations suivantes :
 - Le nom de l'instance, L'identifiant unique qui correspond à cette instance, La version du SGBD pour cette instance, Le nom du serveur sur lequel l'instance est démarrée, La date de création de la base, Depuis quand la base est-elle démarrée, Le statut de la base de données (STARTED/OPEN/MOUNTED)?

INSTANCE	IDENTIFIANT	VERSION	SERVEUR	DATE DE CREATION	DATE DE DEMARRAGE	STATUT
-----	-----	-----	-----	-----	-----	-----

4. Arrêter l'instance « shutdown normal » ;
5. Consulter la vue v\$instance pour vérifier l'état de l'instance
6. Monter l'instance pas à pas, et vérifier l'état de l'instance dans chaque étape.

2. Création d'une nouvelle base de données :

Deux possibilités

- Utiliser l'assistant Oracle : graphique : Pour le lancer : Sous Windows : Menu Démarrer, Programmes, Oracle-OraDb11g_home1/Assistant Configuration de bases de données
- Créer manuellement à l'aide de scripts :
 - Spécifier le nom de l'instance SID
 - Créer le fichier de paramètres PFILE (partir de l'exemple Oracle) : Nom de la base, emplacement des fichiers de contrôle
 - Se connecter à l'instance inactive : SQLPLUS /nolog puis CONNECT / AS SYSDBA
 - Créer un fichier de paramètre serveur SPFILE (à partir du fichier manuel)
 - Lancer l'instance : STARTUP NOMOUNT (Aucune vérification de cohérence à ce niveau)
 - Créer la base de données : CREATE DATABASE...

Remarques complémentaires

- Si la création échoue, visualiser les fichiers d'alertes
 - Arrêter l'instance
 - Supprimer les fichiers créés pendant la tentative
- Pour supprimer une base :
 - Drop database
 - La base doit être montée et fermée
 - Conserve les fichiers d'archivage et les copies
- Pour modifier des paramètres
 - Les commandes ALTER SYSTEM pour les paramètres d'instance (modifie le SPFILE, pas le PFILE !)
 - Les commande ALTER DATABASE pour revenir sur les choix du CREATE DATABASE
- Pour voir les informations sur la base
 - Show parameters, vues 'database_properties' et V\$DATABASE
 - Les commande ALTER DATABASE pour revenir sur les choix du CREATE DATABASE

Configuration du réseau

- Côté serveur
 - Un processus d'écoute utilisant un protocole réseau : listenerSID
 - Un serveur BD Oracle = instance + base
 - Les configurations (=services d'écoute) sont enregistrées dans listener.ora
 - Editable à la main, ou via outils graphiques
- Côté client
 - Configurations (services d'accès) enregistrées dans tnsname.ora
 - Permet au client de se connecter de façon transparente

Atelier 2 : Utilisation de l'Assistant Configuration de bases de données

1. Créer une nouvelle base de données « **BDDWM** » : lancer l'assistant C:\Users\ADmiN>**dbca**
 - a. Après la création, positionner la variable d'environnement ORACLE_SID:
C:\Users\ADmiN>set ORACLE_SID=BDDWM
 - b. Se connecter en tant qu'administrateur : sqlplus sys / as sysdba
 - c. Vérifier le nom et le statut de l'instance active
 - d. Vérifier la taille de la SGA
 - e. Consulter les vues : v\$datafile, v\$controlfile et v\$logfile, pour localiser les fichiers de la base BDDWM.
 - f. Si l'instance est ouverte essayer de la fermer par : shutdown normal
 - g. Consulter la vue v\$instance pour vérifier l'état de l'instance
 - h. Sortir du mode Sqlplus
 2. Configurer le service de l'instance par défaut à mode manuel (démarrage de l'instance en mode manuel)
« oradim -EDIT -SID BDDWM -STARTMODE manual »
 3. Test de différents mode d'arrêt de la base
 - 1-Mode normal**
 - a. Redémarrer le service de la base que vous venez de créer, et se connecter en tant qu'administrateur : sqlplus sys / as sysdba
 - b. Vérifier l'état de l'instance, si la base est ouverte, l'arrêter avec le mode normal
 - c. Vérifier que les utilisateurs ne sont pas déconnectés, et aucune nouvelle connexion n'est acceptée.
- SQL> connect scott/tiger

ERROR:

ORA-01090: shutdown in progress - connection is not permitted

Dès que le dernier utilisateur se déconnecte la base s'éteint.

- d. Test avec une transaction en cours : Créer la tableTest (coll number) à laquelle on applique un « insert », une transaction est donc en cours. La transaction n'est pas arrêtée tant que l'utilisateur n'est pas déconnecté.

2-Mode transactionnel

- a. Redémarrer la base et l'arrêter avec : SQL> shutdown transactional
- b. Vérification que les utilisateurs sont déconnectés si aucune transaction n'est en cours.
Test avec une transaction en cours : La base de données attend que la transaction soit terminer (COMMIT ou ROLLBACK)

3-Mode immédiate

- a. Redémarrer la base et l'arrêter avec : SQL> shutdown immediate
- c. Vérification que les utilisateurs sont déconnectés. Mais on ne le voit pas au niveau client. Test avec une transaction en cours :
On ne voit rien niveau client, cependant si on souhaite valider la transaction on obtient l'erreur suivante :
SQL> commit;
ERREUR à la ligne 1 :
ORA-12571: TNS : échec à l'écriture du paquet
Et la base s'arrête proprement

4-Mode Abort

- a. Redémarrer la base et l'arrêter avec : SQL> shutdown abort
- b. Idem que précédemment. Dans ce cas la mise à niveau avec les redo log n'est pas faite contrairement au mode d'arrêt shutdown immédiate.

Atelier 3 : Création manuellement une base à l'aide de scripts (sous SQLPlus)

1- Création d'une variable DOS ORACLE_SID :

Nous allons utiliser une instance Oracle, qui sera identifiée par son SID (System IDentifier) :

👉 lancer CMD.EXE

👉 créer le SID TEST en tapant : **SET ORACLE_SID=BTEST**

Nous pouvons vérifier que cette variable a bien été créée en tapant SET :

👉 taper : **SET**

👈 DOS présente toutes les variables d'environnement, parmi lesquelles BTEST

2 - Création d'un Service

Nous devons à présent créer un Service Windows qui gèrera notre instance. Nous créons ce service en utilisant ORADIM.EXE.

👉 Taper: la commande

oradim -new -sid BTEST -intpwd mypass -startmode manual

ORADIM -NEW -SID BTEST -INTPWD root -STARTMODE manual

Nous vérifions que le service a bien été créé :

👉 Taper : **services.msc**

(ou utiliser "démarrer | paramètres | panneau de configuration | outils d'administration | services")

👈 Windows affiche les services utilisés par notre système, dont OracleServiceBTest:

De plus, Oracle a créé un fichier de mots de passe pour ce service (dans "OracleHome"\database):

Nom	Taille	Type	Date de modification
archive		Dossier de fichiers	17/10/2009 16:28
core_info_27...	1 Ko	Document texte	26/11/2009 12:14
hc_info	2 Ko	Fichier DAT	25/11/2009 17:53
hc_orcl	2 Ko	Fichier DAT	17/10/2009 17:51
oradba	31 Ko	Application	17/10/2009 16:18
PWDInfo	3 Ko	Fichier ORA	25/11/2009 16:00
PWORD	2 Ko	Fichier ORA	26/11/2009 10:41
SPFILEORCL	3 Ko	Fichier ORA	17/10/2009 16:38

3 - Créer le fichier de paramètres

L'instruction de création de la base utilisera un fichier de paramètres. Ce fichier est un fichier binaire, appelé SPFILE. Ce fichier binaire est généré à partir d'un fichier ASCII INIT.ORA que nous devons créer.

Création des répertoires :

a : Le répertoire. \$ORACLE_BASE\admin, créer un nouveau sous répertoire <BTEST>.

b : Le répertoire. \$ORACLE_BASE\oradata, créer un nouveau sous répertoire <BTEST>.

c : Créer les sous répertoires suivants dans le \$ORACLE_BASE\admin\<BTEST> : cdump, bdump, udump et pfile.

L'installation d'Oracle place un fichier INIT.ORA type "OracleHome"\admin:

🔪 Ouvrir le fichier le fichier de paramètre existant, et le Sauvegarder sous le nom init.ora dans le dossier :
C:\app\vaio\admin\Btest\pfile\init.ora:

🔪 Remplacer le nom **Orcl** par **Btest**

```

db_block_size=8192
control_files=('C:\app\...\Btest\control01.ctl', 'C:\app\...\Btest\control02.ctl')
db_recovery_file_dest=C:\app\vaio\Btest\flash_recovery_area
db_recovery_file_dest_size=2147483648

open_cursors=300
diagnostic_dest=C:\app\vaio
memory_target=424673280

db_domain=""
db_name=btest
processes=150
undo_tablespace=UNDOTBS1

dispatchers="(PROTOCOL=TCP) (SERVICE=btest)"
#####
# Security and Auditing
#####
audit_file_dest=C:\app\...\Btest\adump
audit_trail=db
remote_login_passwordfile=EXCLUSIVE

```

4 - Démarrer l'instance

🔪 retourner dans la ligne de commande DOS et lancer Sql/Plus en tapant:

sqlplus /nolog

🔪 Connecter-vous en tant que SYS

connect / as sysdba

« **CREATE SPFILE** », créer le fichier des paramètres serveur.

🔪 démarrer l'instance en tapant:

Create spfile from pfile='C:\app\...\Btest\pfile\init.ora';

taper par la suite

Startup Nomount

5 – Création de la base de données

Nous exécutons CREATE DATABASE avec les paramètres de création

- Nous créons les fichiers système, puis éventuellement nos tables
- Nous tapons les commandes à exécuter dans un fichier ASCII construit avec l'éditeur de texte, et sauvegardons le texte dans un fichier ayant n'importe quel nom (par exemple the_script.txt) dans Sql/Plus nous invoquons le script en tapant:

Par conséquent:

🔪 lancer votre éditeur préféré (Notepad, par exemple) et taper le script suivant:

```

Set Echo Off
Spool 'C:\app\...\Btest\script\echo_create_database.txt'

CREATE DATABASE btest
  MAXINSTANCES 8
  MAXLOGFILES 56
  MAXLOGMEMBERS 3
  LogFile
    group 1 'C:\app\...\Btest\redo1.log' size 10M reuse,
    group 2 'C:\app\...\Btest\redo2.log' Size 10M reuse
  MAXDATAFILES 1024
  DataFile 'C:\app\...\Btest\system.dbf' Size 50M  autoextend on next 10M maxsize unlimited
  extent management local
  sysaux datafile 'C:\app\...\Btest\sysaux.dbf' Size 10M  autoextend on next 10M maxsize unlimited
  UNDO TABLESPACE UNDOTBS1
    datafile 'C:\app\...\Btest\the_undo.dbf' Size 10M
  default temporary tablespace temp
    tempfile 'C:\app\...\Btest\the_temp.dbf' Size 10M
;
Shutdown
Spool Off

```

✚ sauvegarder ce texte sous: **C:\app\...\Btest\script\create_database.txt**

✚ lancer le script de création en tapant

@ 'C:\app\...\Btest\script\create_database.txt

Vérifier l'état de la base de données et que les fichiers de la base de données ont été créés physiquement et logiquement

SQL> select name, open_mode from v\$database;

NAME OPEN_MODE

BTEST READ WRITE

SQL> alter database mount;

SQL> show parameter control_file

Si la création échoue, visualiser le fichier d'alertes (show parameter user_dump)

- Arrêter l'instance
- Supprimer les fichiers créés pendant la tentative
- Relancer l'ordre create database de nouveau

6 - Création des tables système

La base créée est vide, et il faut exécuter des scripts qui installent des tables et programmes systèmes dans la base:

- catalog.sql crée le dictionnaire de données
- catproc.sql créé les structures pour PL/SQL.

Catalog.sql fait 820K . Il vaut donc mieux lancer ce script en mode ECHO OFF. L'exécution dure une bonne vingtaine de minutes.

Donc: lancer Notepad et taper le script suivant:



```

Set Echo Off
Spool 'C:\app\vaio\admin\Btest\script\echo_catalogs.txt'
@ C:\app\...\product\11.1.0\db_1\RDBMS\ADMIN\catalog.sql
@ C:\app\...\product\11.1.0\db_1\RDBMS\ADMIN\catproc.sql
Spool Off

```

✚ sauvegarder ce texte sous: **C:\app\vaio\admin\Btest\script\catalogs.txt**

✚ retourner dans Sql/Plus et lancer ce script:

@ 'C:\app\...\Btest\script\catalogs.txt

← Sql/Plus lance ces deux scripts (20 minutes)

Afficher le nom des utilisateurs de la base de données.

7 - Création de TableSpace

Nous créons à présent un fichier pour nos données (TableSpace):

✚ lancer Notepad et taper le script suivant:

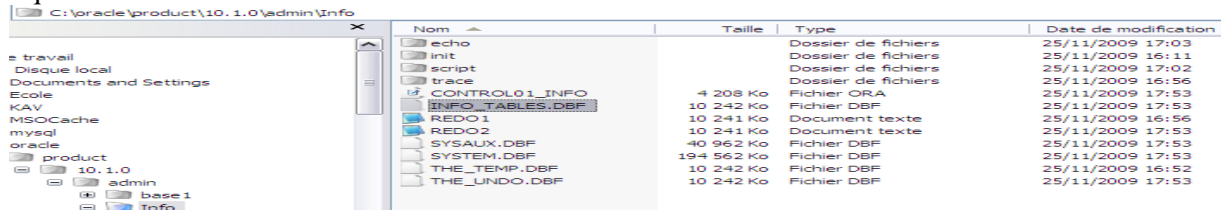
```
Set Echo Off
Spool 'C:\app\...\Btest\script\echo_create_tablespace.txt'
grant connect to system identified by manager;
connect system/manager
create tablespace TEST_tables
datafile 'C:\app\...\Btest\TEST_tables.dbf' size 10m extent management local;
Spool Off
```

✚ sauvegarder ce texte sous: **C:\app\vaio\admin\Btest\script\create_tablespace.txt**

✚ Retourner dans Sql/Plus et lancer ce script :

@ C:\app\...\Btest\script\create_tablespace.txt

✚ Sql/Plus crée le fichier de données



Nom	Taille	Type	Date de modification
echo		Dossier de fichiers	25/11/2009 17:03
init		Dossier de fichiers	25/11/2009 16:11
script		Dossier de fichiers	25/11/2009 17:02
trace		Dossier de fichiers	25/11/2009 16:56
CONTROL01.INFO	4 208 Ko	Fichier ORA	25/11/2009 17:53
INFO_TABLES.DBF	10 242 Ko	Fichier DBF	25/11/2009 17:53
REDO1	10 241 Ko	Document texte	25/11/2009 16:56
REDO2	10 241 Ko	Document texte	25/11/2009 17:53
SYSAUX.DBF	40 962 Ko	Fichier DBF	25/11/2009 17:53
SYSTEM.DBF	194 562 Ko	Fichier DBF	25/11/2009 17:53
THE_TEMP.DBF	10 242 Ko	Fichier DBF	25/11/2009 16:52
THE_UNDO.DBF	10 242 Ko	Fichier DBF	25/11/2009 17:53

8 - Création de Table de Démonstration

A titre de vérification, nous allons créer une table utilisateur, et vérifier son fonctionnement. Pour cela, nous utilisons le script scott.sql qu'Oracle a fourni pour créer le schéma d'utilisateur scott.

lancer Notepad et taper le script suivant:

```
Set Echo Off
Spool 'C:\app\...\Btest\script\echo_demo_table.txt'
connect / as sysdba
@ C:\app\...\product\11.1.0\db_1\RDBMS\ADMIN\scott.sql
select * from dept;
Spool Off
```

✚ sauvegarder ce texte sous **C:\app\...\Btest\script\demo_table.txt**

retourner dans Sql/Plus et lancer ce script:

@ C:\app\...\Btest\script\demo_table.txt