

Youtube lasse9

vendredi 3 juillet 2020 13:44

[Tutoriel Oracle DBA1 \(1\) : La société Oracle et Oracle Database.](#)



DBA1 :

- Larry Ellison is the boss of Oracle
- Oracle DB (RDBMS) (main product of Oracle)
- Version 8: object database support
- Oracle is multi-platform

DBA2

- Entreprise : all functionalities
- Standard : 4 CPUs, limited functionalities
- Personnalisé : windows only, for devs
- Express : free, single processor machines
 - 4Go/11Go size limit, SGA < 1Go
- Autres : ...

DBA3

- Certifs lvls
 - DBA1 : arch, admin..
 - DBA2 : backup, restaurations
 - Turning : perf

DBA4

- Oracle vs others
 - MySQL : opensource
 - Postgres : opensource
 - SQL server : microsoft
 - Can't know what happen inside in details
 - Closed app
 - Windows only
 - DB2 : ibm
 - Same as sql server (Zos, Aix...)
 - Sybase : sybase
- Oracle
 - Can know details of execution in oracle
 - Open application
 - multiplatform

DBA5

- Oracle server (the main installation)
 - Instance : ensemble de processus et zones mémoires (RAM) qui permettent de gérer la base de données
 - Base de données : ensemble de fichiers dans le disque, contenant les informations sur les données les journaux le

contenu

Comme tous les programmes :

CPU : traitement et calcul

RAM : stocker les données temporaires

Disque : données permanent

DBA6

Oracle around 4 things : files, ram, processes, programmes

DBA7

Physique : unités réel existe dans le disque (fichiers)

Logique : c'est une notion pour organiser les unités physique (dossier)

DBA8

Client

Oracle client

Interface to access database, app or program for sql requests

package

Server

Real deal

Oracle server (RAM, disque, base de données (1 or more))

Connexion : you need a package to create a connexion, machine with oracle client, oracle server

DBA9

Package :

How to connect to oracle server

Package/paquet

IP (or machine name) / identify the machine

Port by default 1521 / identify the process of oracle server

SID (system id, each db in oracle server have its SID) / identify the DB

Username/password / identify the user

La requête

DBA10

Nom machine -> DNS / fichier host -> IP

DBA11

Les moyens de transport

Transférer les données vers oracle server

Processus qui font le transport des paquets entre les composants

c'est lui qui transfère les données depuis oracle client vers le serveur

Exemples

Processus client : créer par oracle

ODBC : installé par défaut sur windows 10 (il se base aussi sur le processus client)

JDBC : driver de Java

DBA12

Processus client :

Oracle client déclenche le processus client

c'est un moyen de transport

Il recherche sur la machine puis le port

--

Il crée un tunnel de communication lors la première connexion

Le processus client travaille une seule fois, dans la première connexion

Listener (only if connexion distance)

Le listener reçoit la requête depuis processus client

Les tâches de Listener

Il ouvre le port

Il écoute

Vérifier l'existence de la BD

c'est lui qui assure toutes les connexions à distance

DBA13

Schéma



Server arch

RAM -> SGA & PGA

SGA :

Composants...

PGA : single zone

DBA14

SGA : system global area

SGA sert comme la partie mémoire où toutes les opérations de la base de données ont se produire

C'est un espace mémoire dans le quelle exécute ses traitements

Elle est allouée au démarrage de l'instance et libérée lors de son arrêt

Elle inclut 6 zones mémoires principale

Exemple des opérations :

Vérifier le syntaxe

Exécuter une requête

c'est un espace mémoire partagé par tous les processus de l'instance

Par exemple si plusieurs utilisateurs se connectent en même temps, ils partagent tous le SGA

PGA : Program global area

Cette mémoire est associée à un processus

Cet PGA sert à temporiser les données que manipule le processus, toujours dans un souci d'optimisation

Celle-ci n'est pas partagée, elle est seulement utilisée par des processus serveur ou d'arrière -plan

Elle est allouée lors de démarrage du processus et libérée lors de l'arrêt du processus

DBA15

Disque : persistance

Exemples

Fichier d'installation

Fichier de stockage

Fichier de paramètres

Processus en arrière-plan

Processus qui sont en arrière-plan, il font d'ers taches pendant l'exécution des requêtes

Assurer le fonctionnement interne du SGBD Oracle (gestion de la mémoire, écriture dans les fichiers, ...)

Exemple : LGWR, CRU ...

Programmes

SQL PLUS : se connecte à une base de données

OUI : Oracle Universal Installer (server install)

DBCA : DataBase Creation Assistant (database creation)

DBA16

Data types in oracle :

Client	
Code	Nom
U01	Abdessamad
U02	Saâd
...	...

Metadata : données de dictionnaire

table name and columns

Ce sont des données pour la structure de base de donnée (table, view, champs, ...) ou les informations qui identifient les utilisateurs

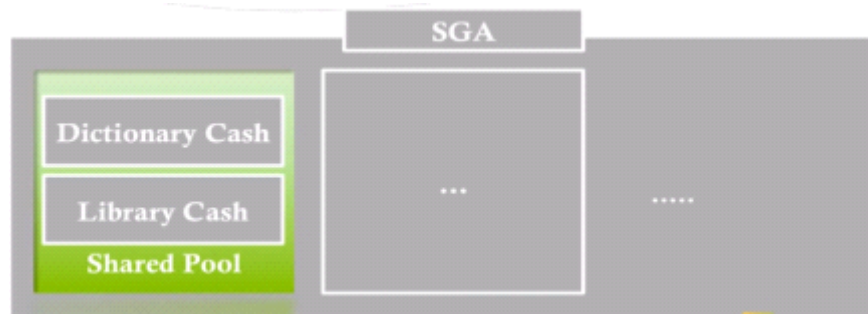
Elle contiennent toutes les informations de toutes les composantes logiques et physiques de la base de

données ainsi que l'instance
 User data
 Rows
 Ce sont des données qui se sont stockées dans une table créée par un utilisateur

DBA17

SGA :

Shared pool :



Zone partagé

Shared pool est la partie du SGA qui est utilisé par tous les utilisateurs

Il contient les requêtes SQL les plus récemment exécutées et l'information du dictionnaire de données la plus récemment utilisée

Contient les informations qui peuvent être réutilisées par les différents utilisateurs (requêtes SQL, programme pl/sql, dictionnaire de données, curseurs)

d'une manière générale, zone pour mémoriser et analyser et traiter les requêtes SQL

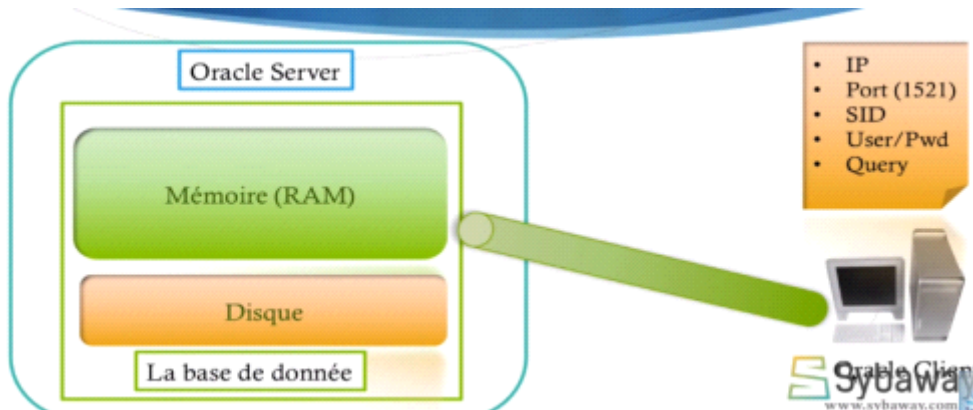
Elle est utilisée pour partager les informations sur les objets de la base de données ainsi que les droits et privilèges accordés aux utilisateurs

il est composée de deux structures; le library cache et le dictionary cache

Library cache :

Dictionary cache :

DBA18



Processus server

Autre moyen de transport

c'est lui qui transfère le package depuis listener vers l'optimizer

Optimizer

Généralement c'est lui qui gère toutes les opérations dans la mémoire à l'aide des processus en arrière-plan

Tâches d'optimizer

Vérifier la syntaxe (if bad syntax it returns the request to oracle client)

Créer les plans d'exécution

Sélectionner le plan le plus court

Exécuter la requête

DBA19

Listener : assurer la connexion à distance

Optimizer

Vérifie la syntaxe dans la zone library cache

Il utilise les données de dictionnaire (Dictionary cache) Metadata

Dictionary cache :

c'est un composant mémoire dans SGA pour stocker les données de dictionnaire (meta data)

Utilisé dans la création de plan d'exécution

Ex : vérifier la syntaxe

Il contient tous les informations nécessaires pour le plan d'exécution

Le dictionary cache stocke toutes les informations nécessaires à l'analyse sémantique de la requête (table? Colonnes? Droits d'accès?)

DBA20

Library cache : contient pour chaque requête récemment exécutée trois informations :

Son texte

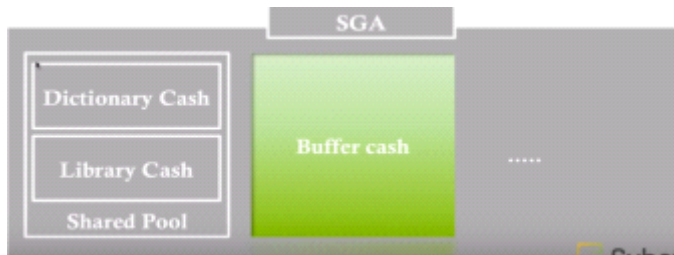
Sa compilation

Son plan d'exécution

Lorsqu'une requête existe encore dans le library cache, oracle ne perd pas son temps à la réexécuter

DBA21

Buffer cache



Metadata are in dictionary cache

Buffer cache :

Cette zone mémoire sert à stocker les blocs de données utilisés récemment

Ce qui signifie que lorsque vous allez lancer une première fois la requête Oracle, cette dernière va se charger de rapatrier les données à partir du disque dur

Mais lors des exécutions suivantes les blocs de données seront récupérés à partir de cette zone mémoire, entraînant ainsi un gain de temps

Contient tous les données transitant de ou vers la BD sous forme de blocks de données

Donc, il s'agit d'un cache permettant d'accélérer l'accès à certaines données de la base

DBA22-23

Plan d'exécution

C'est la manière à suivre pour exécuter la requête

L'optimizer crée plusieurs plans d'exécution

Lors de la création du plan d'exécution, l'OPTIMIZER a besoin de savoir les METADATA (dictionary cache)

Exemple de plan :

Utiliser le balayage complet -full scan (USERDATA in buffer cache)

Pointer directement à la ligne par Index

Oracle doesn't full execute the request when creating plans

Il crée aussi le coût de chaque plan

Il calcule le coût via une simulation d'exécution

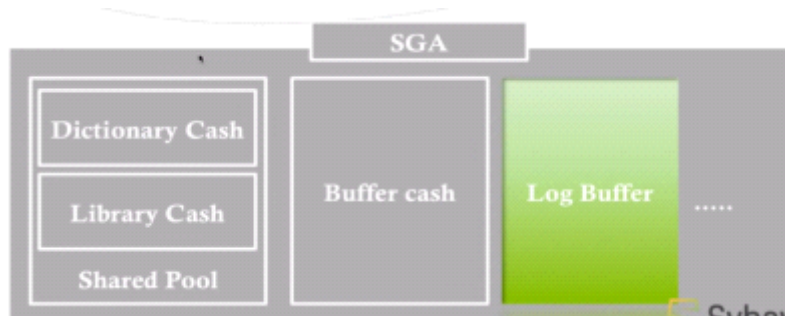
Son rôle est de voir le plan d'exécution pour les requêtes qui sont très lentes

Optimizer

Avant l'exécution il fait un log

Il ne fait que l'exécution de la requête (buffer cache)

DBA24



Log buffer :

C'est une zone mémoire dans SGA pour sauvegarder les logs

Il sauvegarde les logs avant l'exécution de la requête

Ce buffer conserve les traces des transactions, il contient les informations sur les transactions exécutées

Cette dernière est constituée de plusieurs vecteurs, chacun correspond à un bloc de données modifié (ancienne + nouvelle valeur)

Il permet de stocker les enregistrements Redo log, chaque modification correspond à une entrée Redo

Cette zone mémoire sert exclusivement à enregistrer toutes les modifications apportées sur les données de la

base

DBA25

PMON : Process monitor, process en arrière-plan

Son rôle est le nettoyage de tous les composants de BD

Garbage collector

Ce processus se charge notamment de libérer toutes les ressources acquises par un processus client, lorsque celui-ci se termine

Il se charge de nettoyage de la zone mémoire SGA

Il est aussi chargé de surveiller les processus serveurs et les processus dispatchers : si l'un d'eux s'arrêtait anormalement, le PMON se chargerait de libérer les ressources de processus et de le relancer

Exemple :

Le programme qui supprime les sessions directions après la déconnexion

Enlever les plans d'exécution qui sont par choisis par l'optimizer

Détection du plantage des processus utilisateurs

Algorithm used to clear old record from buffer cache

LRU : Least Recently Used

Ce programme est pour identifier l'élément dans le composant mémoire le moins utilisé, si la mémoire est pleine

Utilisé dans tous les composants de SGA

PMON supprime les éléments mentionnés par LRU

DBA26

Les processus

Il permettent une interaction entre les différentes composantes du serveur ainsi qu'avec les utilisateurs

Client -> serveur (client process work)

Serveur -> client (server process work)

Processus client & serveur

Les deux processus sont des moyennes de transport

Le retour de résultat se fait par la moyenne de transport : processus serveur, ni listener ni processus client, il passe par le tunnel

L'interaction entre le serveur et les clients se fait en réalité grâce à ces deux processus, chacun de son côté

Il y'a 3 types

Processus client ou utilisateur

Oracle client exécute ces processus

s'exécute au niveau client

Processus serveur

s'exécute au niveau serveur BD

Processus d'arrière-plan

Assure le bon fonctionnement du serveur

Maximisent la performance du serveur

Démarrent avec ou après (sur demande) le démarrage de l'instance

Certains peuvent être exécutés en n exemplaires

DBA27

Connexion vs session

connexion

Lorsque le client est lié au serveur, on parle d'une connexion

Session

Chaque client, après la première connexion, il crée une session dans la BD (lorsque l'utilisateur s'identifie, il ouvre une session)

La session s'identifie par : SID et Serial#

Plusieurs sessions peuvent être ouvertes en même temps

Elles ne doivent pas dépasser un nombre paramétré

DBA28

bla

DBA29

ye

DBA30

Les fichiers DBF

Ou data file, les fichiers données

Ces fichiers représentent la partie visible de la base de données

Chaque donnée liée aux applications (structure et contenu) est mémorisée dans ce type de fichier



DBA31

DBWR

DataBase Writer ou (DBWn)

Lors la création de plan d'exécution, OPTIMIZER a besoin de savoir les METADATA

Il recherche dans le composant Dictionary cash, s'il n'existe pas il demande à DBWR de récupérer les METADATA depuis le fichier DBF

C'est un programme qui se déclenche si les informations nécessaires ne sont pas dans le Dictionary cash

Il copie les données de Dictionary cash depuis les fichiers DBF

Il copie juste les données nécessaire, parce qu'il reçoit une requête détaillé

Ce processus chargé du transfert des blocks de données modifiés du buffer mémoire du SGA dans les fichiers disque de la base de données

Ce processus est aussi là pour vérifier en permanence le nombre de blocks libres dans le Buffer Cache afin de laisser assez de place disponible pour l'écriture des données dans le buffer (Il utilise aussi PMON avec LRU)

DBA32-33

Arch serveur

Oracle propose deux type d'arch, ces arch définissent la façon de comment oracle travail avec les clients

La différence est entre la com avec processus client et processus server

Un serveur BD est :

Soit en mode dédié (dedicated server)

Une relation un à un (one to one) est fait entre processus utilisateur et processus de serveur (1 processus server = 1 oracle client)

Donc un processus s'occupe d'un client unique (un seul client à la fois)

Pour les connexions < 1000 (recommandations mais on peut faire plus que ça selon la puissance de la machine)

Soit en mode partagé (shared server)

Un processus server s'occupe de plusieurs clients

Faire donc réduire l'utilisation de ressources de système

1 processus server = n processus client

La priorité selon la première connexion

Le mode par défaut est le mode dédié

Le processus server (PS) reçoit les requêtes utilisateur, les exécute et envoie le résultat

En mode dédié, on dédie un PS à chaque utilisateur

En mode partagé, on partage un PS pour un ensemble d'utilisateurs

DBA34

Redo log

Fichier de journalisation / log file

c'est un fichier dans le disque, pour sauvegarder physiquement le logs

Log : qui a fait quoi

Ce sont des fichiers qui permettent une restauration en cas de panne de base de donnée

Il stock les données en binaire, tu ne peux pas le modifier

La base de donnée sera stoppée si le fichier est supprimé

Oracle utilise un ensemble des Redo Log Files

Ces fichiers mémorisent toutes les transactions effectuées par les utilisateurs

On les appelle aussi les fichiers "journaux"

Il représentent un historique complet de toutes les commandes (amenant une modification) passées sur la base de données

L'utilité de tels fichiers est de pouvoir reconstruire les transactions de la base de données dans leur ordre adéquat, si une panne apparaît



DBA35

LGWR : LOG Writer

C'est un processus en arrière-plan et se déclenche pour copier les données depuis le log buffer à REDO log
Le processus LGWR est responsable d'écrire le contenu des buffers dans des fichiers journaux appelés fichiers Redo Log

Le LGWR écrit le contenu du Redo Log Buffer sur les fichiers journaux pour libérer de l'espace au PS

Le LGWR est déclenché par :

Presque 3 milli second

Lors d'un commit : écriture des entrées Redo relatives à la transaction confirmée

Si 1/3 de log buffer est plein



DBA36-37

Groups - Redo Log

Oracle a divisé les fichiers log en sous forme des groupes

Un groupe est une organisation logique, et n'existe pas dans le disque

Il l'utilise pour organiser les fichiers Redo Log

Chaque groupe contient plusieurs membres

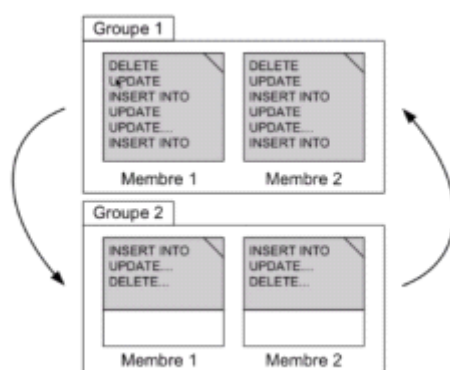
Au moins deux groupes

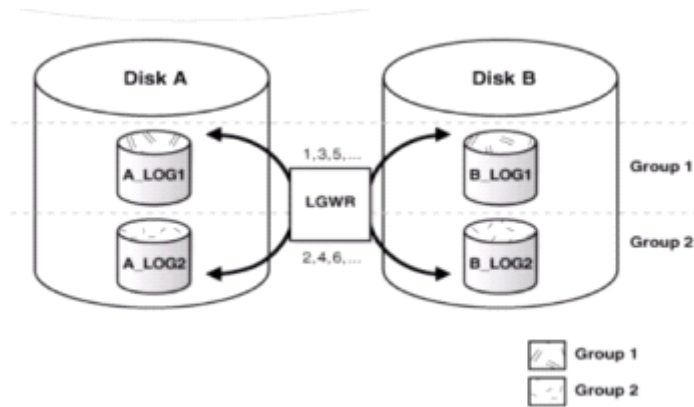
Pour chaque groupe on a au moins un seul membre

Membre = un fichier physique = un fichier redo log

Utilité d'avoir plusieurs membres est la sécurité de log en cas de panne de disques

Deux group min juste pour avoir possibilité de basculer entre au moins deux groupes si l'un des deux est plein





DBA38

Principe multiplexage et cyclique

Multiplexage :

L'écriture est multiplexée à l'intérieur d'un groupe

Groupe

Membre 1 : écrire block 1,4,7

Membre 2 : écrire block 2,5,8

Membre 3 : écrire block 3,6,9

Cyclique :

L'écriture est cyclique entre les groupes

Groupe 1 -> Groupe 2 -> Groupe 3

DBA39

LGWR copie le même log pour tous les membres (multiplexage)

Raisons :

Duplication

Sécurité

Analogie :

Group => dossier

Membre => fichier

Status du groups :

Active : les données / les blocks sur le Redo sont pas encore synchronisé avec le data file

Current : le logwriter est en train d'écrire les transactions

Inactive : Contraire d'active, il est déjà synchronisé est les données dans les fichiers DBF

Unused : N'est pas encore utilisé

DBA40

Switch entre les groups

On bien basculement

Le switch entre les groupes se fait automatiquement si le groupe est plein

On peut faire ça manuellement via une commande

Raison de Switch :

Si on veut supprimer un groupe

Switch, changé d'emplacement et puis supprimer

ARC se déclenche automatiquement s'il y a un groupe plein pour l'archiver

ARC

c'est le processus en arrière-plan responsable d'archivage

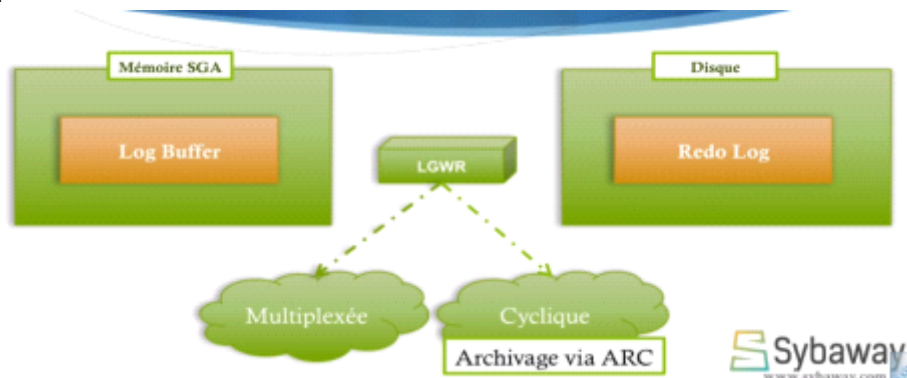
c'est lui qui archive les membres d'un groupe si le groupe est plein (membre = group)

Il met le fichier archivé dans un autre composant dans le disque qui s'appelle archive





DBA41



REDO LOG

- On garde le log pour deux raisons
 - Garder des traces pour ce voir ce qui a passé dans le BD
 - Modifications effectuées sur les données
- Restauration / Recouvrement
 - EN cas de crash du système
- La sécurité & la performance

LOG MINER

C'est le seul outil pour lire le contenu des fichiers REDOLOG

DBA42

Large Pool

C'est une zone mémoire dans laquelle le programme RMAN travail

RMAN :

- C'est le responsable de sauvegarde et restauration
- Il fait le copie et non pas la synchronisation
- Programme qui s'occupe des taches de copie et restauration
- Les détails de ce programme et comment il ce travail sont hors programme

Backup

- Composant dans la partie disque d'Oracle
- Le backup dans le disque se fait par RMAN



DBA43

Java Pool

La zone mémoire pour exécuter un code Java au sein d'oracle

À condition : il exécute les programmes Java qui sont développée par la même version de JDK qui existe dans Oracle

Stocke les objets et applications Java les plus récemment utilisés lorsque la machine virtuelle Java JVM optionnelle est utilisée (à vérifier)



DBA44

Les environnements de développement



DBA45

Streams Pool

Zone mémoires pour la gestion des flux

C'est l'espace mémoires dans lequel oracle Stream travail

Espace pour la synchronisation et non pas la restauration

La synchronisation : en temps réel

C'est le cache des données relatives à la queue de messages utilisées par Oracle

Oracle streams

Une technologie d'oracle pour la synchronisation et la copie en, temps réel entre la BD et une autre

Exemple : PROD et TEST

Une autre société a développé un autre outil pour gérer la synchronisation; il s'appelle GOLDEN GATE

Il n'est pas gratuit

Il se base sur Oracle Stream



DBA46

SGA Fixe

À noter que dans la SGA, il y a une petite zone mémoire appelée SGA fixe, elle inclut des infos sur:
l'état de la base
l'instance
Les verrous

Processus en arrière-plan

PMON : Process MONitor

DBWR : DataBase WRiter

LGWR : LoG WRiter

ARC : ARChive

Composants mémoire

DC : Dictionary Cash

LC : Library Cash

SP : Shared Pool

BC : Buffer Cash

LB : Log Buffer

LP : Large Pool

JP : Java Pool

SP : Streams Pool

DBA47

Télécharger le binaire d'Oracle

Installer Oracle Server via OUI (Oracle Universal Installer)

Création de base de données via DataBase Configuration Assistance (DBCA)

Se connecter à la base de données via SQL Plus

DBA48

DBA49

DBA50

DBA51

DBA52

DBA53

DBA54

DBA55

DBA56

Oracle environment variables

ORACLE_BASE : var d'env ou oracle stock tous ces produits (including the database)

ORACLE_HOME : Instance du produit oracle

ORACLE_SID : SID de la BD

DBA57

DBA58

Oracle whenever it creates a database, it creates 2 users with it, SYS and SYSTEM

Storage mechanism of the database :

File System (default)

Automatic Storage Management (ASM)

Raw devices (raw paritions)

DBA59

DBA60

Block : Smallest unit of storage for allocation and for I/O, it cannot be changed once a database is created

DBA61

DBA62

Users

Oracle crée obligatoirement par défaut pour chaque base de données deux utilisateurs administrateur :

SYS

Arrêter, démarrer la BD

Création, suppression de BD

SYSTEM

SYS est super admin que SYSTEM

On peut aussi créer d'autre utilisateurs

DBA63

La gestion de DB

SQL plus : ligne de commande

Enterprise Manager : Console Web

Autre outils : SQL Developer, Toad for Oracle

Type de connexion

Local ; dans la même machine

Sans passer par le Listener

À distant : un autre machine

Passer obligatoirement par le Listener

qcm blanc

vendredi 3 juillet 2020

18:34

SAUCE :

<https://usermanual.wiki/Document/OCAOCP20Oracle20Database2011g20AllinOne20Exam20Guide.1538343455/help>

https://www.academia.edu/6079107/Oracle_11_g_1z0-052_Admin_I

Chapter 4 Creating an Oracle Database

[https://www.studyblue.com/notes/note/n/oracle-sample-questions/deck/12808879?](https://www.studyblue.com/notes/note/n/oracle-sample-questions/deck/12808879?fbclid=IwAR1w38InKnRoNGNlaCuiWTW0noGoIASLeI9sH3kx-PkVSdBKVwMQasxwa0)

[fbclid=IwAR1w38InKnRoNGNlaCuiWTW0noGoIASLeI9sH3kx-PkVSdBKVwMQasxwa0](https://www.studyblue.com/notes/note/n/oracle-sample-questions/deck/12808879?fbclid=IwAR1w38InKnRoNGNlaCuiWTW0noGoIASLeI9sH3kx-PkVSdBKVwMQasxwa0)

Question 1

Pas encore

répondu

Noté sur 1,00

🚩 Marquer la question

To create a database, in what mode must the instance be? (Choose the best answer.)

Veillez choisir une réponse :

- ☐ a. Started in NOMOUNT mode
- ☒ b. Started in MOUNT mode
- ☐ c. Started in OPEN mode
- ☐ d. Not started

Correct Answer : a

Question 2

Pas encore

répondu

Noté sur 1,00

🚩 Marquer la question

What files are created by the CREATE DATABASE command? (Choose one or more correct answers.)

Veillez choisir au moins une réponse :

- ☒ a. The static parameter file
- ☒ b. The password file
- ☒ c. The online redo log files
- ☐ d. The controlfile
- ☒ e. The SYSTEM tablespace datafile
- ☐ f. The SYSAUX tablespace datafile
- ☐ g. The dynamic parameter file

Created files : Controlfile, online redo log, sysaux tablespace, system tablespace

Correct Answers : d, c, f, e

Question 3

Pas encore
répondu

Noté sur 1,00

Marquer la
question

When a session changes data, where does the change get written?

Veuillez choisir une réponse :

- ☒ a. Nothing is written until the change is committed
- ☐ b. The session writes to the database buffer cache, and the log writer writes to the current online redo log file
- ☐ c. To the data block on disk, and the current online redo log file
- ☐ d. To the data block in the cache, and the redo log buffer

Correct Answer : a

Question 4

Réponse
enregistrée

Noté sur 1,00

Marquer la
question

If a tablespace is created with this syntax:

```
create tablespace tbs1 datafile 'tbs1.dbf' size 10m;
```

which of these characteristics will it have? (Choose all correct answers.)

Veuillez choisir au moins une réponse :

- ☒ a. The datafile will autoextend, but only to double its initial size.
- ☒ b. The extent management will be local
- ☐ c. The file will be created in the DB_CREATE_FILE_DEST directory.
- ☐ d. Segment space management will be with bitmaps
- ☐ e. The datafile will autoextend with MAXSIZE UNLIMITED.

Correct Answers : c, d

Question 5

Pas encore
répondu

Noté sur 1,00

Marquer la
question

1. You issue this statement:

```
audit select on hr.emp by access;
```

but when you issue the command:

```
select * from hr.emp where employee_id=0;
```

no audit record is generated. Why might this be?

Veuillez choisir une réponse :

- ☐ a. The AUDIT_TRAIL parameter is set to NONE
- ☐ b. You are connected as SYS, and the parameter AUDIT_SYS_OPERATIONS is set to FALSE
- ☒ c. The instance must be restarted before any change to auditing comes into effect
- ☐ d. The statement did not access any rows; there is no row with EMPLOYEE_ID equal to zero

Correct Answer : a, the audit trail is set to NONE

Question **6**

Pas encore
répondu

Noté sur 1,00

🚩 Marquer la
question

The AWR is located in the SYSAUX tablespace. If you suspect that it is growing to such a size that it will fill the SYSAUX tablespace, what actions could you take to reduce the likelihood of this happening?

Veillez choisir une réponse :

- ☐ a. Increase the time between snapshots, so that fewer snapshots will be generated
- ☒ b. Relocate the AWR to a tablespace created specifically for storing it
- ☐ c. Adjust the scheduling of the automatic maintenance tasks, so that they will run less frequently
- ☐ d. Reduce the time between snapshots, so that less data will be generated by each one

Question **7**

Pas encore
répondu

Noté sur 1,00

🚩 Marquer la
question

The AWR is located in the SYSAUX tablespace. If you suspect that it is growing to such a size that it will fill the SYSAUX tablespace, what actions could you take to reduce the likelihood of this happening?

Veillez choisir une réponse :

- ☒ a. Increase the time between snapshots, so that fewer snapshots will be generated
- ☐ b. Relocate the AWR to a tablespace created specifically for storing it
- ☐ c. Reduce the time between snapshots, so that less data will be generated by each one
- ☐ d. Adjust the scheduling of the automatic maintenance tasks, so that they will run less frequently

Increasing the time between snapshots will reduce the number stored

Correct Answer : a

Résumé

samedi 4 juillet 2020 02:27

Rôle administrateur oracle :

- **Rôle organisationnel**
 - Définition du schéma conceptuel des données
 - Décision de qui peut accéder à quoi et comment
- **Rôle technique (niveau interne)**
 - Installation du SGBD et les outils associés
 - Création de la base de données et assurer son évolution
 - Gestion des privilèges d'accès
 - Amélioration des performances (implantation optimale)
 - Sécurité et cohérence des données
 - Échange de données entre la base et le monde extérieur

Rôle du SGBD Oracle :

- Définition et manipulation des données (Langage SQL/PLSQL, Procédures, triggers)
- Intégrité (contrainte du modèle relationnel)
- Confidentialité (Privilège, Rôle, Vue)
- Gestion de la Concurrency d'accès
- Gestion des transactions
- SGBD à architecture ouverte :
 - Portabilité d'Oracle (Puisqu'il est écrit en C => Indépendance vis à vis de l'environnement d'exécution)
 - Compatibilité aux normes (SQL92)
 - Distributivité d'Oracle (BD distribuée => Indépendance vis-à-vis du réseau de communication utilisé et de la localisation des données)

Services proposés par les applications ORACLE :

- Service méthodologique (génération de code SQL : DDL)
- Service de développement d'applications (ORACLE FORMS/REPORT, ...)
- Service d'aide à la décision
- Services destinés aux administrateurs de bases de données
- Service de communication (accès à une base de données depuis des applications clientes, Outils permettant la répartition des données)

Architecture ORACLE :

- Noyau
 - **fonctionnalités classiques** : Gestion de l'intégrité et de la cohérence des données, la confidentialité des données, sauvegarde et de restauration, accès concurrents
 - **Autres fonctionnalités** : Optimisation de l'exécution des requêtes, Gestion des accélérateurs, Stockage physique des données
- Dictionnaire de données
 - une base de données qui décrit de façon dynamique une base de données. Son contenu reflète l'image de la base à un instant donné (Objets de la BD, Utilisateur et privilèges, informations relatives à l'activité de la base de données (connexions, ressources utilisées, etc.))
 - Le dictionnaire de données est manipulé avec le langage SQL
- SQL
 - Elle joue le rôle d'interface entre le noyau et les différents outils d'Oracle, Tout accès à la base de données est exprimé en SQL
 - Interprétation des commandes SQL
 - Vérification syntaxique et sémantique
 - Décomposer les commandes en opérations élémentaires et les soumettre au noyau
 - Récupération du résultat et le transmettre à l'application qui a soumis la commande
- PLSQL
 - SQL + Structures de contrôle + Variables + Traitement des erreurs + Traitement des curseurs Oracle
- Services oracle
 - Services cités auparavant

Outils oracle :

- Outils de développement :
 - SQL PLUS : interface interactive à SQL et PL/SQL
 - Oracle Developer : conception et création d'application client/serveur
 - SQL*xxx : pré-compilateurs (exemple : C + SQL => C + fonctions)
 - Oracle Forms/Reports/Graphics (définit des requêtes qui génèrent des graphiques), Procédure Builder (gérer le code PL/SQL sur un serveur de bases de données)

Composants d'une BD Oracle et instance

- En générale :
 - Des fichiers physiques stockés sur le serveur
 - Des processus (threads) chargés en mémoire sur le serveur
 - Des espaces mémoires sur le serveur appelé SGA et PGA (System Global Area)
- En détail :
 - Base de données
 - Fichiers de données : Ils contiennent toutes les données de la base (les structures logiques et physiques y sont stockées (tables, vues,...), et le dictionnaire de données), extension : .dbf
 - Fichiers redo-log : Mémoisent l'historique de tous les ordres modifiant des données ou la structure de la base de données (LDD, LMD), Utilisés en cas de perte des fichiers DATABASE : redo, log
 - Fichiers de contrôle : Ils contiennent la description physique de la base (nom, chemin d'accès de chacun des fichiers, date et heure de création, information concernant la cohérence de la base, ...) Obligatoire au moment du démarrage de la base de données.
 - Fichiers d'administration : Les fichiers servant à paramétrer certains outils, Les différents fichiers de trace et d'alerte générés par Oracle
 - Instance :
 - Type de data :
 - User data (Enregistrements)
 - Meta data (Informations sur les données disponibles dans le dictionnaire de données)
 - L'intérêt générale des buffers/cache c'est d'éviter les lectures disques qui consomment trop de temps, et utilise des lectures des mémoires à la place qui sont moins coûteuses
 - Zone mémoire PGA :
 - Définition :
 - ◆ À PGA is a nonshared memory region that contains data and control information exclusively for use by an Oracle process
 - Exemple :
 - ◆ Each time a cursor is executed, a new run-time area is created for that cursor in the PGA
 - Zone mémoire SGA :
 - Définition :
 - ◆ Zone mémoire partagée par tous les utilisateurs Oracle (SGA), ensemble de buffers contenant des données utilisateurs et des données systèmes
 - Database Buffer cache (cache de blocks données manipulées)
 - ◆ Sert à sauvegarder les blocks de données user utilisés récemment (user data : enregistrements), ce qui fait gagner du temps lors du lancement d'une requête ou on a besoin des mêmes données à partir du disque
 - Redo Log Buffer (Cash de log)
 - ◆ contiennent des informations relatives aux modifications apportées à la base de données, Toutes les données avant leurs mises à jour, toutes les modifications effectuées sur ces données, la trace de toutes les transactions validées ou non encore validées (Block de données anciens et nouveaux), ces opérations sont faites par l'optimizer avant de vérifier si les données de la requête sont déjà disponibles dans le Buffer cache
 - Shared pool/Zone partage, aussi utilise pour stocker des infos sur les objets de la DB et les droits et privilèges des users + les infos du dictionnaire de données récemment utilisées (Un espace pour analyser et réutiliser les ordres SQL déjà exécutés), elle se compose de 2 parties
 - ◆ Library cache
 - ◇ Vérification de la syntaxe par l'Optimizer dans cette zone et cela depuis le dictionary cache, puis la compilation et la génération du plan d'exécution
 - ◇ Si la requête a été déjà exécutée dans le Library cache, oracle ne perd pas son temps à la réexécution
 - ◆ Dictionary cache (Contient les Metadata, ces données sont utilisées lors de l'analyse sémantique d'une requête (tables, colonnes, droits d'accès) par l'optimizer)
 - Zone mémoire pour exécuter les programmes (processus)
 - Rôles processus
 - ◆ Exécuter les actions des applications
 - ◆ Échanger les données entre les mémoires de masse et la mémoire centrale)
 - Type processus
 - ◆ Processus User (Créés une fois qu'un outil Oracle est lancé, Une session utilisateur est une connexion à la base de données par l'intermédiaire d'un processus User)
 - ◆ Processus SGBD :
 - ◇ Process server : Prennent en charge les processus user (un pour chaque processus user)
 - ◇ Process en Background
 - DBWR : Écrit dans les fichiers Database les buffers de données modifiés par la SGA
 - LGWR : Écrit le Buffer Redo Log dans le fichier Redo Log en service (il existe un seul Buffer Redo Log)
 - PMON (Process MONitor) : Assure le recouvrement des processus utilisateurs en cas de problèmes, en libérant les ressources qui étaient utilisées par ces processus (supprime le processus en cours, annule les transactions non validées, libère les verrous et les ressources utilisées dans la SGA)
 - SMON (Système monitor) : Rétablit la cohérence de la base après un incident
 - ARC (Archiver) : Recopie les fichiers Redo Log une fois qu'ils ne sont plus en service sur un support d'archivage (atteint la taille maximale)

Gestion Instance :

- Admin users
 - SYS et SYSTEM : sont créés automatiquement, possèdent le rôle DBA
- Méthode d'authentification
 - authentification OS
 - Fichier de mot de passe
- Fichier de paramètres (Fichier d'initialisation) : La qualité du fonctionnement de la BD dépend de la façon dont l'instance a été configurée, L'instance est configurée dynamiquement lors du démarrage par la lecture d'un fichier de paramètres : INIT<SID>.ORA
- État de base de données ORACLE
 - Base inexistante : La configuration de la nouvelle base est contenue dans le fichier d'initialisation, on doit démarrer l'instance
 - Base non montée : Les paramètres personnalisés sont lus et stockés en mémoire, mémoire définie dans le fichier d'initialisation est réservée, le processus background démarre
 - Base montée : Les fichiers de contrôle sont lus, État permettant d'évaluer la cohérence de la base (test si la base peut s'ouvrir sans récupération des données), sert à faire de la maintenance
 - Base ouverte : Les fichiers de données sont accessibles par les utilisateurs (fonctionnement normale d'une BD)
 - Base fermée : Utile pour réaliser des sauvegardes complètes de la base de données, et modifier les paramètres non dynamiques dans les fichiers de contrôle.

Ouverture et fermeture de BD

- 2 users qui ont droit (les privilèges : SYSDBA, SYSOPER), authentification de 2 façons :
 - Auth OS : user doit appartenir à un groupe (ex : ORA_DBA sur windows crée automatiquement), connexion avec : `CONNECT / AS SYSDBA | SYSOPER`
 - Auth fichier de mot de passe : `ORAPWD FILE=nom PASSWORD=mot_de_passe ENTRIES=valeur` (ou ENTRIES est le nombre maximum d'utilisateurs qui pourront bénéficier du privilège SYSDBA ou SYSOPER). Sous Windows, le fichier de mot de passe s'appelle `PWD<SID>.ORA`

Ouverture de la BD

STARTUP [**FORCE**][**RESTRICT**] [**PFILE**=fichier d'initialisation] [**[[OPEN] | [[RECOVER] | [[MOUNT] | [[NOMOUNT]]];**

- **FORCE** : ferme la BD avec l'option **ABORT** puis l'ouvre (à n'utiliser qu'en cas de force majeure)
- **RESTRICT** : Seuls les utilisateurs ayant le privilège **RESTRICT SESSION** peuvent accéder à la base. Option utilisée en cas d'exportation des données (éviter la modification des données)
- **PFILE** : permet de spécifier l'emplacement du fichier de paramètres à utiliser pour démarrer l'instance
- **OPEN** : permet le démarrage de l'instance et l'ouverture de la base de données (mode par défaut)
- **RECOVER** : un recouvrement de la BD sera lancé avant l'ouverture de base
- **MOUNT** : permet de démarrer l'instance et de monter la base de données. Nécessité d'utiliser la commande «**ALTER DATABASE OPEN** » pour ouvrir la base
- **NOMOUNT** : Permet le démarrage de l'instance. Pour monter et ouvrir la base, il faut utiliser les commandes : «**ALTER DATABASE MOUNT** » puis «**ALTER DATABASE OPEN** »

Fermeture de la BD

Étapes de fermetures

- 1) Les données stockées dans la SGA sont enregistrées sur le disque
- 2) Fermeture des fichiers de données
- 3) Démontage de la base en fermant les fichiers de contrôle
- 4) Arrêt de l'instance en restituant la mémoire au système d'exploitation et en terminant ses processus

Syntaxe : **SHUTDOWN** [**ABORT** | **IMMEDIATE** | **NORMAL**] [**TRANSACTIONAL**];

- **ABORT** : arrêt brutal sans enregistrement des données de la SGA (fermeture en état incohérent) => nécessite un recouvrement lors du prochain démarrage
- **IMMEDIATE** : Oracle n'accepte pas de nouvelles connexions, annule les transactions en cours des utilisateurs connectés, déconnecte ceux-ci puis ferme la base (fermeture en état cohérent)
- **NORMAL** : Oracle n'accepte pas de nouvelles connexions, par contre il attend que tous les utilisateurs se déconnectent pour fermer la base (mode par défaut)
- **TRANSACTIONAL** : Oracle n'accepte pas de nouvelles connexions, attend que tous les utilisateurs aient fini leurs transactions, puis les déconnecte ensuite ferme la base (fermeture en état cohérent)

Lister les paramètres

SHOW PARAMETERS dans SQL PLUS, ou **SELECT * FROM V\$PARAMETER**;

Type de paramètre

- Paramètres statiques : ne peuvent pas être modifiés dynamiquement : il faut modifier la valeur du paramètre dans le fichier de paramètres et redémarrer l'instance.
- Paramètres dynamiques : peuvent être modifiés alors que l'instance est en cours de fonctionnement par un ordre SQL **ALTER SESSION** (pour la session courante) ou par un ordre SQL **ALTER SYSTEM** (pour toute l'instance)

Si l'instance a démarré avec un fichier de paramètres serveur (SPFILE), l'ordre **ALTER SYSTEM** permet :

- d'enregistrer la modification dynamique d'un paramètre conjointement dans le fichier de paramètres serveur
- de modifier un paramètre dans le fichier de paramètres serveur sans modifier sa valeur dans l'instance (très utile pour les paramètres statiques)

Syntaxe : **ALTER SYSTEM SET paramètre = valeur** [...] [**SCOPE** = **MEMORY** | **SPFILE** | **BOTH**];

- MEMORY : la modification concerne uniquement l'instance en cours
- SPFILE : la modification concerne uniquement le fichier de paramètres serveur
- BOTH : la modification concerne l'instance en cours et le fichier de paramètres serveur

Exemples :

```
ALTER SYSTEM SET DB_CACHE_SIZE= 20M;
ALTER SESSION SET NLS_DATE_FORMAT = 'DD/MM/YYYY';
```

Information session

Se trouvent dans la vue dynamique **V\$SESSION**, visible avec : **SELECT sid, serial#, username, type, status FROM V\$SESSION**;

Exemples :

ALTER SYSTEM KILL SESSION '14,5' ; Où 14 est le SID et 5 est le SERIAL#

Création de la base de données

Création de variable d'environnement :

ORACLE_HOME, ORACLE_SID, ORACLE_BASE, ...

Création du fichier de paramètres texte (PFILE)

- **DB_Name** : Nom local de la base de données associé à l'instance
- **DB_Domaine** : Nom du domaine du réseau dans lequel la BD est créée => DB_Name.DB_Domaine : Nom global
- **DB_Block_Size** : Taille en octets, du bloc de données, utilisé par défaut pour l'organisation des fichiers de données et du **BUFFER CACHE** (paramètre non modifiable après la création de la BD. Il doit être entre 2ko et 32ko et doit être un multiple de la taille du bloc du SE)
- **Control_Files** : Nom ou liste des noms des fichiers de contrôle ainsi que leur localisation sur les disques du serveur. ce paramètre permet à Oracle de créer les fichiers de contrôle lors de la création de la BD.
- **Log_Buffer** : valeur en octets définissant la taille du **REDO LOG BUFFER** qui stocke en mémoire les entrées des journaux de reprise.
- **Shared_Pool_Size** : valeur en octets définissant la taille de la **Shared Pool**.
- **SGA_Max_Size** : Taille maximale de la SGA.
- **Processes** : Nombre de processus qui peuvent se connecter à l'instance au même moment (y compris les processus d'arrière-plan).

Création du fichier de paramètres serveur (SPFILE) (Qui sera utilisé durant le runtime)

Il a comme nom **spfile<SID>.ora**, le pfile avec (init<SID>.ora)

Syntaxe : **CREATE SPFILE** [= 'nom_spfile'] **FROM PFILE** [= 'nom_pfile'];

Réciproquement, un fichier de paramètres serveur peut être réexporter au format texte par l'ordre SQL **CREATE PFILE** : **CREATE PFILE** [= 'nom_pfile'] **FROM SPFILE** [= 'nom_spfile'];

Création de la base de données :

Syntaxe :

```
CREATE DATABASE [nom_base] [CONTROLFILE REUSE]
[DATAFILE spécification_fichier_data [...]] [
    UNDO TABLESPACE nom [DATAFILE spécification_fichier_data [...]]
]
[DEFAULT TEMPORARY TABLESPACE nom
    [TEMPFILE spécification_fichier_data [...]]
    [clause_gestion_extents]
]
[LOGFILE [GROUP numéro] spécification_fichier_redo [, ...]]
[ARCHIVELOG | NOARCHIVELOG]
[MAXINSTANCES nombre]
[MAXLOGFILES nombre]
[MAXLOGMEMBERS nombre]
[MAXDATAFILES nombre]
[CHARACTER SET jeu]
[NATIONAL CHARACTER SET jeu]
[SET TIME_ZONE = '+|-hh:mm'];
```

Explications :

- **Spécification_fichier_data**:: 'nom_fichier' [SIZE valeur [K|M]] [clause_auto_extend]
- **clause_auto_extend**:: **AUTOEXTEND** {**OFF** | **ON** [NEXT valeur [K|M]] [MAXSIZE UNLIMITED | valeur [K|M]]}
- **Spécification_fichier_redo**:: ('nom_fichier' [, ...]) [SIZE valeur [K|M]]
- **CONTROLFILE REUSE** :: Réutilise les fichiers de contrôles existants
- Le **fichier de données** contenant les tables du dictionnaire de données est créé et affecté par Oracle au tablespace **SYSTEM** créé automatiquement par Oracle lors de l'exécution de la commande **CREATE DATABASE** => il faut au moins créer un fichier de données (clause **DATAFILE**)
- **UNDO TABLESPACE** : permet de définir les caractéristiques du tablespace utilisé pour les segments d'annulation
- **DEFAULT TEMPORARY TABLESPACE** : permet de définir les caractéristiques du tablespace temporaire par défaut, Les journaux de reprise sont créés à partir des spécifications données après la clause **LOGFILE**. Il faut au moins créer deux journaux. Il est possible de créer des groupes de journaux de reprise qui seront multiplexés.
- **ARCHIVELOG** ou **NOARCHIVELOG** indique si la nouvelle BD fonctionne avec archivage des journaux ou non.
- **MAXINSTANCES** : nombre maximum d'instance associées à la BD.
- **MAXLOGFILES** : nombre maximum de groupes de journaux de reprise.
- **MAXLOGMEMBERS** : nombre maximum de membres par groupe de journaux de reprise.
- **MAXDATAFILES** : nombre maximum de fichiers de données pour la BD.
- **CHARACTER SET** et **NATIONAL CHARACTER SET** : permet de définir le code des caractères dans lesquels seront stockées les données.
- **SET TIME_ZONE** : permet de définir le fuseau horaire de la BD.
- la commande **CREATE DATABASE** permet aussi de créer deux utilisateurs **SYS** et **SYSTEM**. Elle exécute aussi le script **SQLBSQ** permettant de créer les tables du dictionnaire de données. À la fin, il monte la base et l'ouvre.

Exemple :

```
CREATE DATABASE biblio DATAFILE 'c:\oracle\oradata\bio\system01.bdf' SIZE 100M AUTOEXTEND ON NEXT 10M
LOGFILE
GROUP 1 ('c:\oracle\oradata\bio\redo01a.log', 'c:\oracle\oradata\bio\redo01b.log') SIZE 20M,
GROUP 2 ('c:\oracle\oradata\bio\redo02a.log', 'c:\oracle\oradata\bio\redo02b.log') SIZE 20M,
GROUP 3 ('c:\oracle\oradata\bio\redo03a.log', 'c:\oracle\oradata\bio\redo03b.log') SIZE 20M
UNDO TABLESPACE undotbs
DATAFILE 'c:\oracle\oradata\bio\undotbs01.bdf' SIZE 100M AUTOEXTEND ON NEXT 10M MAXSIZE 1024M
DEFAULT TEMPORARY TABLESPACE temp
TEMPFILE 'c:\oracle\oradata\bio\temp01.bdf' SIZE 100M AUTOEXTEND ON NEXT 10M MAXSIZE 1024M
NOARCHIVELOG
CHARACTER SET WE8ISO8859P1 --jeu de caractères principal
NATIONAL CHARACTER SET UTF8 --jeu de caractères secondaire
SET TIME_ZONE = '+00:00'
MAXINSTANCES 1
MAXLOGFILES 8
MAXLOGMEMBERS 4
MAXDATAFILES 128
```

Création des vues du dictionnaire de données

À travers le compte **SYS**, il faut exécuter le script **CATALOG.SQL** localisé dans le répertoire **ADMIN** de **ORACLE_HOME/RDBMS**

Ce script crée plusieurs types de vues en plus de celles dynamiques commençant par **V\$**. Ces vues sont préfixées en fonction de leur accès et des informations qu'elles contiennent :

- USER_** : donnent des informations sur les objets appartenant à l'utilisateur connecté
 - ALL_** : donnent des informations sur les objets auxquels a accès l'utilisateur connecté
 - DBA_** : donnent des informations sur tous les objets de la base (accessibles seulement pour les utilisateurs ayant le rôle **SELECT_CATALOG_ROLE** ou le privilège **SELECT ANY DICTIONARY**)
- Création des packages contenant des procédures et des fonctions facilitant la maintenance de la BD, il faut exécuter le script **CATPROC.SQL**

Tablespace

- Unité logique utilise pour regrouper un groupe d'objets logiques (tables, vues, séquences,)
- Chaque objet logique doit être associé à un et un seul tablespace
- Chaque tablespace est identifié par un nom et lui sont associés un ensemble de fichiers
- Facilite par exemple les opérations de sauvegarde et de restauration des données
- Un tablespace peut être supprimé même s'il contient des données
- Une BD doit avoir au moins un tablespace appelé **SYSTEM** qui contient le dictionnaire de données et un deuxième pour stocker les objets de la base

Deux états :

- **Activé (online)**
- **Désactivé (offline)**, Rq : Le tablespace **SYSTEM** ne peut jamais être désactivé

Comment décider du nombre de tablespace utilisé :

Performance : répartir les fichiers de la base de données sur différents disques a une effet sur la performance

Organisation :

- Faciliter les opérations de maintenance
- Gérer des données utilisées pour des applications différentes dans une seule base de données
- Spécialiser des tablespaces pour des tâches techniques interne à Oracle

Syntaxe :

```
CREATE TABLESPACE nom
DATAFILE spécification_fichier_data [...],
[MINIMUM EXTENT valeur [K|M]]
[ONLINE|OFFLINE] [PERMANENT|TEMPORARY]
[DEFAULT clause_stockage]
[BLOCKSIZE valeur [k]];
```

Clause_stockage ::

```
STORAGE (
[INITIAL valeur [K|M]]
[NEXT valeur [K|M]]
[MINEXTENTS valeur]
[MAXEXTENTS (valeur|UNLIMITED)]
[PCTINCREASE valeur]
)
```

- MINIMUM EXTENT** : indique que les extensions allouées pour les segments de ce tablespace seront au moins de la valeur de cette option ou de son multiple.
- ONLINE** : les utilisateurs ont accès aux objets du tablespace.
- OFFLINE** : les utilisateurs n'ont pas accès aux objets du tablespace.
- PERMANENT** : Ce tablespace peut contenir des objets permanents et des segments temporaires (valeur par défaut).
- TEMPORARY** : Ce tablespace ne peut contenir que des segments temporaires.
- INITIAL** : Taille de la première extension lors de la création du segment dans le tablespace (par défaut = 5 block de données).
- NEXT** : Taille de la deuxième extension, lorsque la première est utilisée (par défaut = 5 blocks de données).
- PCTINCREASE** : Pourcentage d'augmentation de la taille de l'extension à partir de la troisième (par défaut = 50).
- MINEXTENTS** : Nombre d'extensions affectées au segment lors de sa création (par défaut = 1).
- MAXEXTENTS** : Nombre maximum d'extensions qui peuvent être créées pour le segment (valeur par défaut = 121).

Remarque : Les options de stockage par défaut s'appliquent à tous les nouveaux objets créés dans le tablespace dont l'option de stockage n'a pas été renseignée.

Exemple :

```
CREATE TABLESPACE app_data
DATAFILE ('c:\oracle\oradata\BIBLIO\data01.dbf' SIZE 20M, 'd:\oracle\oradata\BIBLIO\data02.dbf' SIZE 10M
AUTOEXTEND ON NEXT 5M MAXSIZE 50M MINIMUM EXTENT 150K
ONLINE;
```

Création d'un tablespace temporaire

Exemple:

```
CREATE TEMPORARY TABLESPACE app_temp
TEMPFILE 'C:\oracle\oradata\BIBLIO\temp01.dbf' SIZE 20M
AUTOEXTEND ON NEXT 5M MAXSIZE 50M;
```

Modification d'un tablespace

Remarque : Seuls les nouveaux objets pour lesquels aucun type de stockage n'a été défini seront concernés par ce nouveau stockage par défaut.

Exemple :

```
ALTER TABLESPACE app_data
ADD DATAFILE 'f:\oracle\oradata\BIBLIO\data03.DBF' SIZE 40M AUTOEXTEND ON NEXT 5M MAXSIZE 80M;
ALTER DATABASE
DATAFILE 'C:\oracle\oradata\BIBLIO\data01.DBF' RESIZE 30M;
ALTER TABLESPACE app_data
ONLINE;
ALTER TABLESPACE app_data
OFFLINE;
ALTER TABLESPACE nom
READ ONLY;
ALTER TABLESPACE app_data
READ WRITE;
ALTER TABLESPACE app_data DEFAULT STORAGE (INITIAL 256k NEXT 256k PCTINCREASE 25);
DROP TABLESPACE app_data INCLUDING CONTENTS AND DATAFILES CASCADE CONSTRAINTS;
```

Déplacer un fichier de données :

```
CONNECT / AS SYSDBA
SHUTDOWN IMMEDIATE
HOST move D:\oracle\oradata\BIBLIO\data01.dbf d:\oracle\oradata\BIBLIO\data01.dbf
STARTUP MOUNT
ALTER DATABASE RENAME FILE 'C:\oracle\oradata\BIBLIO\data01.dbf' TO 'D:\oracle\oradata\BIBLIO\data01.dbf';
ALTER DATABASE OPEN;
```

Création de tables :

Création :

```
CREATE TABLE employees(empno NUMBER(4), last_name VARCHAR2(30) deptno NUMBER(2))
PCTFREE 20 PCTUSED 50
STORAGE (INITIAL 200K NEXT 200K PCTINCREASE 0 MAXEXTENTS 50)
TABLESPACE data01
AS SELECT * FROM scott.employees;
```

Remarque :

PCTFREE : Free space in a data block, By default, its value is 10. The statement "PCTFREE is set as 10" means that 10% space would be kept free in all data blocks in the table for its future growth

Suppression :

```
DROP TABLE scott.departments CASCADE CONSTRAINTS;
```

Allocation manuelle d'extent :

```
ALTER TABLE scott.employees ALLOCATE EXTENT (SIZE 500K DATAFILE '/DISK3/DATA01.DBF')
```

Remarque :

The analyze command can validate that tables and indexes are in sync with each other.

```
ANALYZE TABLE scott.employees VALIDATE STRUCTURE;
```

Classification des index

Logique

Colonne unique ou concaténée
Unique ou non unique

Physique

Partitionné ou non partitionné
Organisé en B-tree ou bitmap

Création d'index organisée en BTREE

```
CREATE INDEX scott.emp_lastname_idx
ON scott.employees(last_name)
PCTFREE 30 STORAGE (INITIAL 200K NEXT 200K PCTINCREASE 0 MAXEXTENTS 50)
TABLESPACE indx01;
```

Création d'index à clés inverse

```
CREATE UNIQUE INDEX scott.ord_ord_no_idx
ON scott.ord(ord_no) REVERSE
PCTFREE 30 STORAGE (INITIAL 200K NEXT 200K PCTINCREASE 0 MAXEXTENTS 50)
TABLESPACE indx01;
```

Création d'index bitmap

```
CREATE BITMAP INDEX scott.ord_region_id_idx
ON scott.ord(region_id) PCTFREE 30 STORAGE (INITIAL 200K NEXT 200K PCTINCREASE 0 MAXEXTENTS 50)
TABLESPACE indx01;
```

Allocation ou Désallocation d'un Espace d'Index

```
ALTER INDEX scott.ord_region_id_idx ALLOCATE EXTENT (SIZE 200K DATAFILE '/DISK6/indx01.dbf');
ALTER INDEX scott.ord_ord_no_idx DEALLOCATE UNUSED;
```

Reconstitution d'Index (Déplacement vers un nouveau tablespace)

```
ALTER INDEX scott.ord_region_id_idx REBUILD TABLESPACE indx02;
```

Vérification de la Validité de l'Index

```
ANALYZE INDEX scott.ord_region_id_idx VALIDATE STRUCTURE;
```

Suppression d'Index

```
DROP INDEX scott.dept_dname_idx;
```

Définition des contraintes lors de la création des tables

By default constraints are created as NON DEFERRABLE but this can be overridden using the DEFERRABLE keyword, A deferred constraint is only checked at the point the transaction is committed.

```
CREATE TABLE emp (empno NUMBER(4) NOT NULL, last_name VARCHAR2(30) NOT NULL, deptno NUMBER(2) NOT NULL, PRIMARY KEY (empno, last_name));
```

```
CREATE TABLE scott.employees(empno NUMBER(4))
CONSTRAINT emp_pk PRIMARY KEY
DEFERRABLE USING INDEX STORAGE(INITIAL 100K NEXT 100K)
TABLESPACE indx01, last_name VARCHAR2(30)
CONSTRAINT emp_in_nn NOT NULL, deptno NUMBER(2)) TABLESPACE data01;
```

```
ALTER TABLE scott.departments DISABLE CONSTRAINT dept_pk CASCADE;
```

ENABLE VALIDATE is the same as **ENABLE**. The constraint is checked and is guaranteed to hold for all rows.

ENABLE NOVALIDATE means the constraint is checked for new or modified rows, but existing data may violate the constraint.

DISABLE NOVALIDATE is the same as **DISABLE**. The constraint is not checked so data may violate the constraint.

DISABLE VALIDATE means the constraint is not checked but disallows any modification of the constrained columns.

Supprimez une table et toute clé étrangère la désignant à l'aide de cette commande :

```
DROP TABLE departments CASCADE CONSTRAINTS;
```

Triggers

```
ALTER TRIGGER scott.emp_conv_in DISABLE;
ALTER TABLE scott.employees ENABLE ALL TRIGGERS;
ALTER TABLE scott.employees DROP CONSTRAINT emp_in_uk;
DROP TRIGGER scott.audit_dept;
```

Gestion des utilisateurs

Liste de Contrôle pour la Création d'Utilisateurs :

- o Choisissez un nom d'utilisateur et un mécanisme d'authentification
- o Identifiez les tablespaces dans lesquels l'utilisateur doit stocker des objets
- o Décidez des quotas pour chaque tablespace
- o Affectez un tablespace par défaut et un tablespace temporaire
- o Créez un utilisateur
- o Accordez des privilèges et des rôles à l'utilisateur

Définition du mot de passe initial :

```
CREATE USER Mohamed
IDENTIFIED BY ensias1 DEFAULT TABLESPACE data01
TEMPORARY TABLESPACE temp
QUOTA 15M ON data01
PASSWORD EXPIRE;
```

```
ALTER USER Mohamed IDENTIFIED BY ensias2 PASSWORD EXPIRE;
ALTER USER Mohamed QUOTA 30M ON data01;
DROP USER Mohammed;
```

Remarques :

- Utilisez la clause **CASCADE** si le schéma contient des objets :
DROP USER Mohamed CASCADE;
- Utilisez le mot-clé **EXPIRE** pour obliger les utilisateurs à réinitialiser leur mot de passe
- utilisez **QUOTA UNLIMITED** avec précautions

Gestion des profils

Définition

Sont des ensembles nommés de limites de ressources et de mots de passe

Sont affectés aux utilisateurs par la commande **CREATE/ALTER USER**

Peuvent être activés ou désactivés

Peuvent limiter les ressources du système au niveau de la session ou de l'appel

Création d'un Profil : Limite de Ressources

```
CREATE PROFILE MonProfil LIMIT
SESSIONS_PER_USER 2
CPU_PER_SESSION 10000
IDLE_TIME 60
CONNECT_TIME 480;
```

Affectation de Profils à un Utilisateur

```
CREATE USER user3 IDENTIFIED BY user3
DEFAULT TABLESPACE data01
TEMPORARY TABLESPACE temp
QUOTA UNLIMITED ON data01
PROFILE MonProfil;
```

```
ALTER USER scott PROFILE MonProfil;
```

Activation des Limites de Ressources

```
ALTER SYSTEM SET RESOURCE_LIMIT=TRUE;
```

Modification d'un Profil

```
ALTER PROFILE default LIMIT
SESSIONS_PER_USER 5
CPU_PER_CALL 3600
IDLE_TIME 30;
```

```
DROP PROFILE MonProfil
```

```
DROP PROFILE MonProfil CASCADE;
```

Gestion des Privilèges

types de privilèges :

- **SYSTEME** : permet aux utilisateurs d'effectuer des opérations particulières dans la base de données
- **OBJET** : permet aux utilisateurs d'accéder à un objet particulier et de le manipuler

Privilèges système :

- Il existe environ 80 privilèges système
- Le mot-clé **ANY** dans les privilèges signifie que les utilisateurs possèdent ce privilège dans tous les schémas
- La commande **GRANT** ajoute un privilège à un utilisateur ou un groupe d'utilisateurs
- La commande **REVOKE** supprime les privilèges

Attribution de Privilèges Système

```
GRANT CREATE SESSION, CREATE TABLE TO user1;
GRANT CREATE SESSION TO scott WITH ADMIN OPTION;
```

Révocation des Privilèges Système

```
REVOKE CREATE TABLE FROM user1;
REVOKE CREATE SESSION FROM scott;
```

Attribution de Privilèges Objet

```
GRANT EXECUTE ON dbms_pipe TO public;
GRANT UPDATE(ename,sal) ON emp TO user1 WITH GRANT OPTION;
```

Révocation des Privilèges Objet

```
REVOKE execute ON dbms_pipe FROM scott;
WITH GRANT OPTION;
```

Si on fait **WITH GRANT option**, le user ayant "**GRANTED**" peut attribuer un privilège à d'autres **USER**

Gestion des Rôles

Avantages des Rôles :

- o Gestion simplifiée des privilèges
- o Gestion dynamique des privilèges
- o Disponibilité sélective des privilèges
- o Privilèges accordés au moyen de l'OS
- o Pas de suppression de privilèges en cascade
- o Performances améliorées

Création des rôle

```
CREATE ROLE sales_clerk;
CREATE ROLE hr_clerk IDENTIFIED BY bonus;
This example creates the role called hr_clerk, but now it is password protected with the password of bonus
CREATE ROLE hr_manager IDENTIFIED EXTERNALLY;
You create database accounts that must be authenticated by the operating system or network service
```

```
ALTER ROLE sales_clerk IDENTIFIED BY commission;
ALTER ROLE hr_clerk IDENTIFIED EXTERNALLY;
ALTER ROLE hr_manager NOT IDENTIFIED;
GRANT sales_clerk TO scott;
GRANT hr_manager TO scott WITH ADMIN OPTION;
ALTER USER scott DEFAULT ROLE hr_clerk, sales_clerk;
ALTER USER scott DEFAULT ROLE ALL;
ALTER USER scott DEFAULT ROLE ALL EXCEPT hr_clerk;
ALTER USER scott DEFAULT ROLE NONE;
```

La commande **SET ROLE** active et désactive les rôles

Les rôles par défaut d'un utilisateur sont activés à la connexion

```
..
```

Un mot de passe est parfois exigé pour activer un rôle

```
SET ROLE sales_clerk IDENTIFIED BY commission;  
SET ROLE ALL EXCEPT sales_clerk;  
SET ROLE hr_clerk;  
SET ROLE NONE;
```

```
REVOKE sales_clerk FROM scott;  
REVOKE hr_manager FROM PUBLIC;  
DROP ROLE hr_manager;
```

Affectation de privilège aux rôles

Syntaxe :

```
GRANT privileges ON object TO role_name
```

Exemples :

```
GRANT select, insert, update, delete ON suppliers TO test_role;  
GRANT all ON suppliers TO test_role;  
REVOKE delete ON suppliers FROM test_role;  
REVOKE all ON suppliers FROM test_role;  
GRANT execute ON Find_Value TO test_role;  
GRANT EXECUTE privileges on a function or procedure to a role in Oracle.
```