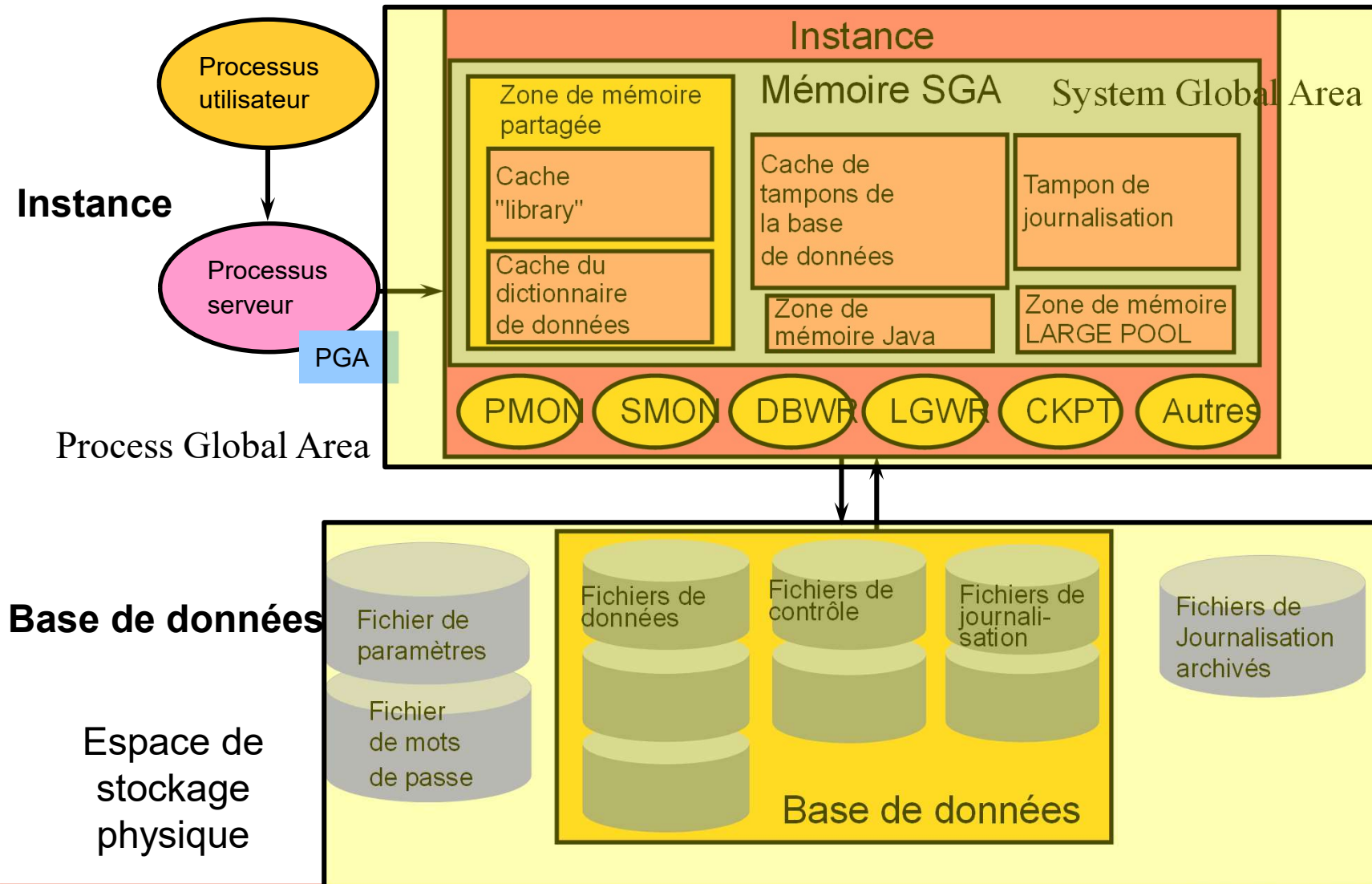


# Administration d'une base de données Oracle

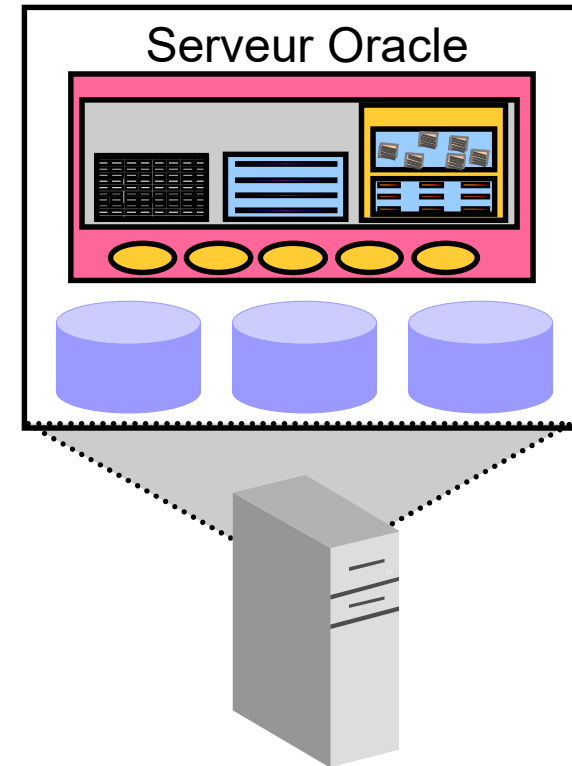
# Architecture du SGBD Oracle



# Serveur Oracle

## Un serveur Oracle :

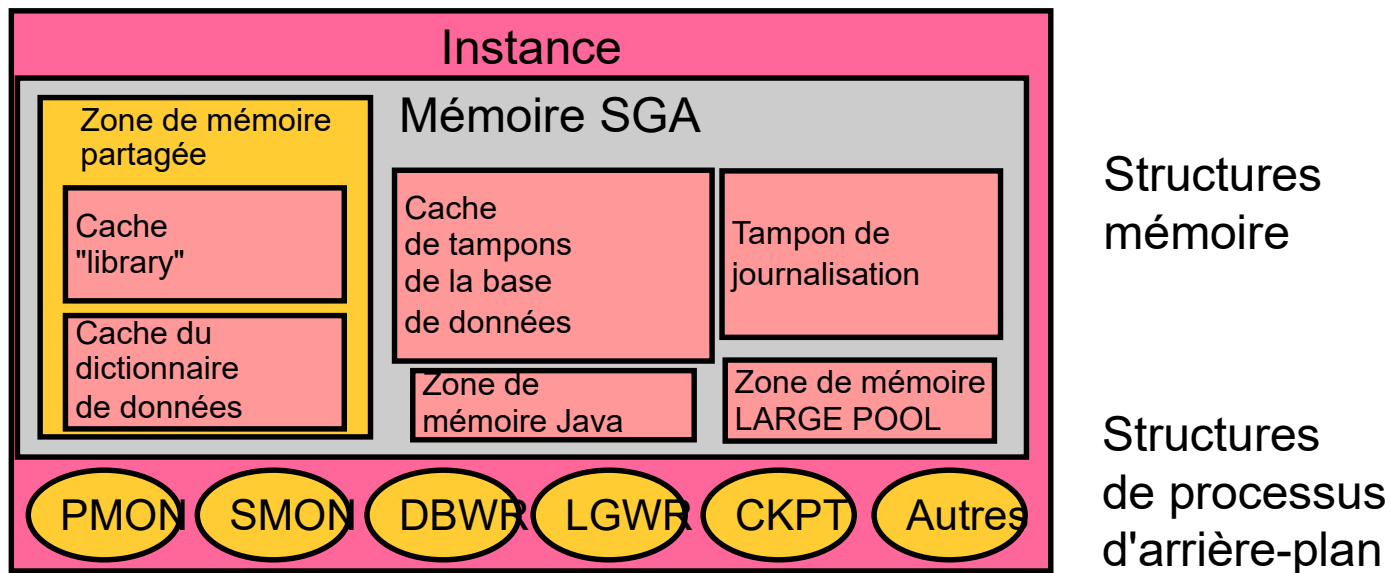
- Est un système de gestion de base de données qui offre une méthode de gestion des informations ouverte, complète et intégrée,
- Gère de façon fiable dans un environnement multiutilisateur une quantité importante de données.
- Sécurisé et propose des solutions efficaces de récupération des données après incident
- Est constitué d'une **instance** et d'une **base de données** Oracle.



# Instance Oracle

## Une instance Oracle :

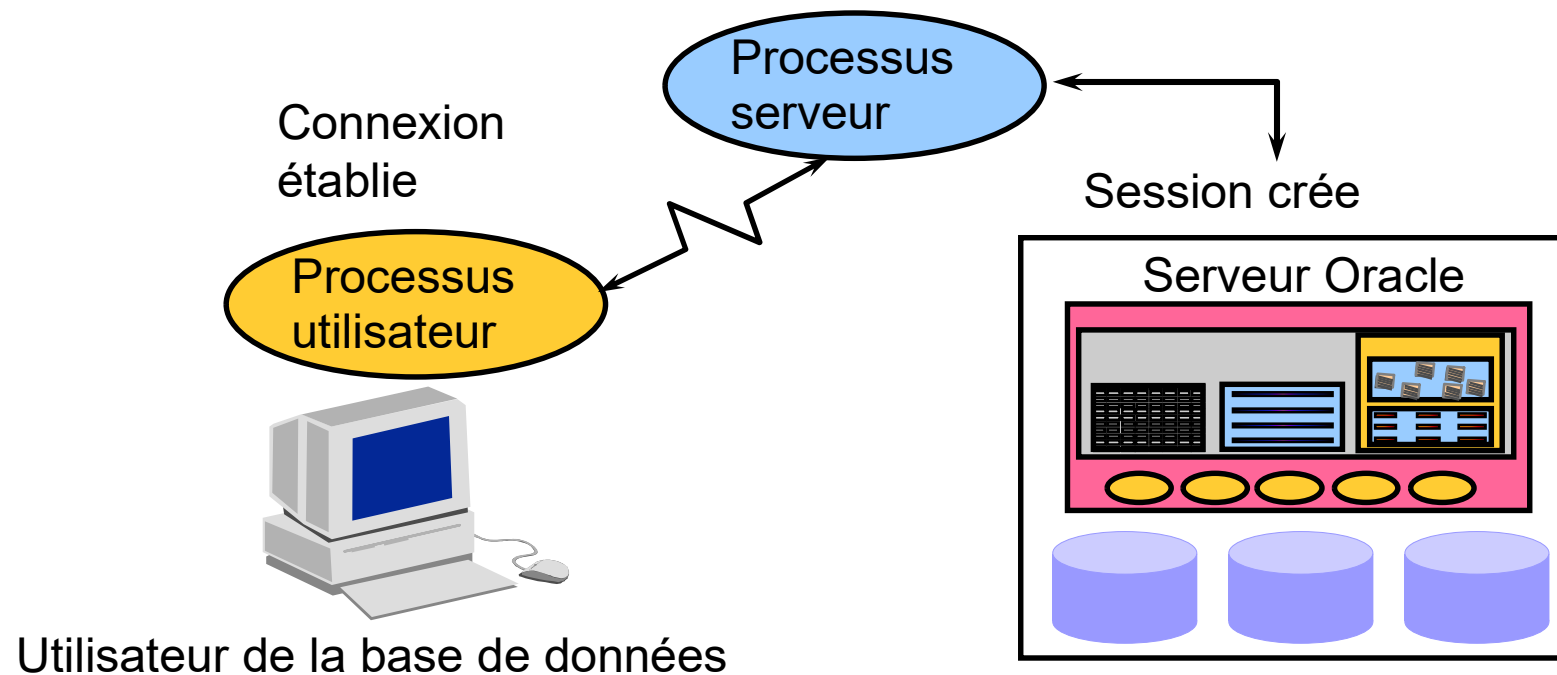
- Permet d'accéder à une base de données Oracle,
- N'ouvre qu'une seule base de données,
- Est constituée de structures de processus d'arrière-plan et de structures mémoire.



# Etablir une connexion et créer une session

## Se connecter à une instance Oracle :

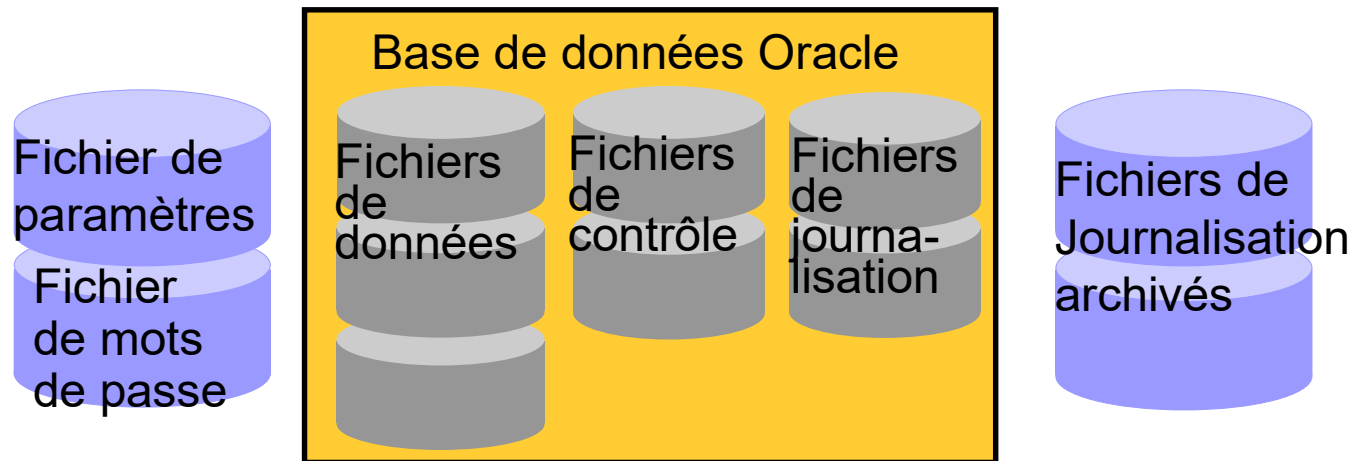
- Etablir une connexion utilisateur
- Créer une session



# Base de données Oracle

## Une base de données Oracle :

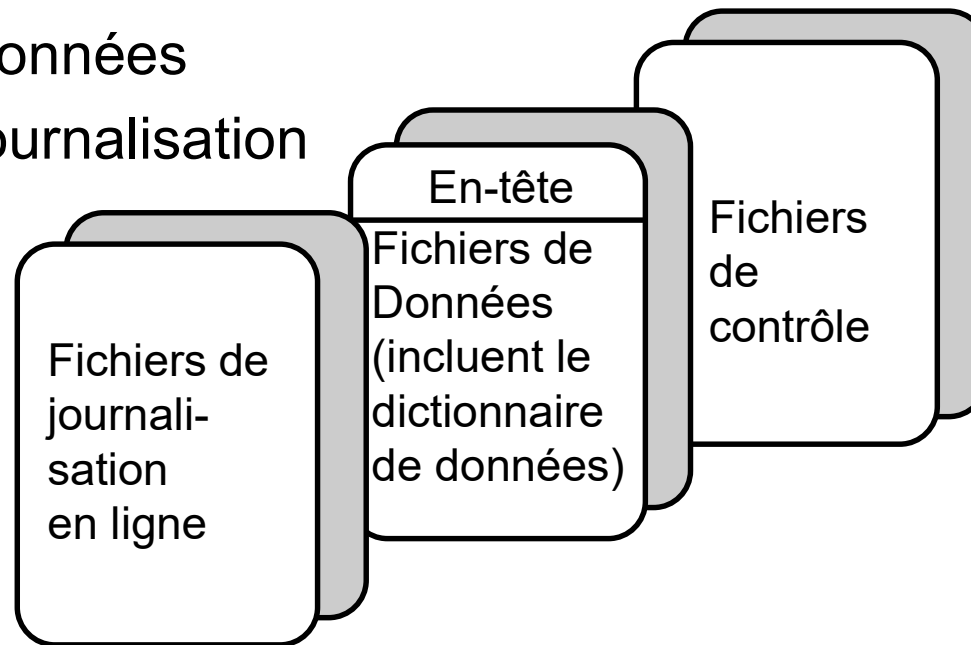
- Est un ensemble de données traitées comme une seule et même entité,
- Est constituée de trois types de fichier.



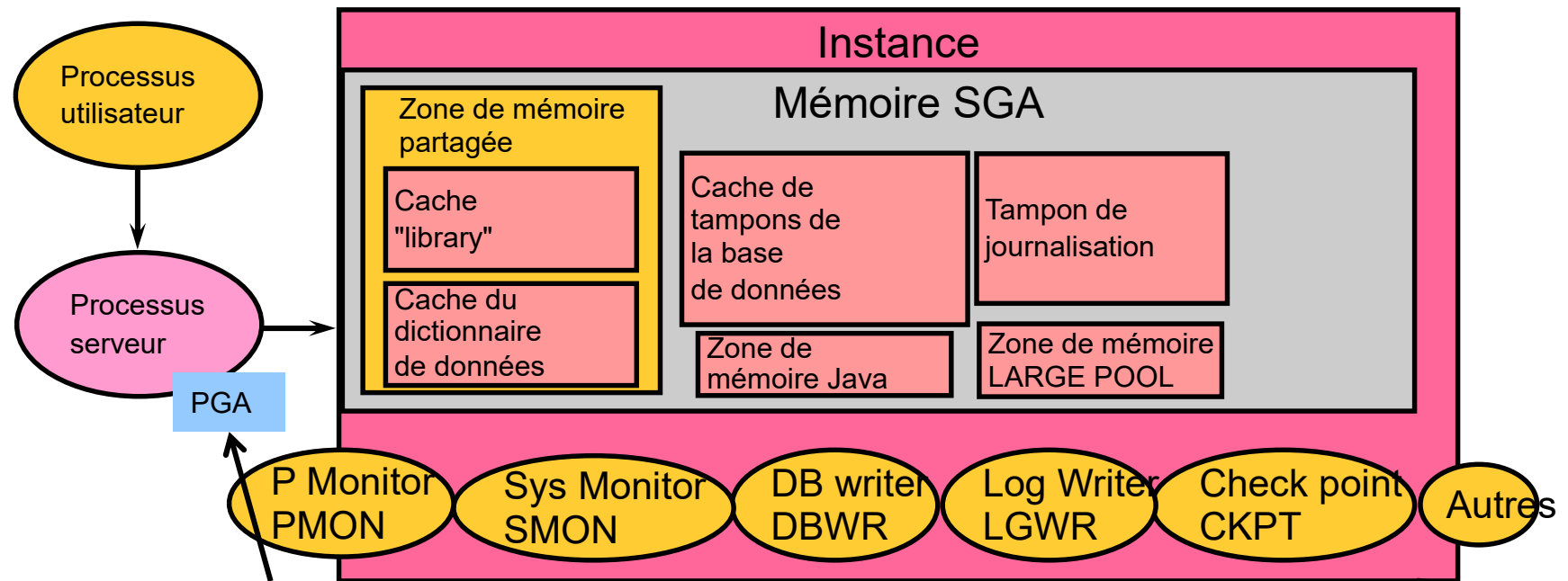
# Structure physique

**La structure physique comprend trois types de fichier :**

- Fichiers de contrôle
- Fichiers de données
- Fichiers de journalisation



# Présentation des principaux composants



**Mémoire PGA:** réservée à chaque processus utilisateur qui se connecte à une base de données Oracle

**Mémoire SGA:** Allouée au démarrage de l'instance et qui est une composante fondamentale d'une instance Oracle (mémoire partagée)

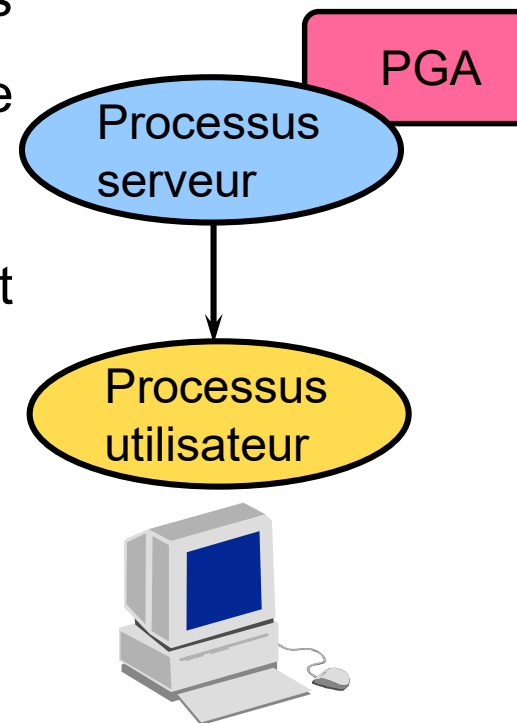


# Mémoire SGA

- **La mémoire SGA est constituée de plusieurs structures mémoire :**
  - **La zone de mémoire partagée (Shared pool):** Contient les requêtes SQL les plus récemment exécutés et les informations du dictionnaire de données les plus récemment utilisées.
  - **Le cache de tampons de la base de données (Database Buffer Cache) :** Cette structure stocke les données utilisatrices les plus récentes tirées des tables.
  - **Le tampon de journalisation (Redo Log Buffer) :** Ce buffer est utilisé pour enregistrer les modifications effectuées sur la base de données.
  - **D'autres structures :** gestion des verrous, données statistiques, ...

# Mémoire PGA

- Mémoire réservée à chaque processus utilisateur qui se connecte à une base de données Oracle.
- Elle est allouée lorsqu'un processus est créé.
- Elle est libérée à la fin du processus.
- Elle n'est utilisée que par un processus.



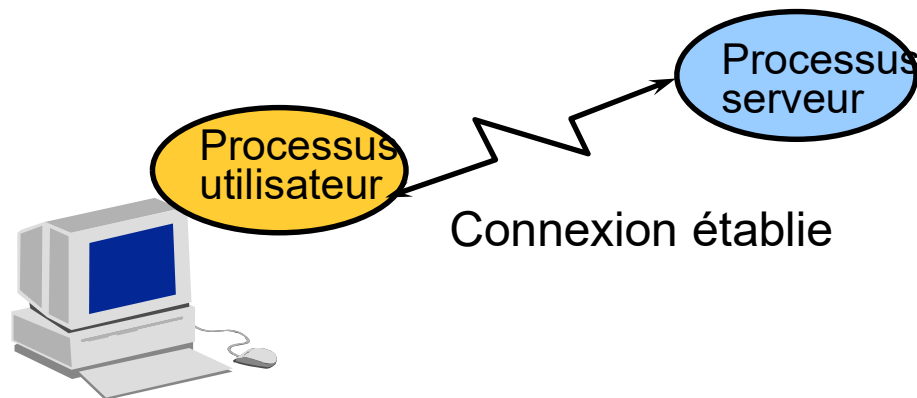
## Structure de processus

Oracle utilise différents types de processus :

- **Le processus utilisateur**, est démarré au moment où un utilisateur de la base de données tente de se connecter au serveur Oracle,
- **Le processus serveur**, établit la connexion à l'instance Oracle et se démarre lorsqu'un utilisateur ouvre une session,
- **Les processus d'arrière-plan**, lancés au démarrage d'une instance Oracle.

# Processus utilisateur

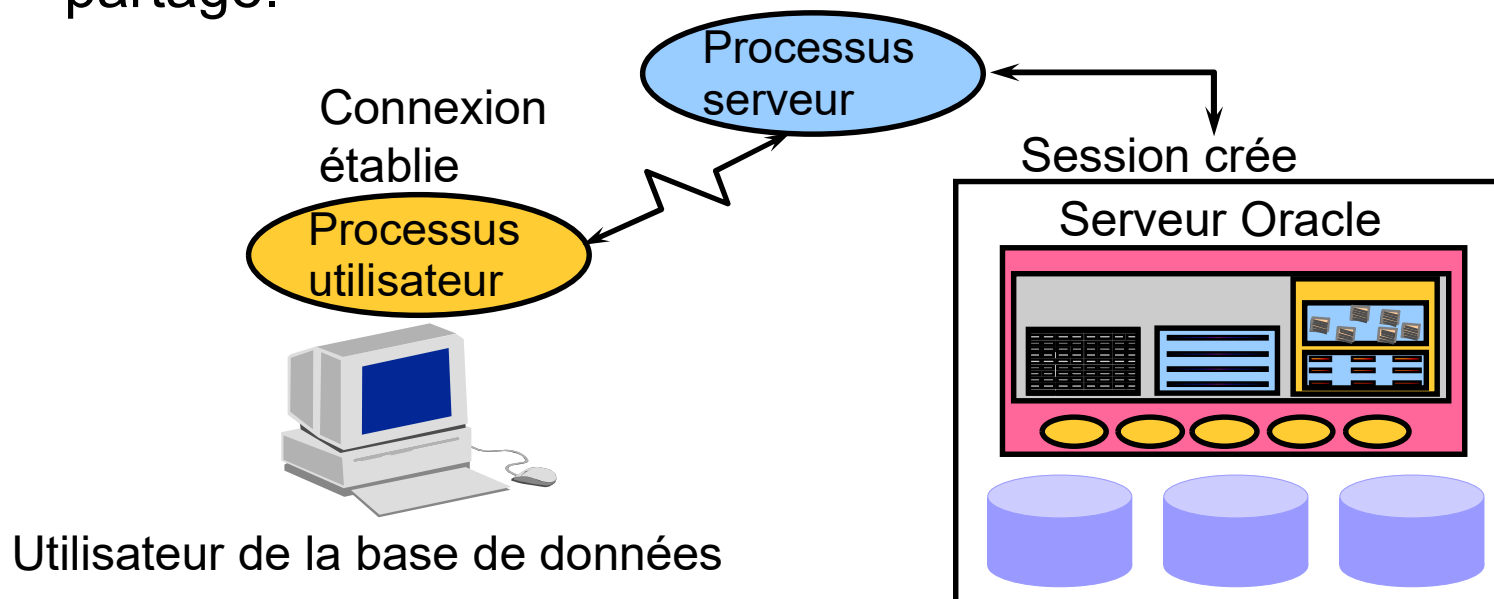
- Programme qui demande une interaction avec le serveur Oracle.
- Ce processus doit d'abord établir une connexion.
- Il n'entre pas directement en interaction avec le serveur Oracle.



Utilisateur de la base de données

# Processus serveur

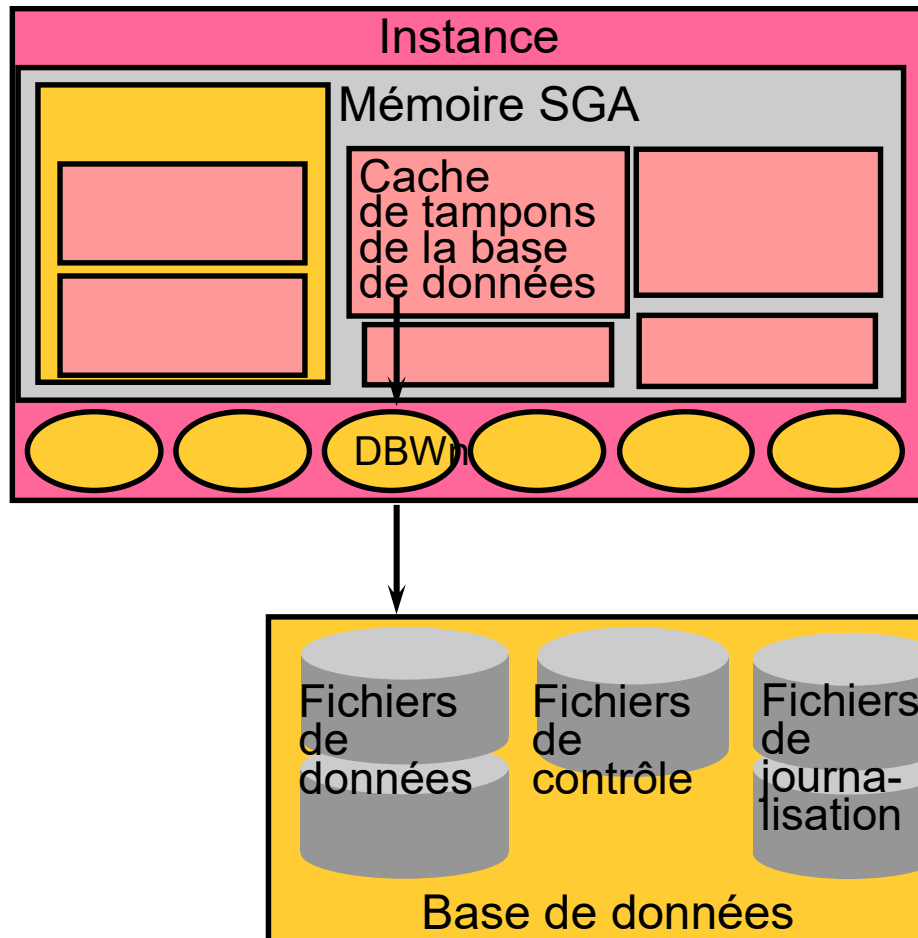
- Programme qui entre directement en interaction avec le serveur Oracle.
- Il répond aux appels générés et renvoie les résultats.
- Il peut s'agir d'un serveur dédié ou d'un serveur partagé.



## Processus d'arrière plan

- Gèrent et appliquent les relations entre les structures physiques et les structures mémoires.
- Processus d'arrière-plan obligatoires
  - DBWR PMON CKPT
  - LGWR SMON
- Processus d'arrière-plan facultatifs
  - ARCn LMDn RECO
  - CJQ0 LMON Snnn
  - Dnnn Pnnn
  - LCKn QMNN

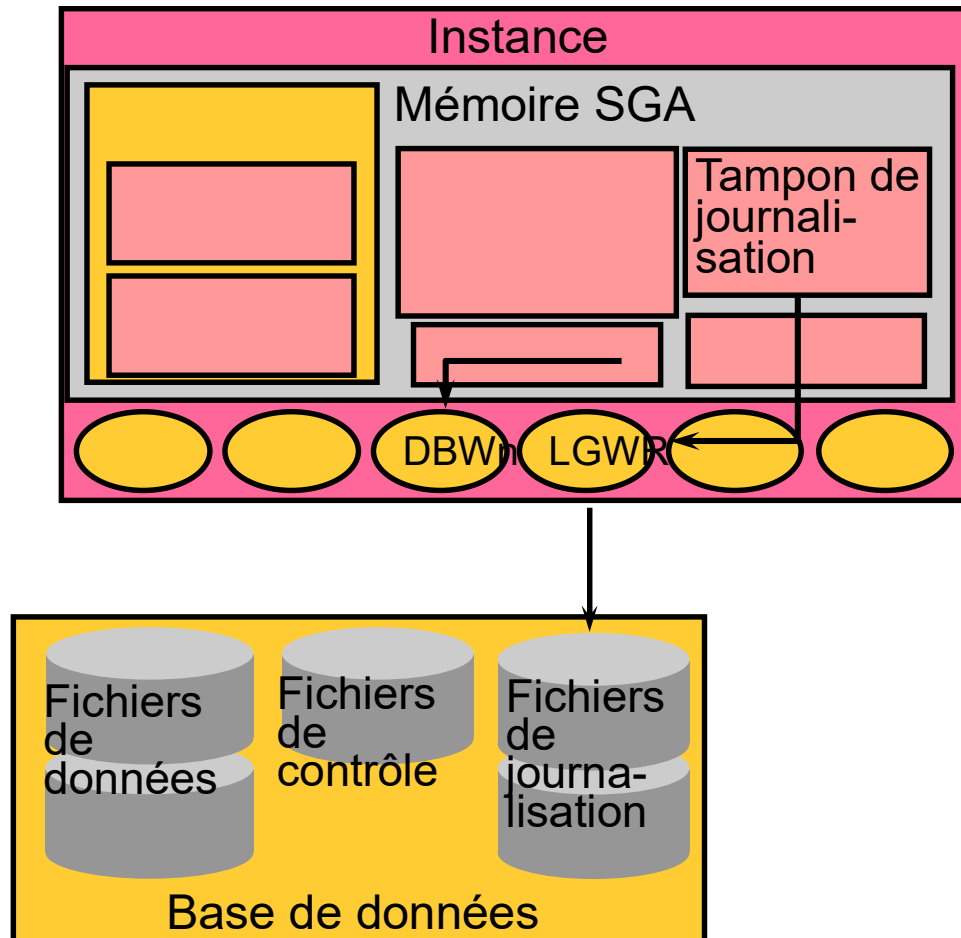
# Processus database writer (DBWR)



DBWR est chargé d'écrire le contenu des buffers dans les fichiers de données. Il écrit dans les cas suivants :

- Aucune mémoire tampon disponible
- Temps imparti dépassé
- Tablespace hors ligne
- Point de reprise
- Tablespace en lecture seule
- DROP ou TRUNCATE sur une table
- ...

# Processus LGWR (Log Writer)

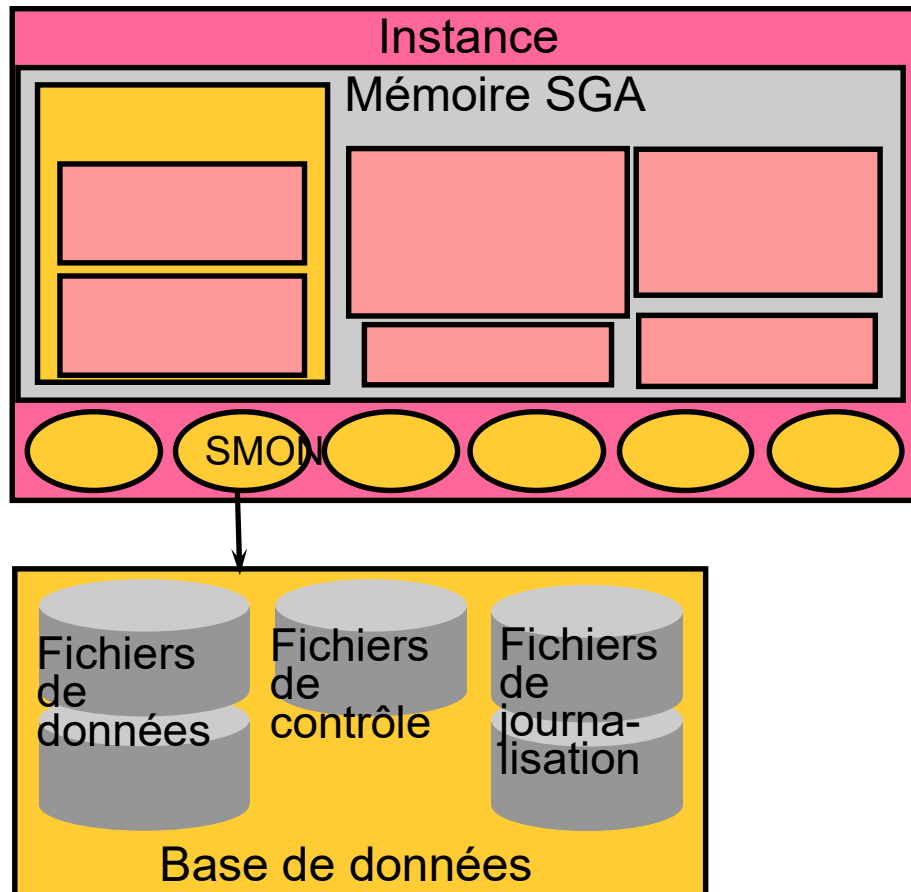


LGWR écrit dans les cas suivants :

- Validation
- 1/3 du cache est occupé
- La journalisation atteint 1 Mo
- Toutes les trois secondes
- Avant que le processus DBWR ne procède à une opération d'écriture



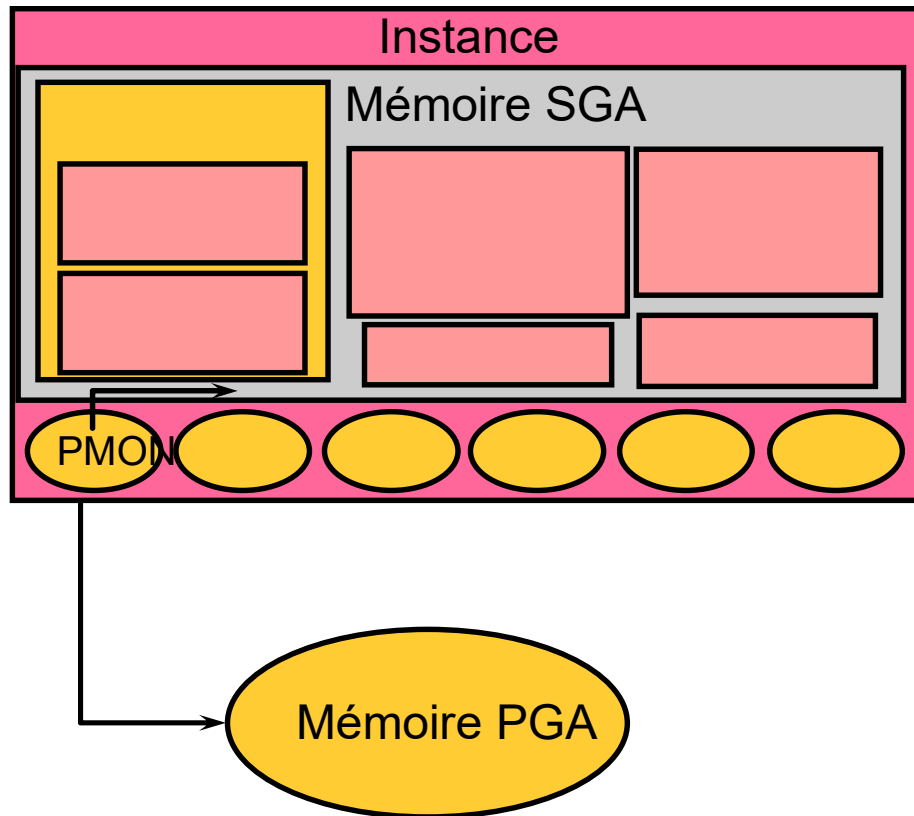
# Processus SMON (System Monitor)



## Responsabilités :

- Récupération de l'instance :
  - Réimplémente des modifications dans les fichiers de journalisation,
  - Ouvre la base de données pour permettre l'accès aux utilisateurs,
  - Annule les transactions non validées.
- Fusion de l'espace libre
- Libération des segments temporaires.

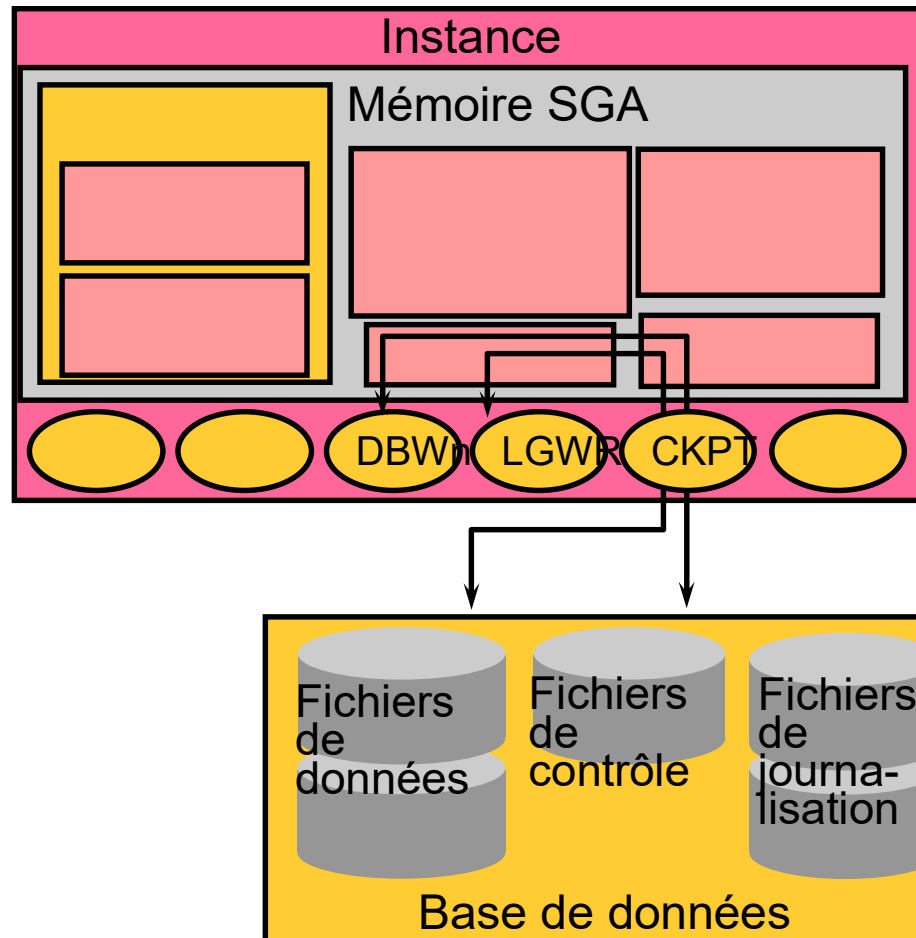
# Processus PMON (Process Monitor)



Suite à l'échec de processus, PMON exécute des opérations de nettoyage :

- Annule la transaction
- Libère des verrous
- Libère d'autres ressources
- Redémarre les répartiteurs interrompus.

# Processus CKPT (Checkpoint)

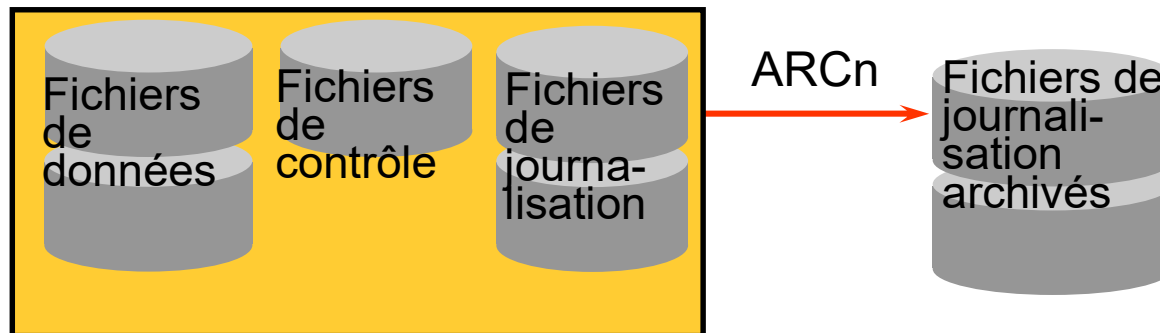


Ce processus est chargé :

- De signaler DBWR aux points de reprise,
- De mettre à jour les en-têtes de fichiers de données avec les informations sur le point de reprise,
- De mettre à jour les fichiers de contrôle avec les informations sur le point de reprise.

## Processus ARCn (processus d'archivage)

- Processus d'arrière-plan facultatif
- En mode `ARCHIVELOG`, il archive automatiquement les fichiers de journalisation en ligne .
- Il enregistre toutes les modifications apportées à la base de données.



## Traiter les instructions SQL

- Connexion à une instance via :
  - Le processus utilisateur,
  - Le processus serveur.
- Les composants du serveur Oracle utilisés dépendent du type d'instruction SQL :
  - Les interrogations renvoient des lignes.
  - Les instructions LMD consignent les modifications.
  - La validation garantit la récupération de la transaction.
- Certains composants du serveur Oracle n'interviennent pas dans le traitement des instructions SQL.

# Contrôle d'accès des utilisateurs

# Privilège système

- ☐ Droits pour exécuter des requêtes.
- ☐ DBA est le plus haut niveau de privilèges.
  - ✓ Donner aux utilisateurs l'accès à la base de données.
- ☐ Les utilisateurs doivent posséder:
  - ✓ Des **privilèges système** pour se connecter à la base de données,
  - ✓ Des **privilèges objets** pour manipuler des données.

# Privilèges DBA

- ❑ Il existe plus de 100 privilèges pour les utilisateurs et les rôles.

Privilège système	Opération autorisée
CREATE USER	Autorise de créer des utilisateurs
DROP USER	Autorise de supprimer des utilisateurs
DROP ANY TABLE	Autorise de supprimer toutes les tables dans tous les schémas
BACKUP ANY TABLE	Autorise de sauvegarder toutes les tables dans tous les schémas.
SELECT ANY TABLE	Autorise d'effectuer les requêtes SELECT dans tous les schémas.
CREATE ANY TABLE	Autorise de créer des tables dans tous les schémas.



# Créer un utilisateur

- ❑ Un DBA peut créer des utilisateurs avec la commande

**CREATE USER:**

**Syntaxe:**

**CREATE USER** user1 **IDENTIFIED BY** mot\_de\_passe

# Privilèges utilisateurs

- ❑ Après la création d'un compte utilisateur, le DBA peut lui accorder des privilèges.

- **Syntaxe:**

```
GRANT privilege1 ,privilege2, ...  
TO user1, user2, role1, PUBLIC ...;
```

- **Exemple:**

```
GRANT CREATE SESSION TO user1;
```

# Privilèges utilisateurs

Privilège	Opération autorisée
<b>CREATE SESSION</b>	Autorise la connexion à la base de données
<b>CREATE TABLE</b>	Autorise à créer des tables
<b>CREATE SEQUENCE</b>	Autorise à créer des séquences
<b>CREATE VIEW</b>	Autorise à créer des vues
<b>CREATE PROCEDURE</b>	Autorise à créer des procédures, des fonctions ou des packages

# Présentation des rôles

- **Définition**

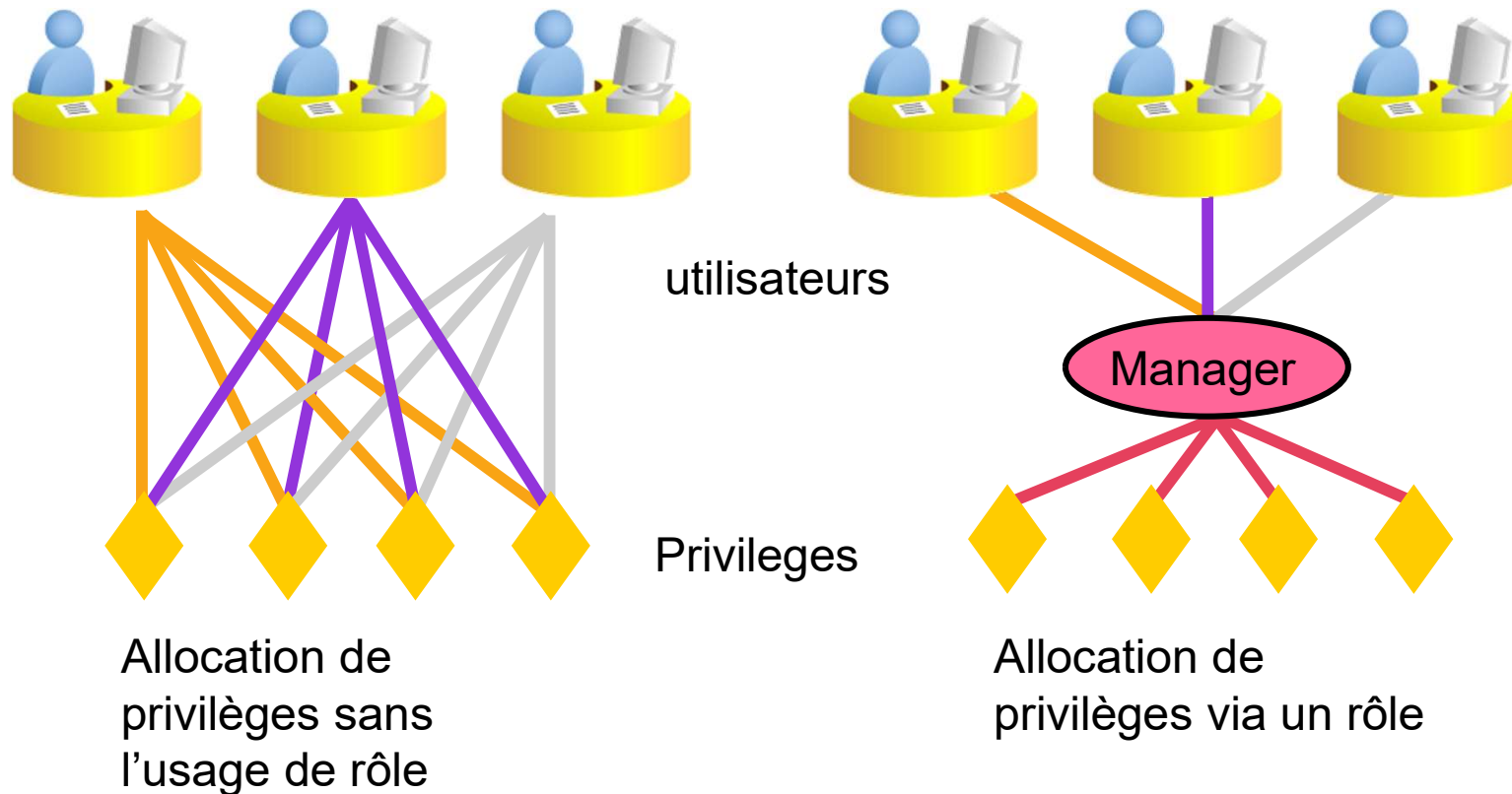
- Un rôle est un concept Oracle qui permet de regrouper plusieurs privilèges et / ou rôles afin de les affecter ou retirer en bloc à un utilisateur et / ou un rôle.

- Un rôle facilite la gestion des privilèges.
- Pour des raisons de sécurité, un mot de passe peut être assigné à un rôle.
- Oracle fournit un certain nombre de rôles par défaut (connect, resource, dba, exp\_full\_database, imp\_full\_data\_base, select\_catalog\_role, delete\_catalog\_role / execute\_catalog\_role, ...)
- Pour créer un rôle, il faut avoir le privilège **"CREATE ROLE"**

# Créer et accorder un rôle

## Exemple:

un chef de département possède tous les privilèges alors qu'une secrétaire ne peut exécuter qu'une requête **SELECT**.



# Créer et accorder un rôle

- ❑ Ensemble de privilèges.

- Syntaxe:

```
CREATE ROLE nom_role;
```

- ❑ Assignment de privilèges à un rôle.

```
GRANT privilege1, privilege2, ...  
TO nom_rôle;
```

- ❑ Assignment d'un rôle à des utilisateurs.

```
GRANT nom_rôle  
TO user1,user2,...;
```

# Liste des rôles

- Les rôles ( obtenir des informations)
  - Interrogez les vues suivantes du dictionnaire de données :
    - **DBA\_ROLES** : Tous les rôles qui existent dans la base de données
    - **DBA\_ROLE\_PRIVS** : Rôles accordés à des utilisateurs et à des rôles
    - **ROLE\_ROLE\_PRIVS** : Rôles accordés à des rôles
    - **DBA\_SYS\_PRIVS** : Privilèges système accordés à des utilisateurs et à des rôles
    - **ROLE\_SYS\_PRIVS** : Privilèges système accordés à des rôles
    - **ROLE\_TAB\_PRIVS** : Privilèges objet accordés à des rôles
    - **SESSION\_ROLES** : Rôles activés par l'utilisateur

**Exemple 1** : liste de tous les rôles de la base

```
sql> SELECT * FROM dba_roles ;
```

<u>ROLE</u>	<u>PASSWORD</u>
CONNECT	NO
RESOURCE	NO
DBA	NO
EXP_FULL_DATABASE	NO
IMP_FULL_DATABASE	NO
MONITORER	NO
RL_ADMIN_SECU	NO
RL_CONNECT	NO
.....	

# Etablir des rôles par défaut

- Un utilisateur peut se voir accorder un grand nombre de rôles.
- Un utilisateur peut se voir accorder un rôle par défaut.
- Vous pouvez limiter le nombre de rôles par défaut d'un utilisateur.

```
ALTER USER scott  
        DEFAULT ROLE hr_clerk, oe_clerk;
```

```
ALTER USER scott DEFAULT ROLE ALL;
```

```
ALTER USER scott DEFAULT ROLE ALL EXCEPT  
        hr_clerk;
```

```
ALTER USER scott DEFAULT ROLE NONE;
```



# Activer et désactiver les rôles

- Désactivez un rôle accordé à un utilisateur pour le révoquer temporairement.
- Activez un rôle pour l'accorder temporairement.
- La commande **SET ROLE** permet d'activer et de désactiver les rôles.
- Les rôles par défaut d'un utilisateur sont activés à la connexion.
- Un mot de passe peut être requis pour activer un rôle.

# Activer et désactiver les rôles

```
SET ROLE hr_clerk;
```

```
SET ROLE oe_clerk IDENTIFIED BY order;
```

```
SET ROLE ALL EXCEPT oe_clerk;
```

# Révoquer des rôles accordés à des utilisateurs

- La révocation d'un rôle accordé à un utilisateur requiert l'option **ADMIN OPTION** ou le privilège **GRANT ANY ROLE**.
- Pour révoquer un rôle, utilisez la syntaxe suivante :

```
REVOKE oe_clerk FROM scott;
```

```
REVOKE hr_manager FROM PUBLIC;
```

# Modification de mot de passe

- ❑ Le DBA crée et initialise le mot de passe d'un utilisateur.
- ❑ Le MDP peut être changé à l'aide de la commande **ALTER USER**

## •Syntaxe:

```
ALTER USER nom_utilisateur  
IDENTIFIED BY nouveau_mdp;
```

## Exemple:

```
ALTER USER scott  
IDENTIFIED BY passe1;
```

## Privilèges objet

- Attribuer le droit d'effectuer des opérations sur des tables, vues, séquences ou procédures spécifiques.

Object Privilege	Table	View	Sequence	Procedure
ALTER	√		√	
DELETE	√	√		
EXECUTE				√
INDEX	√			
INSERT	√	√		
REFERENCES	√			
SELECT	√	√	√	
UPDATE	√	√		

## Privilèges objet (suite)

- ❑ Les privilèges objet varient d'objet à objet.
- ❑ Le propriétaire d'un objet possède tous les privilèges sur celui-ci.
- ❑ Le propriétaire d'un objet peut donner n'importe quel privilège objet à un autre utilisateur grâce à la requête **GRANT**.

### •Syntaxe:

```
GRANT privileges [(colonne1,...)]  
ON nom_objet  
TO nom_utilisateur| role | PUBLIC  
[WITH GRANT OPTION];
```

Donne au nouvel utilisateur la possibilité d'accorder les privilèges sur cet objet

# Privilèges objet

## Exercice1

- Attribuer le privilège de sélection (SELECT) de la table « *employe* » aux utilisateur « *amine* » et « *brahim* ».

## Exercice2

- Attribuer le privilège de mise à jour (UPDATE) des colonnes « *fonction* » et « *salaire* » de la table « *employe* » au rôle MANAGER.

## Confirmation des Privilèges

Dictionnaire de données	Description
<b>ROLE_SYS_PRIVS</b>	Les privilèges système donnés à un rôle.
<b>ROLE_TAB_PRIVS</b>	Les privilèges sur les tables donnés à un rôle.
<b>USER_ROLE_PRIVS</b>	Rôles accessibles par l'utilisateur.
<b>USER_TAB_PRIVS_MADE</b>	Les privilèges objet accordés à l'objet de l'utilisateur.
<b>USER_TAB_PRIVS_RECD</b>	Les privilèges objet accordés à l'utilisateur.
<b>USER_COL_PRIVS_MADE</b>	Les privilèges objet accordés sur des colonnes appartenant à l'utilisateur
<b>USER_COL_PRIVS_RECD</b>	Les privilèges objet que possède un utilisateur sur des colonnes
<b>USER_SYS_PRIVS</b>	List des privilèges système accordés à l'utilisateur.



# Retirer les Privilèges

- ❑ Retirer des privilèges à d'autres utilisateurs à l'aide de la commande **REVOKE**.

- **Syntaxe:**

```
REVOKE privilege1, privilege2, ... | ALL  
ON      nom_objet  
FROM user1, user2, ... | role | PUBLIC;
```

# Quelques requêtes utiles

Requête	Description
Conn sys as sysdba;	Se connecter en tant qu'administrateur ' <b>SYS</b> '
Conn user1/passe	Se connecter en tant qu'utilisateur ' <b>user1</b> '
disc	Se déconnecter de la base
Show user	Affiche l'utilisateur connecté
Desc nom_table	Affiche les attributs et leurs types
select * from SESSION_PRIVS ;	Les privilèges de l'utilisateur connecté
SELECT table_name FROM user_tables;	Les tables de l'utilisateur connecté

## Quelques requêtes utiles (2)

Requête	Description
<code>alter user scott account unlock;</code>	Activer le schéma 'SCOTT'
<code>select * from DBA_SYS_PRIVS where grantee='CONNECT' ;</code>	Les privilèges système assignés au rôle <b>'CONNECT'</b>
<code>select * from DBA_SYS_PRIVS where grantee='RESOURCE' ;</code>	Les privilèges système assignés au rôle <b>'RESOURCE'</b>
<code>select * from DBA_SYS_PRIVS where grantee='DBA' order by PRIVILEGE</code>	Les privilèges système assignés au rôle <b>DBA</b>