

9. Exercices

■ Exercices A : chap.2 & 3

2.1 Quel est le nom des instances actives sur votre machine host ?

2.2 Découverte de l'arborescence d'Oracle et de quelques fichiers sensibles

a) localiser les fichiers suivants et indiqués leur rôle :

- . Cour_ALRT.log (NT) ou alert_COURS.log (Unix)
- . catalog.sql
- . courLGWR.trc (NT) ou lgwr_cour.trc (Unix)
- . ora*.trc (NT) ou ora_*.trc(Unix)
- . Oracle.exe (NT) ou oracle (Unix)

- . initCOUR.ora (NT) ou initCOURS.ora (Unix)

- . Dbshut, dbstart et oratab (Unix)

2.3 paramètres d'initialisations

- a) donner les différentes possibilités pour visualiser les paramètres d'initialisation d'une instance active.
- b) lister via un ordre SQL uniquement les paramètres modifiés

2.4 Le dictionnaire de données d'Oracle

- a) donner la liste des vues du dictionnaire de données d'Oracle(dict) triée par nom

2.5 Donner la liste des utilisateurs (v\$session : username, serial#, sid, paddr, program, terminal, ident) connectés sur votre instance courante. *Ident* est une colonne rajoutée pour identifier de façon explicite les process background et les process utilisateurs. S'il s'agit d'un process utilisateur, ident vaut "PROCESS BACKGROUND" sinon il vaut "PROCESS UTILISATEUR".

2.6 Déterminer la taille de la SGA

- a) taille globale (v\$sga)
- b) taille détaillée (v\$sgastat)

9. Exercices

■ Exercices A : chap.2 & 3 (suite)

2.7 Indiquer si la taille des buffers de données est suffisante ou non en calculant le ratio suivant :

$$R = (1 - (\text{Physical reads} / (\text{db block gets} + \text{consistent gets}))) * 100$$

Dans le cas ou elle n'est pas suffisante, proposer des mesures pour améliorer la situation.

Notes :

a) Nom des statistiques intéressantes :

- db block gets : nombre de blocs lus
- consistent gets : nombre de blocs recherchés dans le buffer RBS ou le RBS
- physical reads : nombre de blocs lus sur disque.

b) db block gets + consistent gets = nombre total de blocs lus en mémoire ou sur disque

2.8 Déterminer la taille de la PGA associée à votre session en mode server dédié (v\$session, v\$sesstat, v\$statname).

2.9 Donner le texte de la requête lancée par chaque utilisateur connecté sur la base (v\$sqlarea, v\$session).

9. Exercices

■ Exercices B : chap. 3 & 4

3.1. Créer d'une base sous Oracle 9i

Le nom de la base sera : DBCOURS.CERAM.FR Le nom de l'instance DBCOURS.

Les variables d'environnement :

ORACLE_HOME=/oracle/oracle92

ORACLE_BASE=/oracle

Les fichiers de la base doivent être localisés sous \$ORACLE_BASE/oradata/DBCOURS dans l'arborescence suivante :

disk1 localisation du 1er fichier de contrôle et du 1er fichier de chaque groupe de redo log

disk2 localisation du 2e fichier de contrôle et du 2e fichier de chaque groupe de redo log

disk3 localisation du 3e fichier de contrôle et du 2e fichier de chaque groupe de redo log

tssys localisation des fichiers du tablespace système

tsundorbs localisation des fichiers du tablespace RBS des rollback segments

tstemp localisation des fichiers du tablespace TEMP temporaire

tstools localisation des fichiers du tablespace TOOLS ou seront stockés les objets des outils oracle

tsusers localisation des fichiers du tablespace de travail USERS

archive localisation des archives

Nota : s'il ya d'autres fichiers de données à localiser, définir de façon pertinente leur localisation

Les fichiers de trace doivent être localisés sous

\$ORACLE_BASE/admin/DBCOURS dans l'arborescence suivante :

pfile localisation du fichier des paramètres d'initialisation

bdump localisation des fichiers de trace des process background

udump localisation des fichiers de trace des process serveurs

core localisation des fichiers core

Le fichier des paramètres serveurs pfile doivent être localisé sous \$ORACLE_HOME/database

9. Exercices

■ Exercices B : chap. 3 & 4

3.1. Créer d'une base sous Oracle 9i (suite)

Gestion de la mémoire (valeurs minimales)

- cache des données : 16Mo
- shared pool : 32Mo
- taille de la pga : 25Mo
- java pool : 5 Mo (à augmenter à au moins 20 Mo si l'on pratique Java dans la base)
- large pool : 2 Mo
- zone de tri : 524288 octets

Jeu de caractères de la base

- utiliser unicode (al32utf8)

Les rollbacks segments doivent être gérés automatiquement

Mode de fonctionnement par défaut : serveur dédié

Modifier les paramètres d'initialisations suivants :

- NLS_DATE_FORMAT=DD-MM-YYYY
- TIMED_STATISTICS=TRUE
- LOG_ARCHIVE_START=TRUE
- OS_AUTHENT_PREFIX=SE\$
- ORACLE_TRACE_ENABLE=true
- RESOURCE_LIMIT=true

3.2. Modification de la structure d'une base

1. Ajouter un fichier redo log à chaque groupe de Redo log.
2. Ajouter un fichier de contrôle
3. Vérifier la présence de ces fichiers au niveau OS et Oracle

9. Exercices

■ Exercices C : chap. 4

4.1. Lister l'ensemble des fichiers de données, de contrôle et redo log composant la base DBCOURS

4.2 création de tablespaces

Vous avez reçu la mission de mettre en place une application de gestion d'une base aérienne. Les tables de cette base (PILOTE-AVION-VOL) ont une volumétrie de 5 Mo et les indexes une volumétrie de 2 Mo. Il est nécessaire de séparer les données et les indexes. Dimensionner pour cela deux tablespaces. Ces tablespaces doivent être gérés localement.

Pour les tables :

Nom : TS_TAB_AIRBASE_x

Nom fichier : ts_tab_airbase₁

Localisation des fichiers :

\$ORACLE_BASE/oradata/DBCOURS/TSTAB_x/

Pour les indexes

Nom : TS_IND_AIRBASE_x

Nom fichier : ts_ind_airbase₁

Localisation des fichiers :

\$ORACLE_BASE/oradata/DBCOURS/TSIND_x/

Lister les informations sur vos tablespaces

Nota :

a) x est votre numéro de binôme

4.3. Rollbacks segments

Déterminer le mode de gestion des segments d'annulations de votre base.

Tenter de créer un rollback segment :

Nom : Rbs_x Note : x numéro binôme

Capacité : 1 Mo maximum.

Localisation : dans le même tablespace que les autres rollbacks segments existants sauf system

9. Exercices

■ Exercices C : chap. 4

4.4 Modifier le schéma de la base aérienne (script TPADM_airbase1.sql) comme suit :

- localiser toute les tables dans le tablespace TS_TAB_AIRBASEx
- localiser tous les indexes y compris ceux créés implicitement dans le tablespace TS_IND_AIRBASEx
- apdater les paramètres de stockage de chacune des tables PILOTE_x (1Mo), AVION_x (500K), VOL_x (2 Mo). Ces tables peuvent croître de 10 à 50 % par an.
- les paramètres de stockage des indexes seront fixés à 10% de ceux de la table concernée.

4.5 Créer le schéma de la base aérienne sous le compte SYSTEM

4.6 Insérer quelques enregistrements dans la table PILOTE_x. La transaction doit se dérouler dans le rollback segment RBS_x s'il a pu être créé. Vous devez mettre en évidence la présence de la transaction dans ce rollback segment si l'étape précédente à été concluante.

4.7 Lister les informations sur le segment (dba_segments) et les extensions (dba_extents) de la table PILOTE_x

4.8 Donner la somme, le minimum, le maximum et la moyenne de l'espace libre en nombre de blocs par tablespace

4.9 Supprimer le schéma de la base aérienne précédemment créé.

9. Exercices

■ Exercices D : chap. 5

- 5.1. Créer un utilisateur oracle ayant votre nom XXXx avec les caractéristiques suivantes :
 - tablespace par défaut : users
 - tablespace temporaire : temp
 - mot de passe : à votre choix

- 5.2. Connectez vous comme étant XXXx. Que constatez vous ?

- 5.3. Etant System, Affectez le privilège CREATE SESSION à l'utilisateur et reconnectez vous comme étant XXXx. Si la connexion a réussi, tentez de créer la table PILOTE_x. Que constatez vous ? Remedier au problème.

- 5.4. Etant System, retirer les privilèges accordés à XXX et supprimez la table qu'il à créée

- 5.5. Etant System, créer un rôle RL_XXX qui permettent à l'utilisateur XXX de se connecter, de créer des tables, d'insérer des lignes dans ses tables et de créer des clusters

- 5.6. Créer un rôle RL_ADMIN_x qui vous permet de supprimer, consulter et alterer des tables dans n'importe quel schéma. Ce rôle doit être créé avec un mot de passe.

- 5.7. Affectez RL_XXX et RL_ADMIN_x à l'utilisateur XXX. Attention le rôle RL_ADMIN_x ne doit pas être un rôle par défaut

- 5.8. Connectez vous comme étant XXX. Tentez de créer la fameuse table PILOTE_x dans votre schéma.
Ensuite utilisez votre rôle RL_ADMIN_x pour supprimer la table pilote de votre voisin.

- 5.9. Connectez vous comme étant System. Rétirez le privilège de création d'une table à XXX. Injectez dans le schéma de XXX les tables VOL_x et AVION_x manquantes. Y insérer des enregistrements

- 5.10. Etant System, tentez de donner à votre voisin, le droit de consulter, insérer et modifier les tables de XXX. Que constatez vous ? Faire la même chose étant XXX. Demander à votre voisin de tester.

- 5.11. Etant System, créer un profile PF_XXX qui permet à XXX de ne pas pouvoir créer plus de 2 sessions et d'être déconnecté si pendant deux minute, il n'a rien fait. Testez et vérifier les informations dans le dictionnaire

9. Exercices

■ Exercices E : chap. 6

Partie 1: SAUVEGARDE ET RESTAURATION MANUELLE

6.1. Opérer une sauvegarde complète de votre base. Attention, la base doit être arrêtée proprement.

Identifier pour cela, de façon efficace(en un seul ordre SQL) la liste des fichiers à sauvegarder

6.2. Passer votre base en mode sans archive

6.3. Provoquer de l'activité dans la base en ajoutant un fichier (100K) dans votre tablespace TS_IND_AIRBASEx. Insérer et valider des enregistrements dans votre table PILOTEx. Prendre une sauvegarde complète

6.4. Modifier avec votre éditeur favori un de vos fichiers de contrôle (à cette fin, si nécessaire, arrêter la base) et tentez de répartir. En cas de problème, remédier à la situation sans se servir de la sauvegarde. Vérifier la présence de vos objets

6.5. Provoquer une nouvelle activité en insérant de nouveaux enregistrements dans la tablePILOTEx. Provoquer plusieurs fois un changement de groupe de Redo Log

6.6. Modifier avec votre éditeur favori, le fichier de données de votre tablespace TS_TAB_AIRBASEx (à cette fin, si nécessaire, arrêter la base) et tentez de répartir. En cas de problème, remédier à la situation.

6.7. Répartir du backup opéré en en 1. Assurez vous que vous êtes bien en mode avec archive.

Testez que les archives sont bien générées

6.8. Ajouter un fichier (200K) dans votre tablespace TS_TAB_AIRBASEx. Provoquer la création d'une extension dans ce fichier. Prendre un backup de votre fichier de contrôle. Provoquer une forte activité dans ce tablespace

9. Exercices

■ Exercices E : chap. 6

6.9. Détruire par erreur le fichier créer en 7 (si utile, base arrêtée). Tenter de redémarrer. En cas de problème, remédier à la situation et vérifier la présence de vos objets

6.10. Détruire le tablespace TS_TAB_AIRBASEx. Enlever physiquement les fichiers de TS_TAB_AIRBASEx. Provoquer une activité dans la base sur les tables de votre utilisateur XXX.

6.11. Vous vous apercevez maintenant que vous n'auriez du supprimer TS_TAB_AIRBASEx, vous ne voulez pas non plus perdre l'activité engendrée en 9 sur les tables de XXX. Prendre toutes mesures qu'il faut pour remédier à la situation.

Partie 2: SAUVGARDE ET RESTAURATION AUTOMATIQUE (RMAN)

6.12 Configurer RMAN afin de pouvoir effectuer des sauvegardes et restauration automatique. Vous devez être connecté au catalogue RMAN. S'assurer que la base est bien en mode avec Archive et qu'elle est enregistré dans le catalogue

```
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE DISK TO 'f:\ora_cf%F';  
CONFIGURE RETENTION POLICY TO RECOVERY WINDOW OF 15 DAYS;  
CONFIGURE BACKUP OPTIMIZATION ON;  
CONFIGURE DEFAULT DEVICE TYPE TO DISK;  
CONFIGURE CONTROLFILE AUTOBACKUP ON;
```

```
REGISTER DATABASE;  
REPORT SCHEMA; -- visualise la structure de la sauvegarde
```

6.13 Effectuer une sauvegarde de la base entière avec RMAN. L'ensemble des fichiers composants la base est concerné.

6.14 Simuler la restauration et le recouvrement d'un fichier de contrôle

6.15 Simuler la restauration et le recouvrement d'un fichier de données

9. Exercices

■ Exercices F : chap. 7

7.1. Exporter en mode batch, les données de l'utilisateur XXX

7.2. Ajoutez dans la table VOLx localisé dans TS_TAB_AIRBASEx, la colonne suivante :
remarq varchar2(2000);

Modifier les enregistrements existants dans cette table afin d'y ajouter des commentaires.
Faire en sorte qu'il y ai des chaînages.

Pour être sûr que les chaînages ont été générés, lancer les actions suivantes :

```
svrmgr>ANALYZE TABLE VOLx COMPUTE STATISTICS;
```

```
svrmgr> SELECT table_name, chain_cnt FROM dba_tables where table_name='VOLx';
```

7.3. Opérer un export de toutes les tables présentent dans le tablespace TS_TAB_AIRBASEx.

7.4. Profiter de l'occasion pour réorganiser ce tablespace et éliminer les chaînages sur VOLx.
Vérifier l'effet de vos actions. Vous utiliserez bien sûr IMPORT pour réinjecter les données
dans TS_TAB_AIRBASEx. Utilisez à votre guise OEM ou la console.

7.5. Utiliser Entreprise Manager afin de visualiser les données de la base (tablespaces,
utilisateurs, table, index, ...). Intéressez vous plus particulièrement aux objets qui ont été créés
par vous.

7.6. Familiarisez vous avec SQLLOADER, en traitant l'exemple 5 du cours chap. 7.2. Ce
travail doit être fait depuis OEM ainsi que la création des tables sous-jacentes

9. Exercices

■ Exercices G : chap. 8

8.1 Option procédurale : Utilisation de trigger pour l'audit

Création d'un trigger qui permet (lors d'une suppression, modification ou insertion) d'insérer dans une table journal, l'état avant et après d'une ligne de la table EMP.

```
CREATE TRIGGER audit_employe
  AFTER INSERT OR DELETE OR UPDATE ON emp FOR EACH ROW
  BEGIN
    -- La raison du déclenchement du trigger doit être passée en paramètre
    -- à la fonction ci-dessous.
    -- AUDITPACKAGE.SET_REASON(texte_de_la_raison_d'audit)

    /*
    IF auditpackage.reason IS NULL THEN
      raise_application_error(-20201, 'Une raison doit être donnée'
        || ' grâce à AUDITPACKAGE.SET_REASON('
          texte_de_la_raison_d'audit)');
    END IF ;
    */
    -- Si la condition précédente est remplie alors ...
    INSERT INTO audit_employe VALUES
      (:old.EMPNO, :old.ENAME, :old.JOB, :old.MGR, :old.HIREDATE,
      :old.SAL, :old.COMM, :old.DEPTNO, :new.EMPNO,
      :new.ENAME, :new.JOB, :new.MGR, :new.HIREDATE,
      :new.SAL, :new.COMM, :new.DEPTNO,
      'Mise a jour dans la table emp' , user, sysdate);
    --auditpackage.reasion, user, sysdate);

  END;
```

/

Note :

- vous devez d'abord créer la table *audit_employe* (page suivante) dans votre schéma.

9. Exercices

■ Exercices G : chap. 8 (suite)

8.1 Option procédurale (suite)

```
CREATE TABLE audit_employe(  
  EMPNO          NUMBER(4) ,  
  ENAME         CHAR(10),  
  JOB           CHAR(9),  
  MGR          NUMBER(4),  
  HIREDATE     DATE,  
  SAL          NUMBER(7,2),  
  COMM        NUMBER(7,2),  
  DEPTNO      NUMBER(2) ,  
  NEW_EMPNO   NUMBER(4) ,  
  NEW_ENAME   CHAR(10),  
  NEW_JOB     CHAR(9),  
  NEW_MGR     NUMBER(4),  
  NEW_HIREDATE DATE,  
  NEW_SAL    NUMBER(7,2),  
  NEW_COMM   NUMBER(7,2),  
  NEW_DEPTNO NUMBER(2) ,  
  AUDIT_REASON VARCHAR2(40),  
  UTILISATEUR VARCHAR2(25),  
  DAT_EVENT   date) ;
```