

SGBD

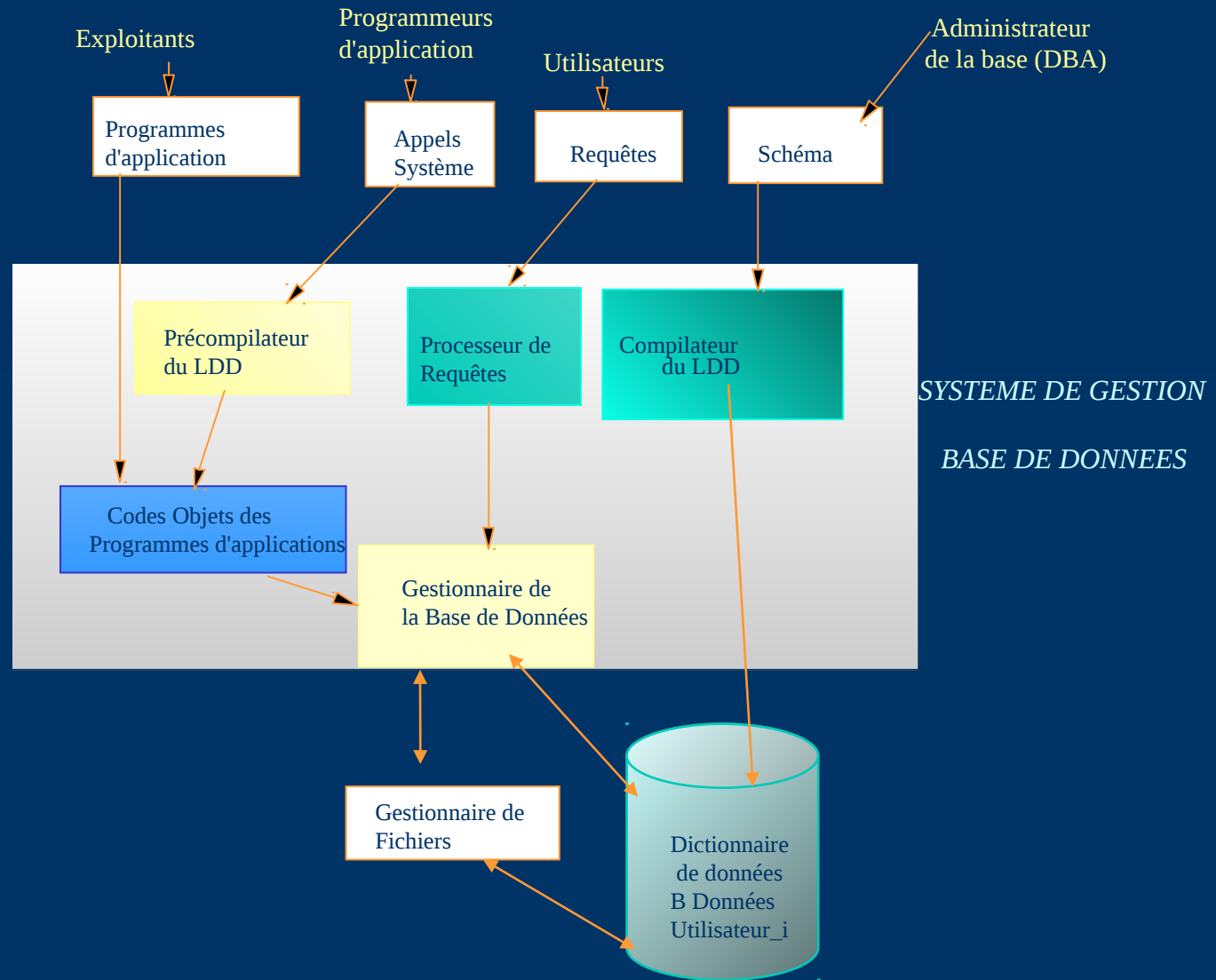
Architecture Oracle

Généralités

Un **SGBD** est un logiciel plus ou moins complexe dont les fonctionnalités principales sont :

- - création de bases de données
- - gestion des données (persistance, interrogation, modification)
- - gestion de la concurrence (multi-utilisateurs)
- - gestion de la sécurité (droits, reprise sur panne)

SGBD : Fonctionnalités générales



Utilisateurs

- **Administrateurs**

 - DBA général

 - Manager de BD utilisateur

- **Utilisateurs « interrogateurs »**

- **Utilisateurs « spécifiques »**

 - Utilisateurs concepteurs d'applications

 - Administrateur de données (Data Manager)

Utilisateurs

- **Administrateurs**
 - DBA général
 - création BD
 - création Utilisateurs
 - Manager de BD utilisateur
 - création schéma
- **Utilisateurs « interrogateurs »**
 - création / utilisation requêtes
- **Utilisateurs « spécifiques »**
 - Utilisateurs concepteurs d'applications
 - création de programmes
 - Administrateur de données (Data Manager)

Rôle DBA

Cas spécifique d 'ORACLE

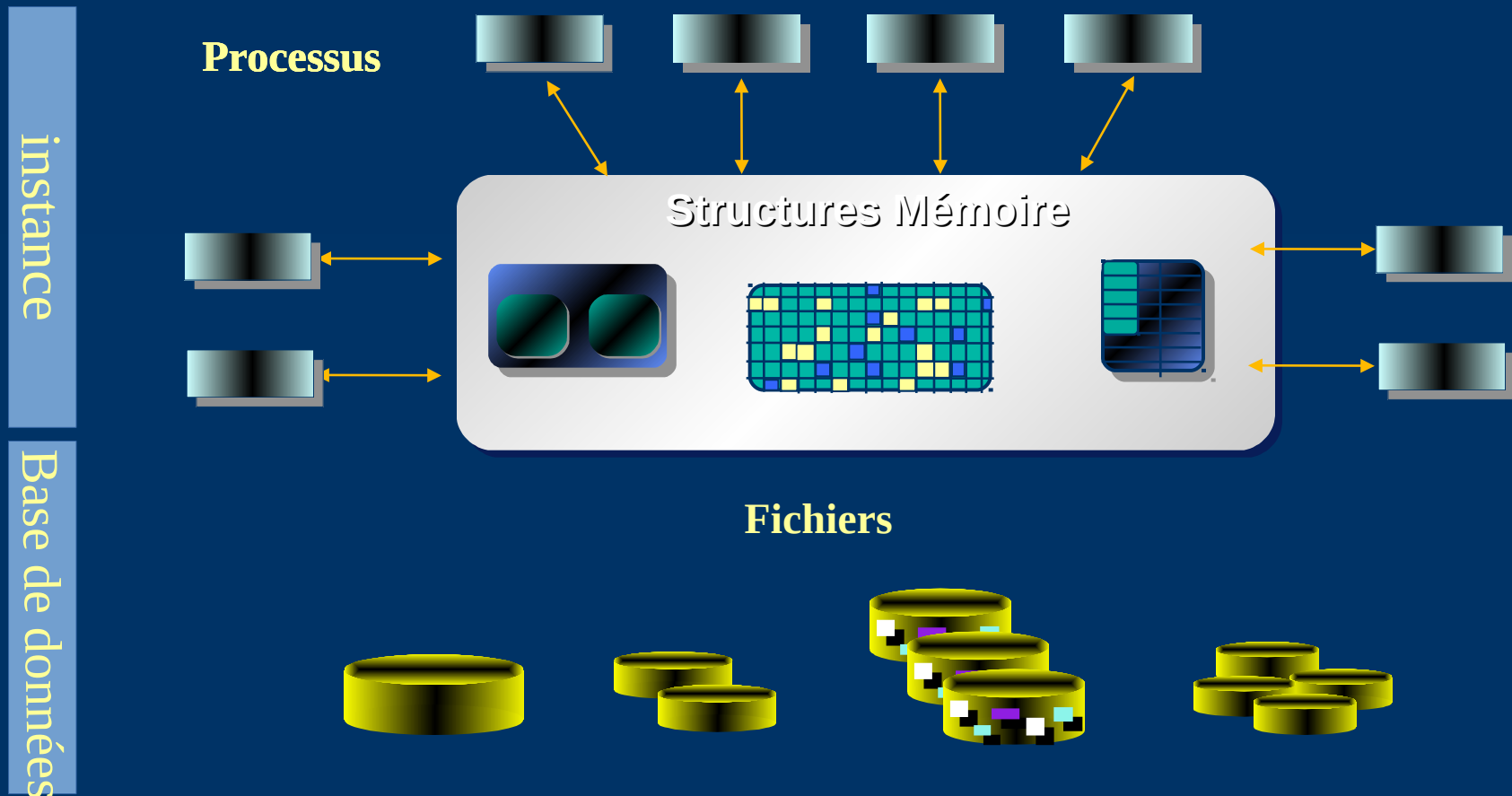
Un administrateur de base de données (DBA) doit comprendre l'architecture du serveur Oracle pour pouvoir gérer une base de données Oracle,

pour cela il doit :

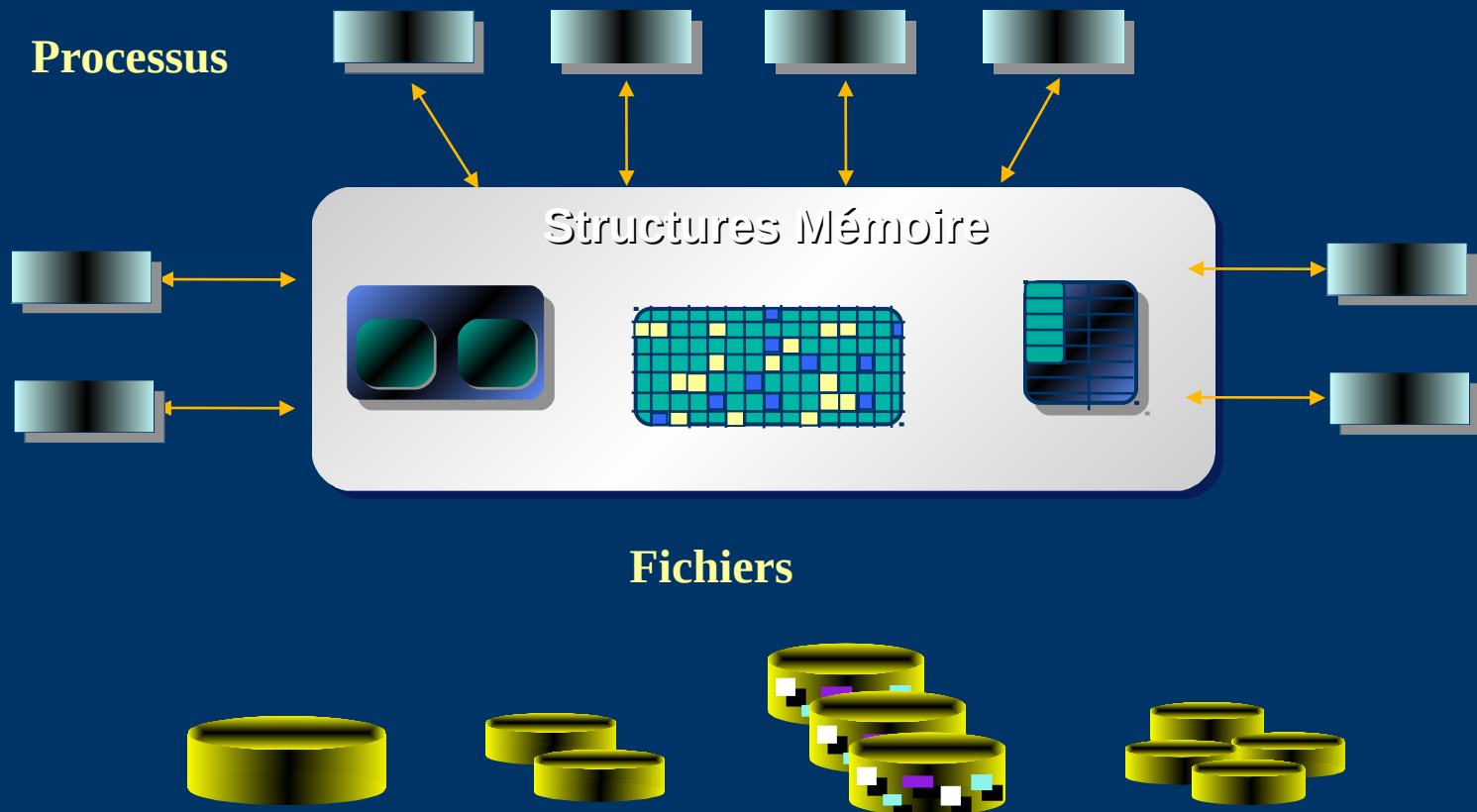
- **Reconnaître les composants de l'architecture d'Oracle.**
- **Identifier le but de chacun des types de composants.**

Serveur Oracle : instance + BD

Mémoire, processus, stockage (permanence des données)



L'instance Oracle



Les Structures Mémoire

Principale structure : System Global Area (SGA)

> 2% de la taille totale de la base données (fichiers).

Répartition :

50% Cache de données (database buffer cache)

40% Shared Pool

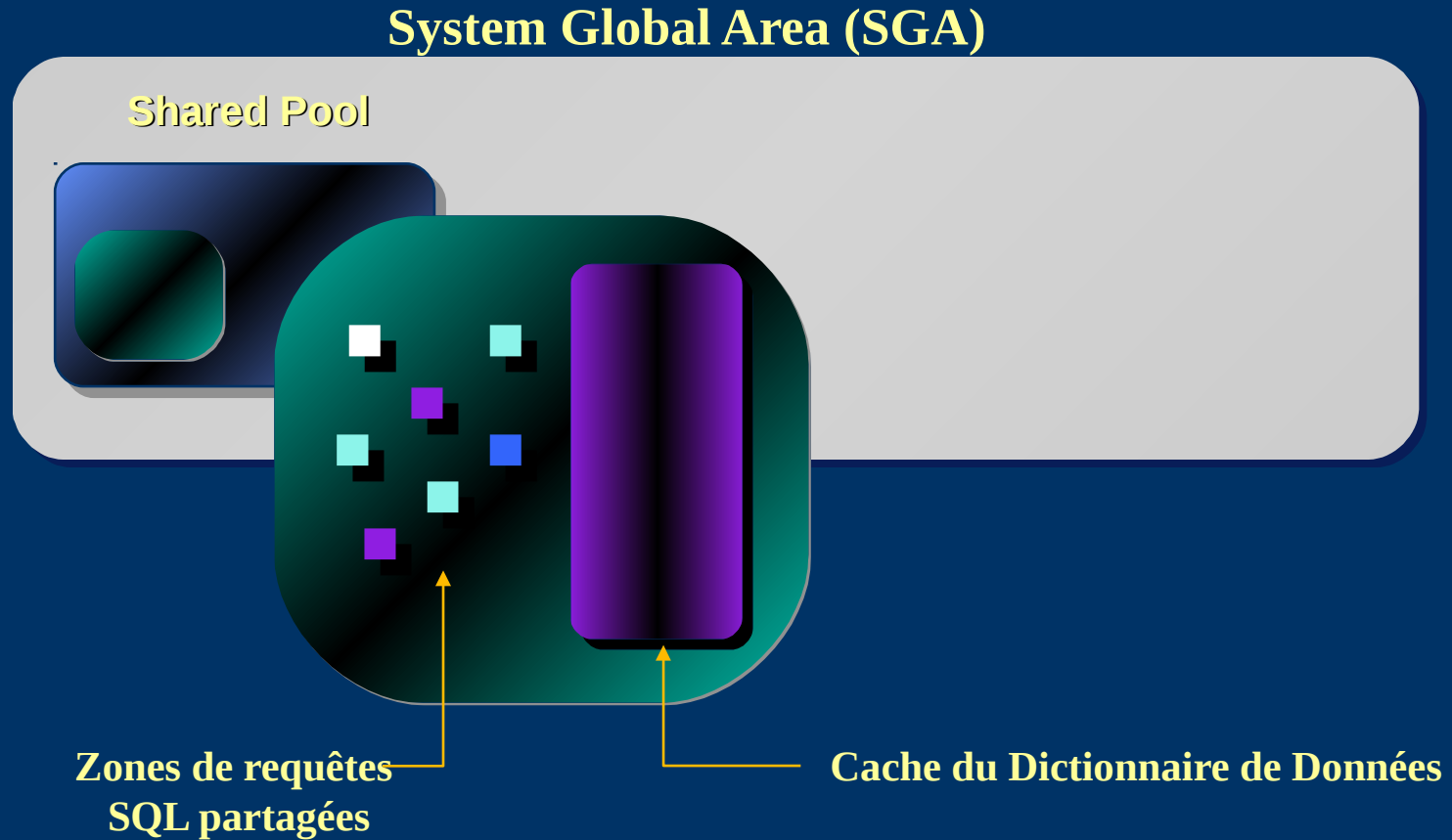
10% Redo log Buffers

Autres structures facultatives : large pool, java pool

La vue V\$SGASTAT : taille des structures SGA par exemple
espace libre dans la SGA :

```
select * from v$sghostat where name='free memory'
```

La Shared Pool : réduire le temps et le coût de l'exécution



Shared pool : 3 structures (2 principales)

Zones mémoire accessibles à tous les usagers

- Dictionary cache : objets du dictionnaire de données
- Library cache : SQL et PL/SQL partagés
 - `select namespace from v$librarycache;`
- Zone globale utilisateurs (UGA) : informations sur les connexions multithread (valable pour une architecture partagée, pour une architecture dédiée l'UGA est dans la PGA).
- données manquantes dans « library cache » comme dans « dictionary cache » plus coûteuse que dans le « data buffer cache » des données => priorité de l'optimisation de la « shared pool » (cf hit ratio, algorithme LRU)

Algorithmes associés aux structures

Exploiter au mieux la mise en cache de blocs de données

- Algorithmes dépendants des données :
 - LRU (Least Recently Used) : les blocs les moins utilisés de manière récente sont enlevés pour faire de la place à de nouveaux blocs (nécessite de tracer les accès)
 - Autres algorithmes : MRU (Most Recently Used), LFU (Least Frequently Used)
- Notions de « cache hit » et « cache miss » : le bloc est en mémoire et il est trouvé = succès de cache (à l'inverse défaut de cache)
- Calcul du cache hit ratio
- Cache versus buffer : dans la notion de cache (ou antémémoire), plutôt une notion de duplication de l'information, dans la notion de buffer (tampon) plutôt une notion d'information qui transite sans impliquer nécessairement une duplication
- Mémoires cache : **accroître les performances** (très important en raison du positionnement OLTP) par le principe de localité :
 - **Localité spatiale :**
 - **Localité temporelle**

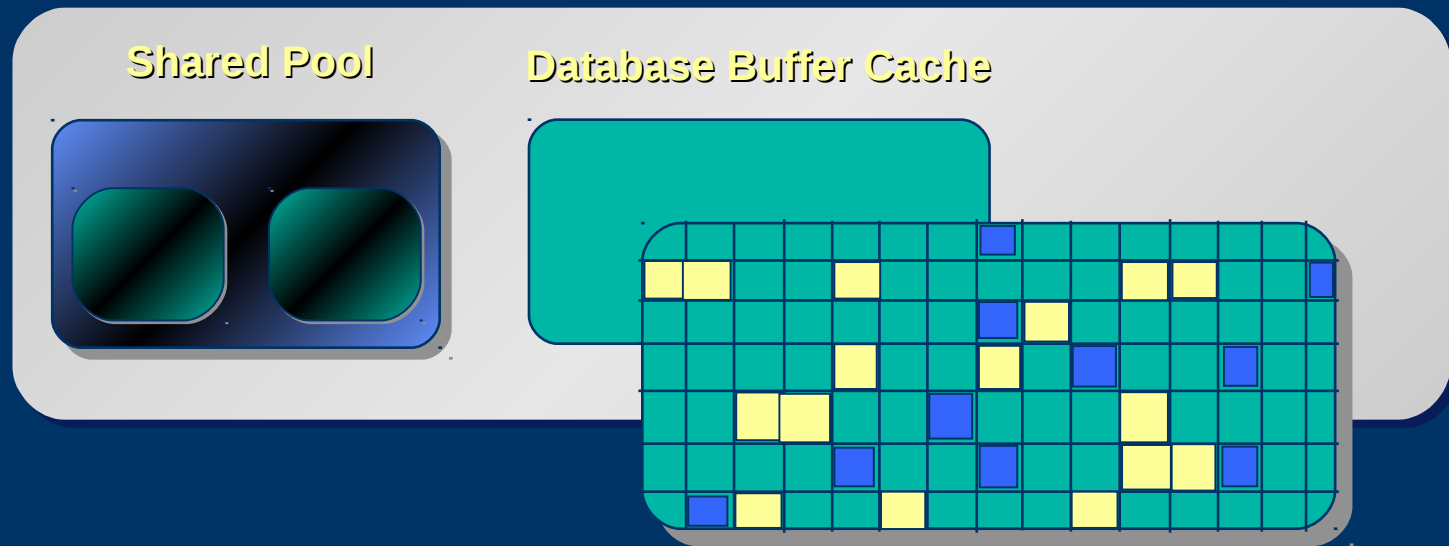
Shared pool : vues

Vues du méta-schéma d'intérêt

- `v$librarycache`, `v$sqlarea`, `v$sql`, `v$sesstat`, `v$sysstat`, `v$sqltext`, `v$db_object_cache`
- **Library cache** : SQL (ordres analysés mais aussi plans d'exécution : intérêt des variables liées) et PL/SQL partagés
- **Zone globale utilisateurs (UGA)** : informations sur les connexions multithread.
- données manquantes dans « library cache » comme dans « dictionary cache » plus coûteuse que dans le « data buffer cache » des données => priorité de l'optimisation de la « shared pool » (cf hit ratio, algorithme LRU)

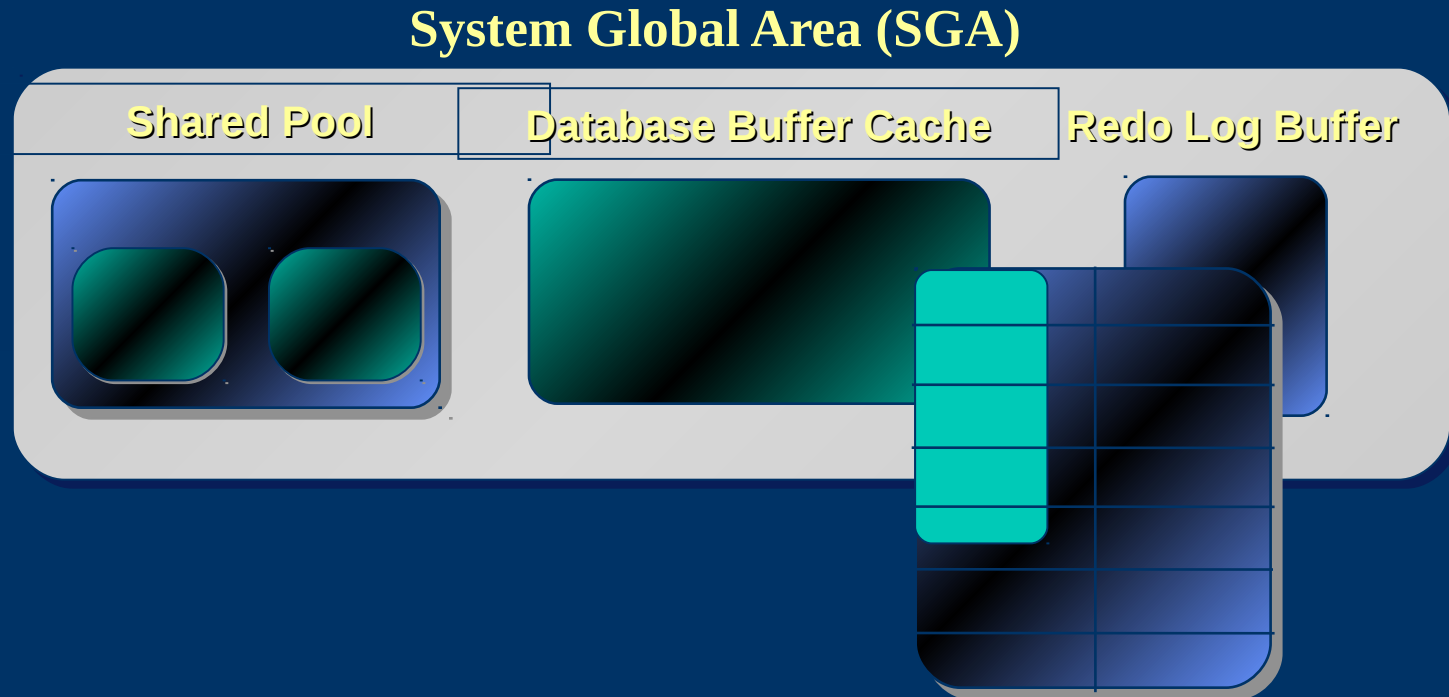
Le Database Buffer Cache

System Global Area (SGA)



Blocs : donnée, index, rollback (ou undo), temporaire
(mnémonique : DIRT)

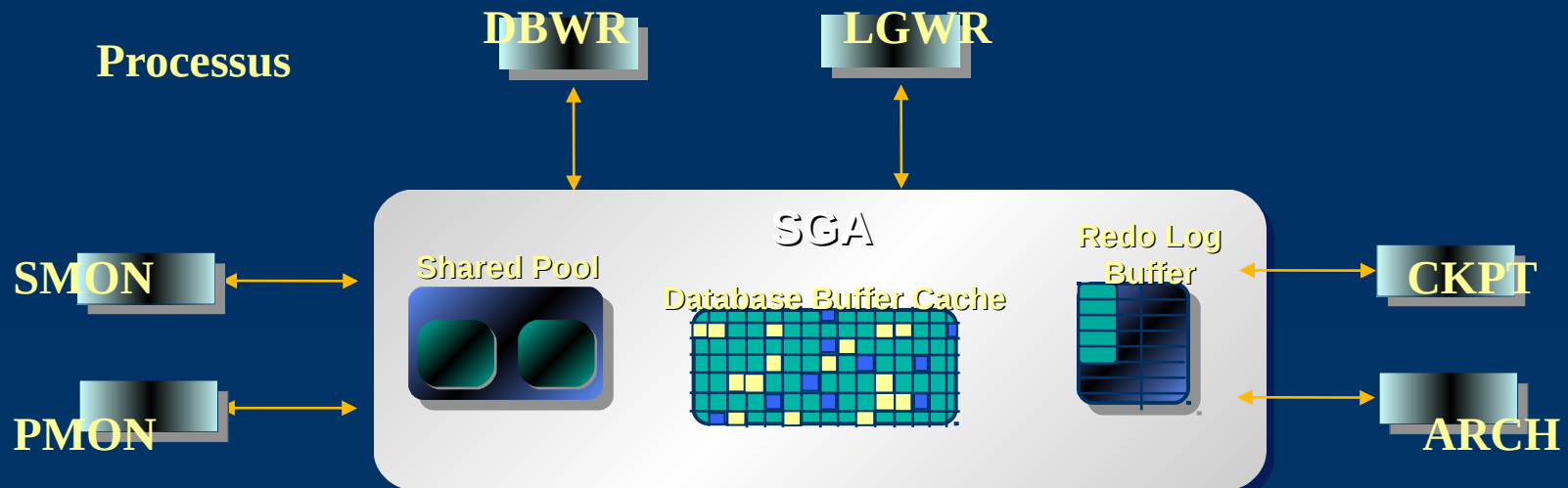
Le Redo Log Buffer (tampon journalisation)



Informations sur les transactions réalisées (bloc : ancienne et nouvelle valeurs)

Les Processus

« Background »



- 4 processus sont obligatoires
 - ☒ **PMON, SMON, DBWR, LGWR** pour exécuter une instance Oracle. Les autres processus sont optionnels
- Si l'un de ces quatre processus échoue, l'instance sera **détruite** et devra être redémarrée.

PMON

PMON : Processus Monitor

- Nettoie les connexions terminées de façon anormale
- Défait les transactions non validées
- Libère les verrous qui avaient été posés par un processus qui s'est terminé en erreur
- Libère les ressources SGA allouées par le processus en erreur

SMON

SMON : Processus System Monitor

- Réalise la restauration automatique d'instance
- Récupère l'espace occupé par des segments temporaires qui ne sont plus utilisés
- Fusionne les zones contiguës d'espace libre dans les fichiers de données

DBWR (écrivain)

DBWR : Processus Database Writer

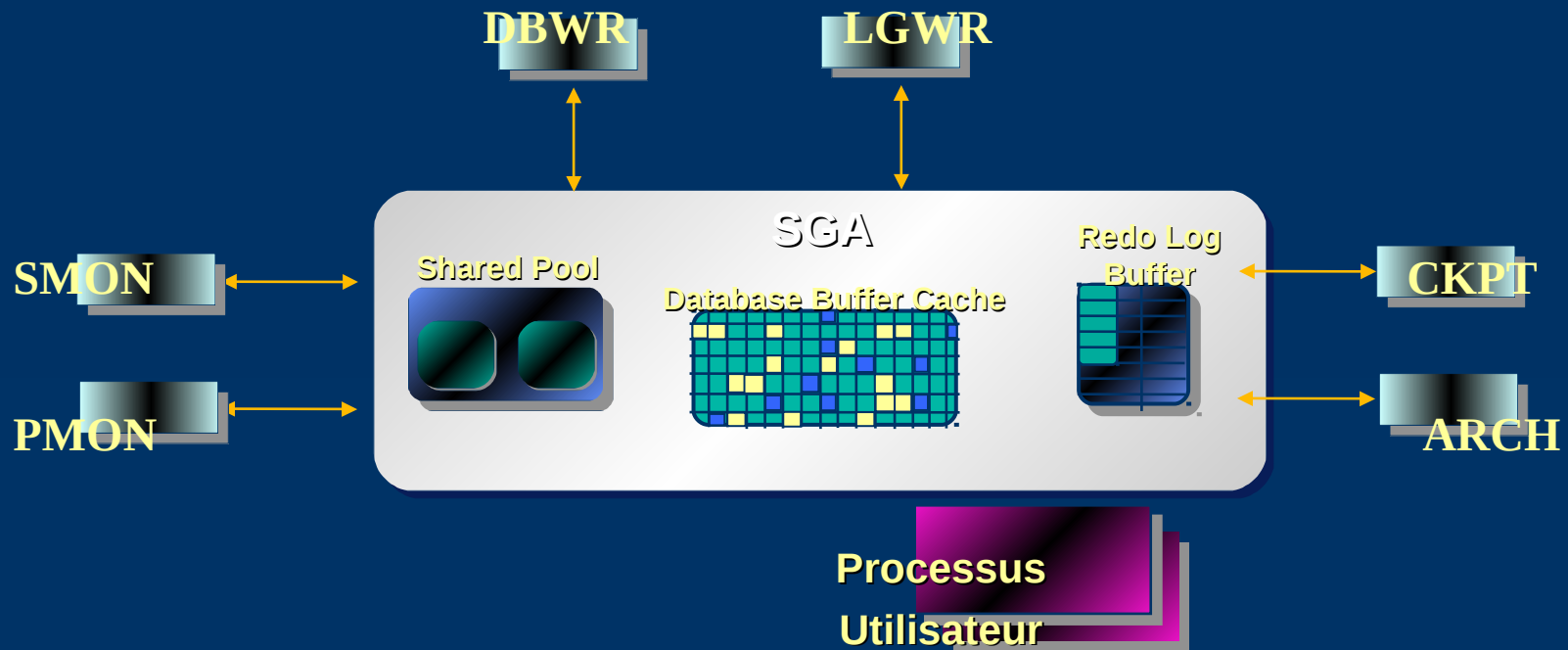
- Le DBWR gère les tampons (buffers) de la base de données pour que les processus serveurs puissent toujours trouver des buffers libres en SGA.
- Le processus DBWR écrit les buffers modifiés vers les fichiers de données.
- Utilise un algorithme (LRU) qui garde les blocs les plus récemment utilisés en mémoire,
- Diffère les écritures en vue d'optimiser les E/S

LGWR (journalisation)

LGWR : Processus Log Writer

- Le LGWR écrit les entrées du buffer redo log dans les fichiers redo log lorsque :
 - Un commit se produit,
 - le buffer est rempli au tiers,
 - le DBWR achève l'écriture des buffers de données lors du checkpoint,
 - Un time-out se produit.

Les Processus Utilisateurs

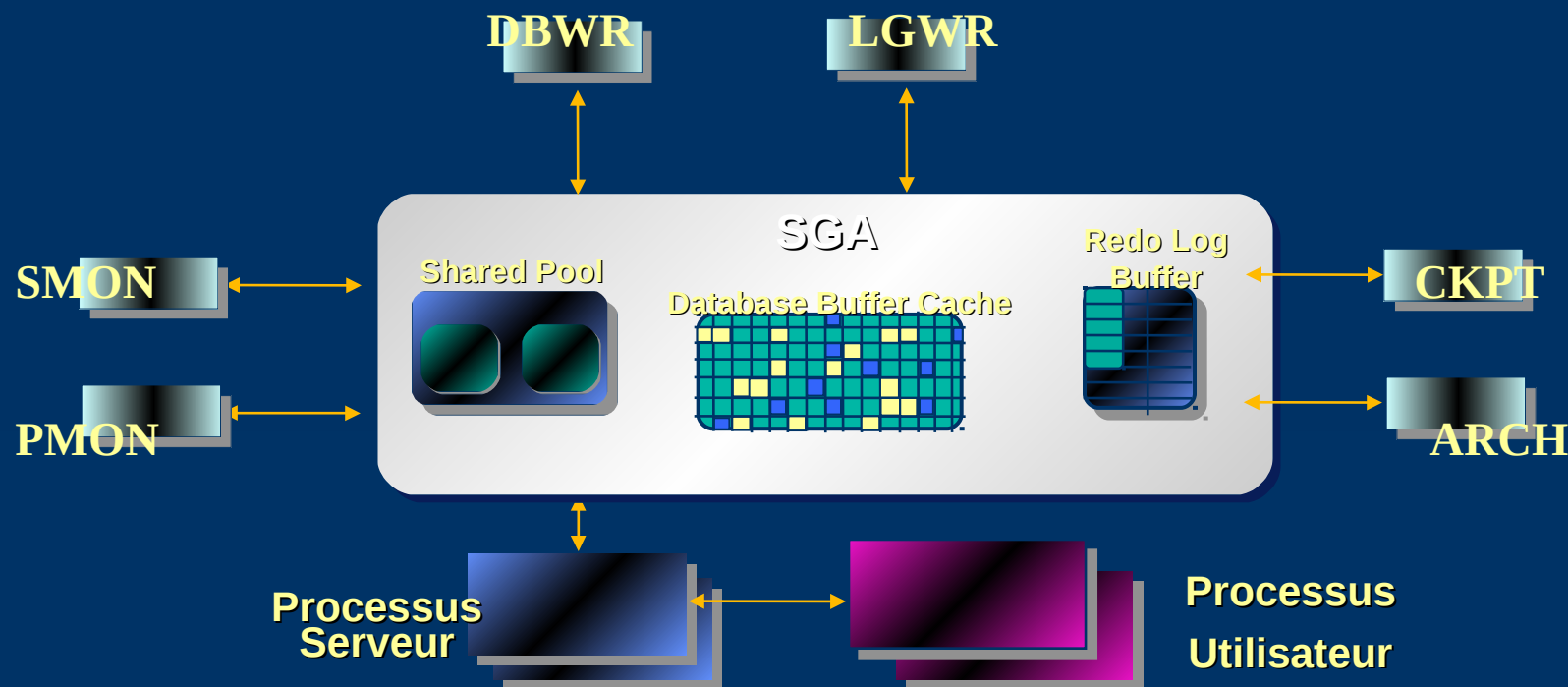


➤ Le processus utilisateur

Exécute l'outil ou l'application et est considéré comme étant le client. Exemple : Sqlplus, Server Manager, Oracle Forms

➤ Il transmet les commandes SQL au processus serveur et en reçoit les résultats

Les Processus Serveurs



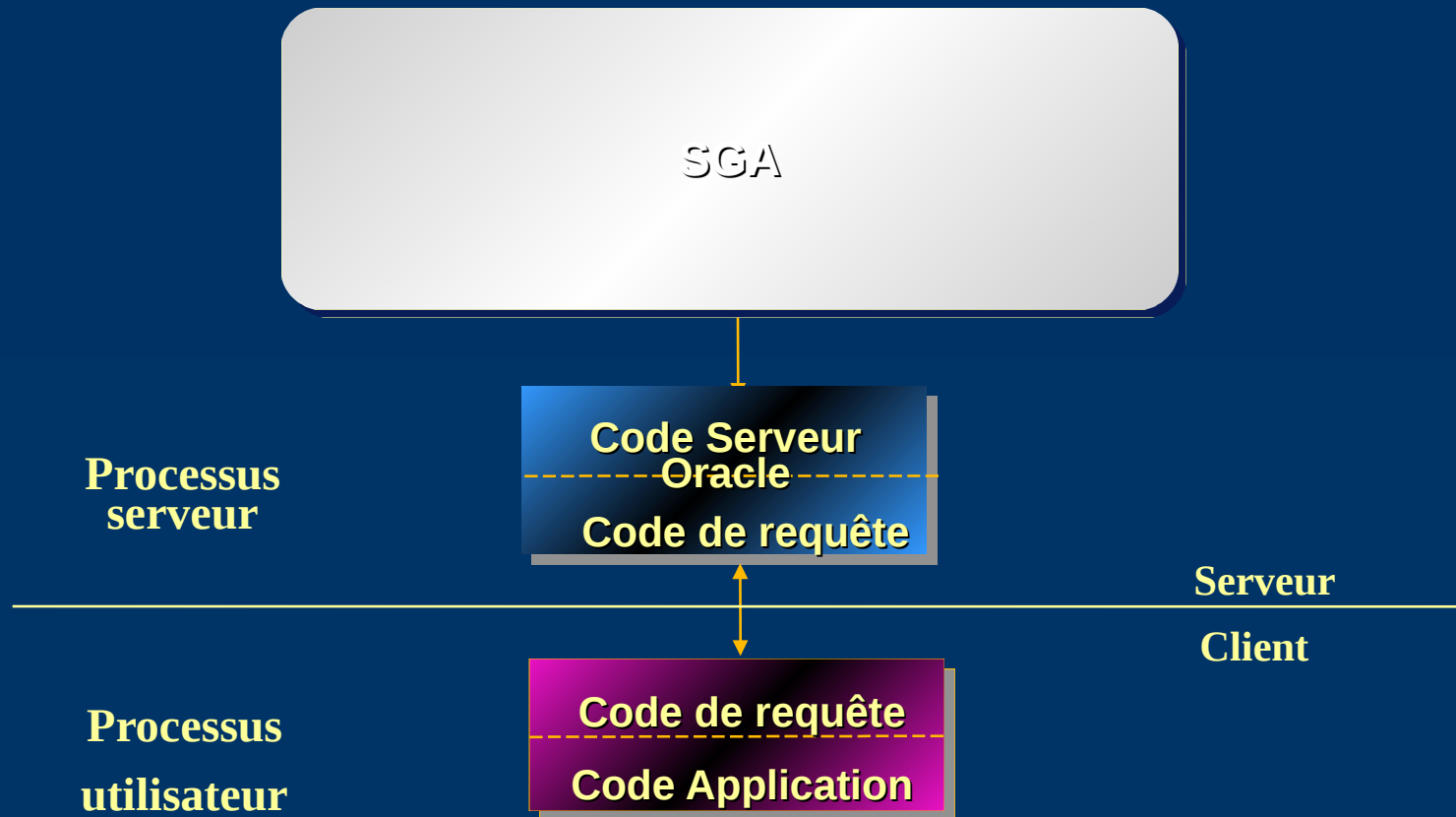
➤ Le processus serveur

Analyse et exécute les commandes SQL

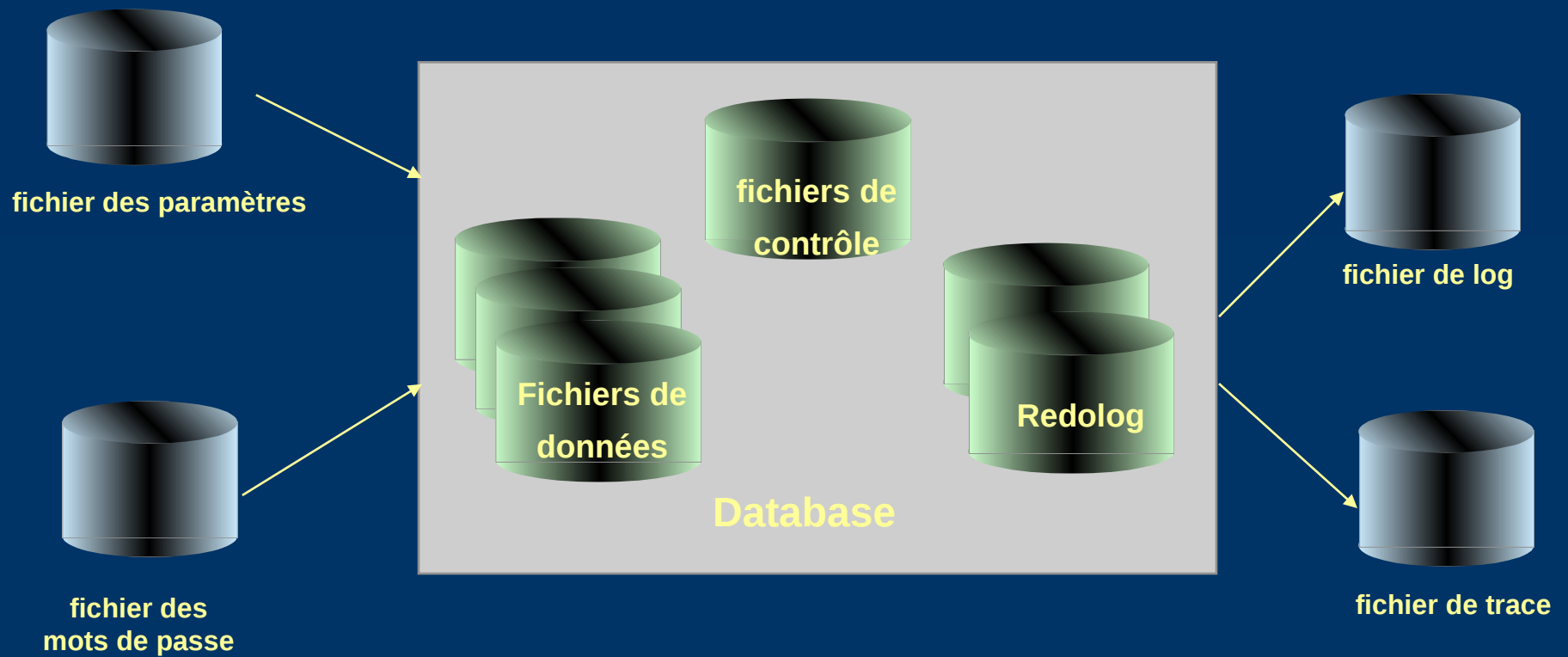
Lit les blocs de données à partir du disque dans les tampons partagés de la base de données de la SGA

Restitue les résultats des commandes SQL au processus utilisateur

Processus Serveurs Dédiés



La structure d'une Base Oracle



Traitement d'une requête Select

Parse (soft ou hard) (1) – Execute (2) –
Fetch (3)

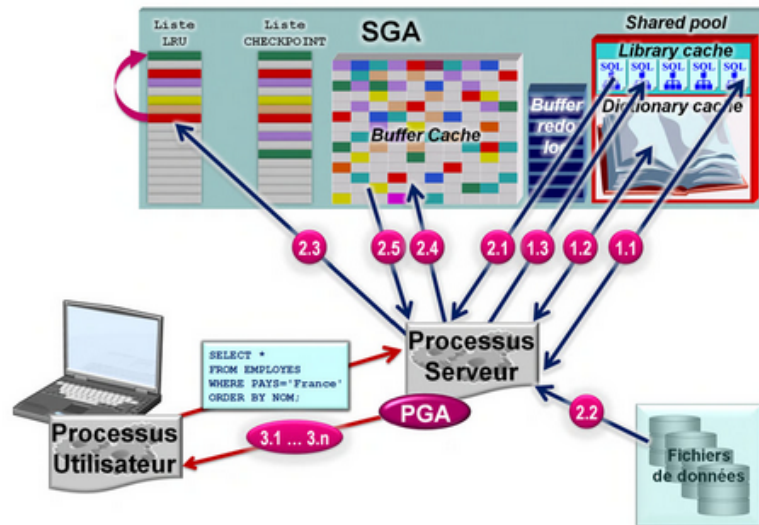


Figure extraite de Oracle 12c Administration E. Bizoi

Traitement d'une requête Select

Parse (soft ou hard)

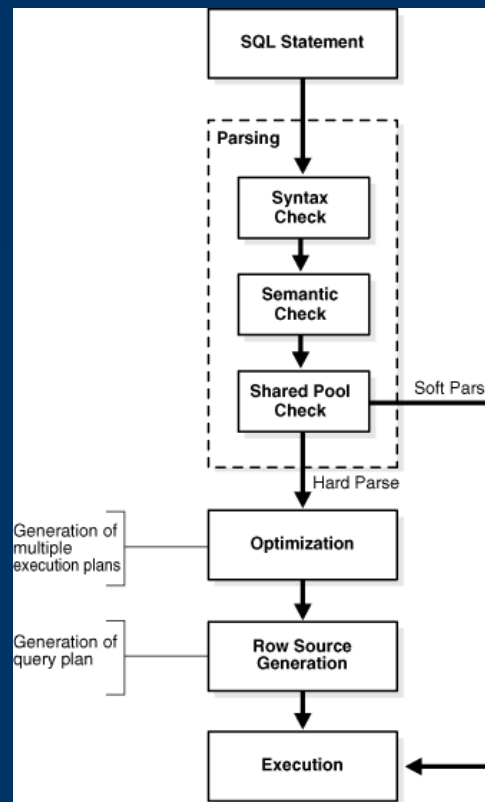
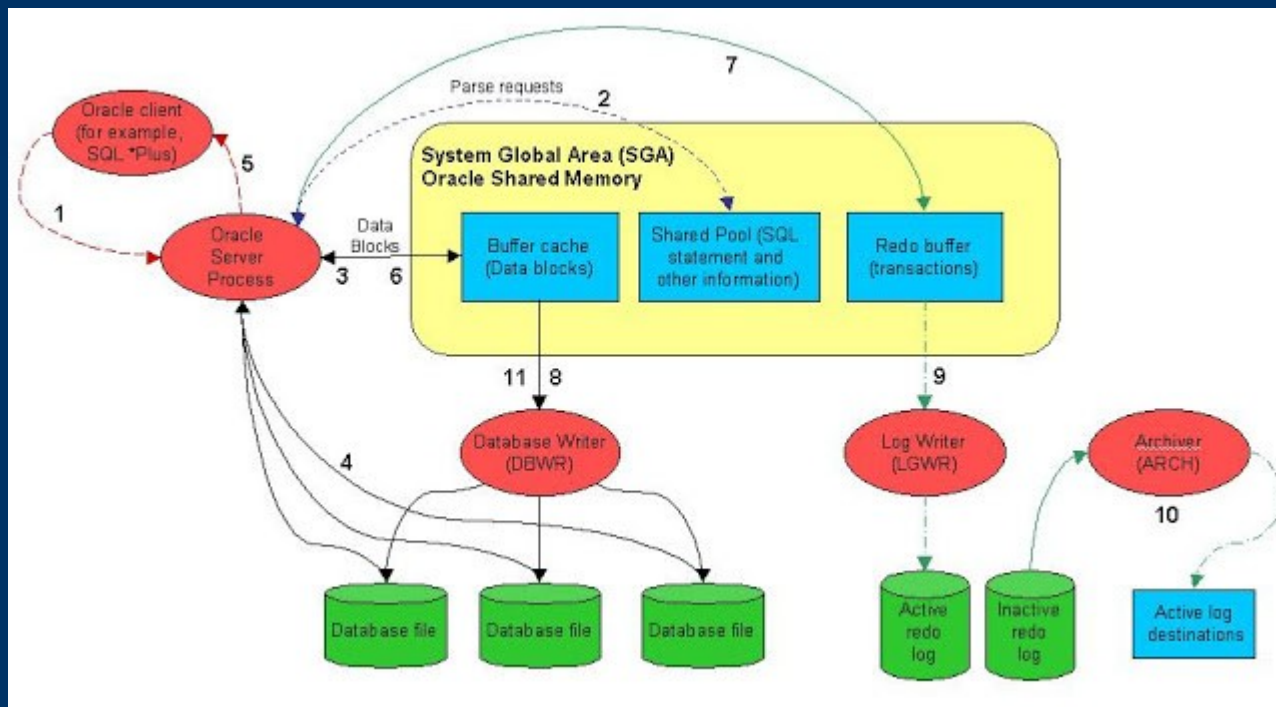


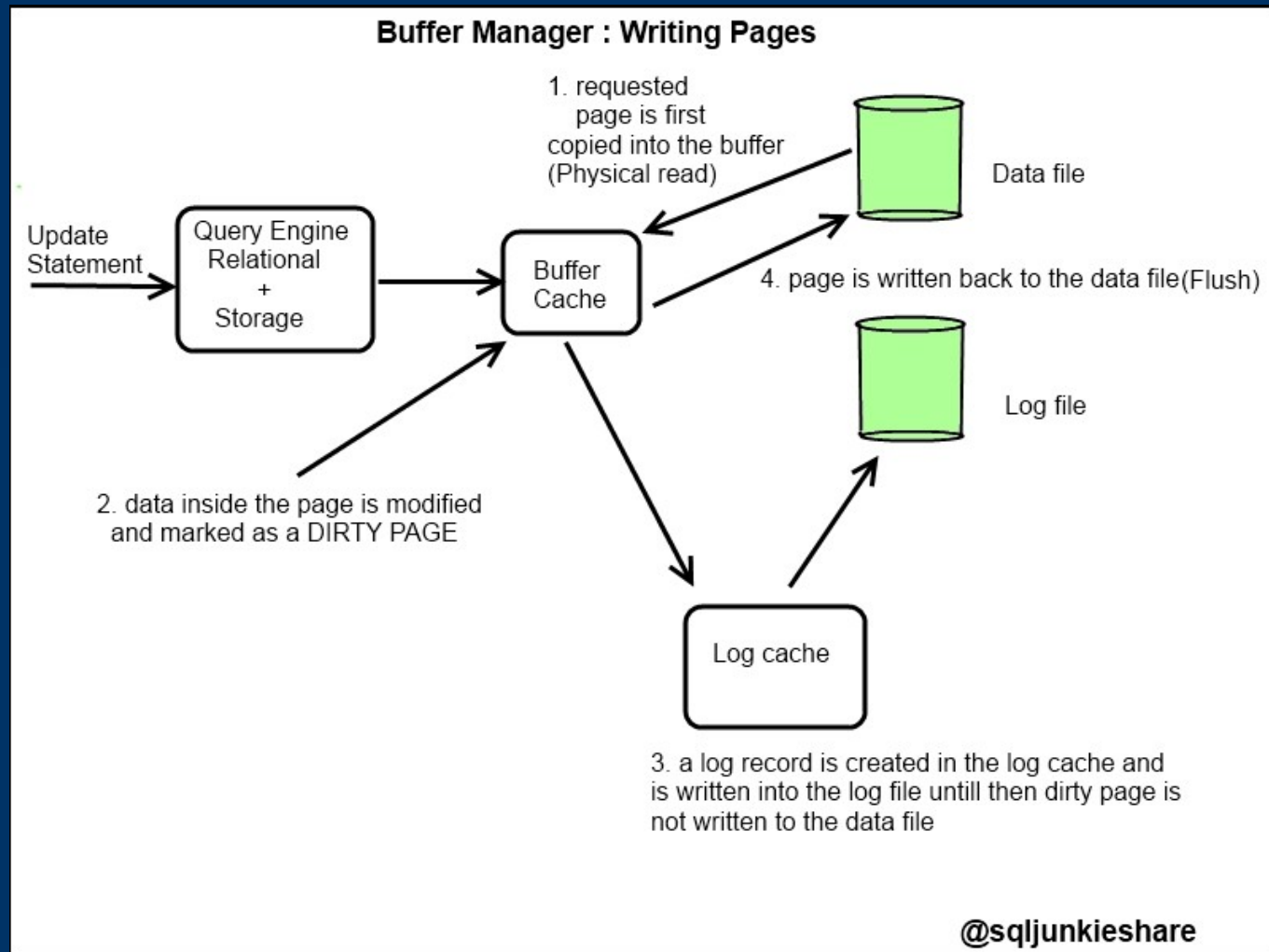
Figure extraite de doc Oracle

Traitement d'une requête Select / update

Parse – Execute – (Fetch pour select)



Traitement d'une requête update



Création d'une Base Oracle

L'administrateur DBA

*En fonction de la future utilisation, détermine
un certain nombre de paramètres
fixant les tailles minimales des zones « mémoire » ,
ainsi que le nombre des fichiers nécessaires
pour le bon fonctionnement*

Création d'une Base Oracle

- **Fichiers de paramètres**
- **Fichiers de données et les fichiers log de la base de données**
- **Fichiers de contrôle (qui comportent le nom de la base de données, les informations nécessaires à la restauration de la base, etc.)**
- **Fichiers de Redo Log (qui contiennent toutes les modifications effectuées sur la base de données, et sont utilisés pour la restauration des données).**
- **Fichier alerte (qui est un journal chronologique de messages et d'erreurs)**
- **Fichier trace (qui contiennent l'information sur les erreurs internes détectées par un processus serveur notamment)**

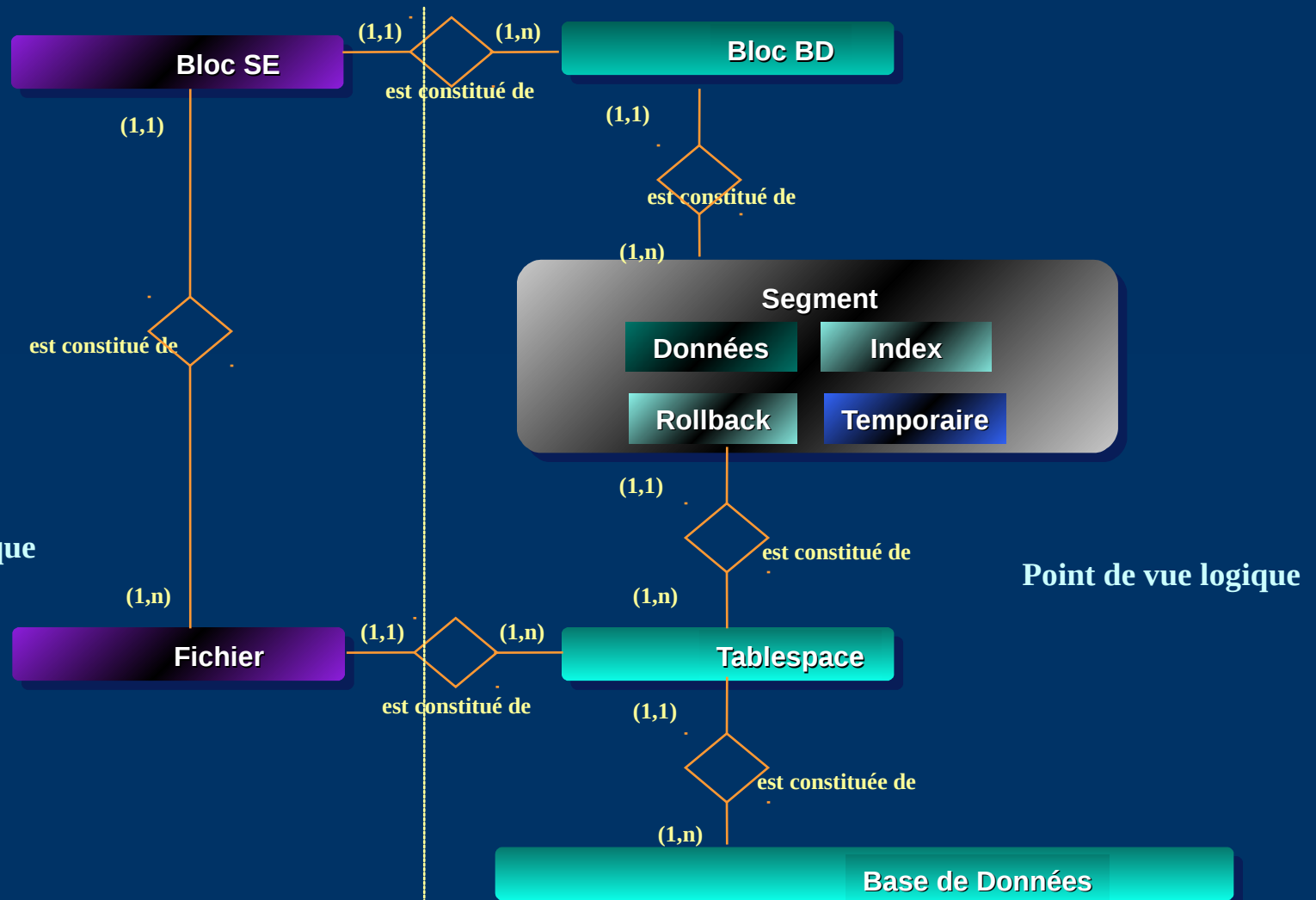
Création d'une Base Oracle

L'administrateur DBA

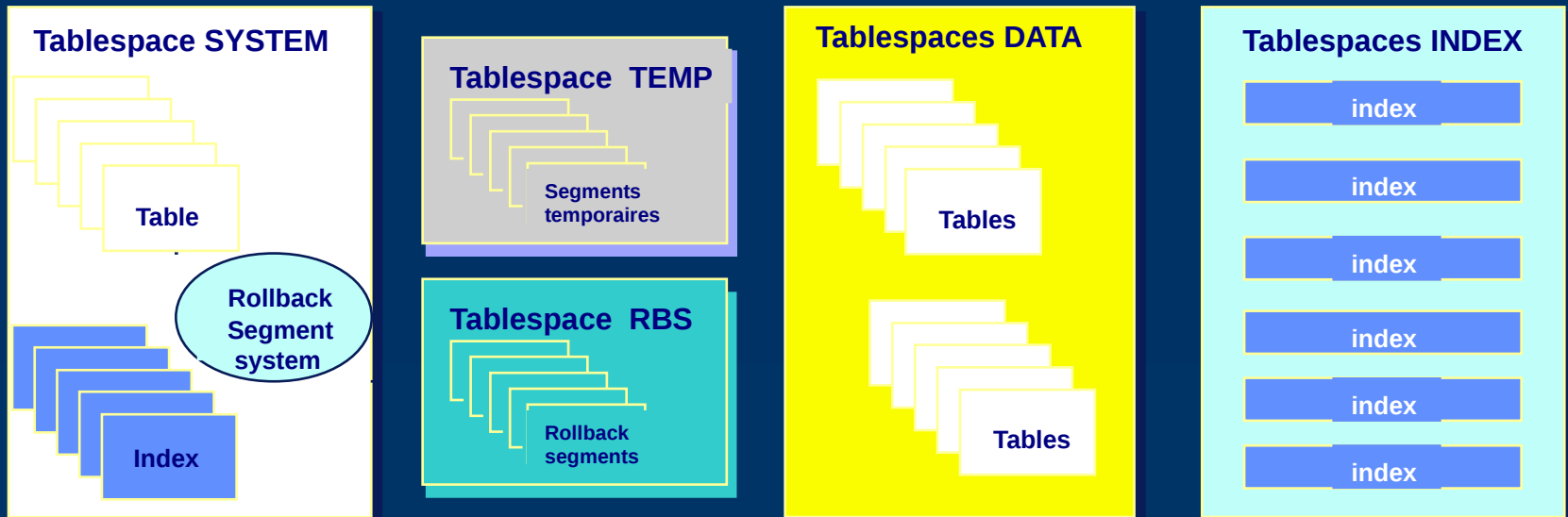
Processus en trois phases

- création de la « base »
- démarrage de l'instance de la « base »
(l'environnement des processus et de la mémoire)
- montage de la base
(elle est alors accessible aux utilisateurs ayant le privilège d'administration)
- ouverture de la base
(elle est alors accessibles aux autres utilisateurs)

L'organisation du stockage d'Oracle (modèle EA)



Structure logique de la base de données



- La base de données Oracle peut être divisée logiquement en un certain nombre de tablespaces distincts.
- Les tablespaces contiennent des segments de base de données
- Le tablespace SYSTEM est indispensable. (Dictionnaire de données, définition des procédures et triggers..)

Les Tablespaces

➤ Le Tablespace SYSTEM

- ☞ Indispensable pour que la Base de Données puisse fonctionner.
- ☐ Contient les informations du dictionnaire de données, les définitions des procédures stockées, des packages, et des triggers de la Base de Données

➤ Les Tablespaces autres (pour stockage des bd utilisateurs)

- ☐ Permettent plus de flexibilité dans l'administration de la Base de Données
- ☐ Consistent en segments de rollbacks, temporaires, tables, index, etc ...

Création d' utilisateurs

L 'administrateur DBA

Doit allouer aux futurs utilisateurs un certain nombre de « droits ou privilèges » en fonction de leurs besoins.

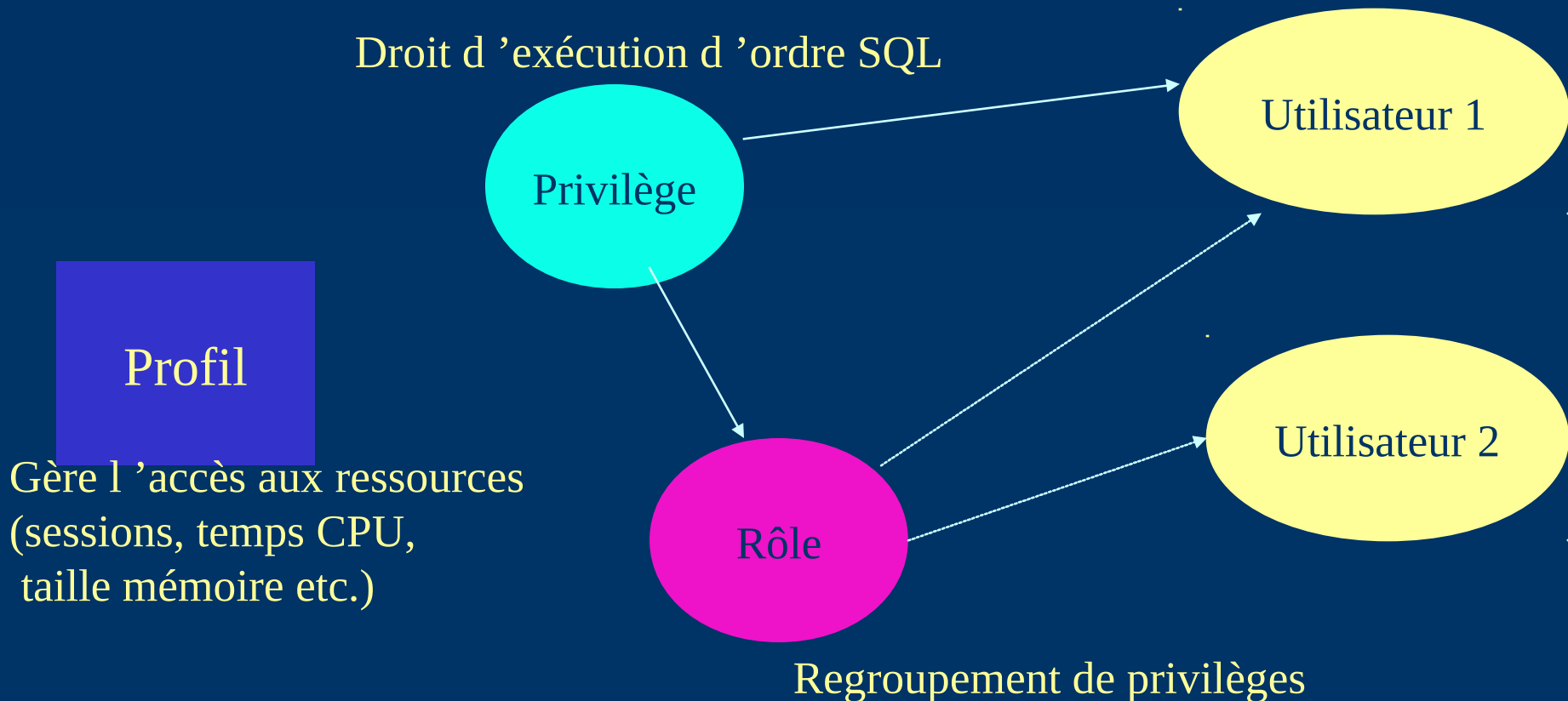
Problèmes de sécurité

si les données sont « sensibles »

elles doivent être protégées des accès « frauduleux »

Création d'utilisateurs

L'administrateur DBA



Gestion de schémas et de données

Data Manager ?

- **Créer le schéma de la base utilisateur**
 - Tables / Vues**
 - Contraintes / Index**
 - Triggers, procédures, packages**
- **Allouer des droits sur ces objets à d'autres utilisateurs**
- **Vérifier la cohérence des données**