

Exercices sur le contenu du module (préparation examen écrit)

1. Architecture Oracle

1.1 Éléments mis en jeu

Une figure incomplète de l'organisation physique d'une base de données Oracle vous est donnée. Vous complèterez les informations manquantes en ce qui concerne les structures, fichiers et processus mis en jeu.

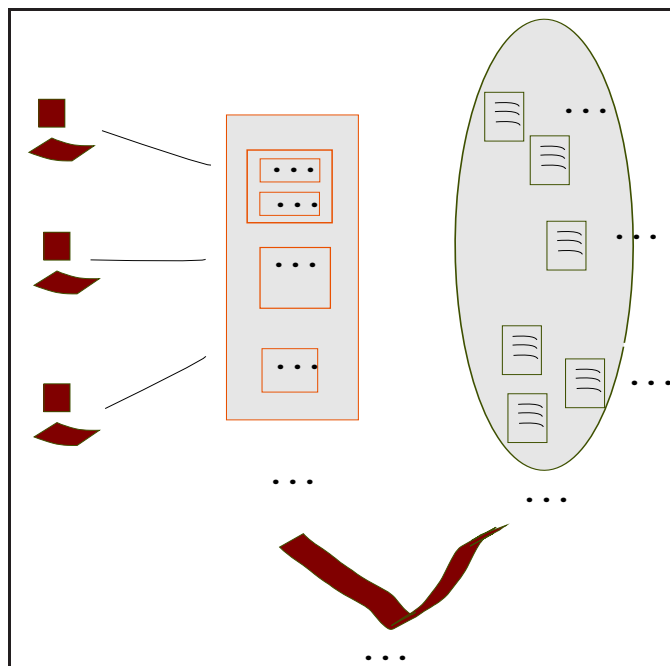


FIGURE 1 – Architecture physique Oracle (structures et processus)

Vous aurez à positionner en complément des composants généraux (serveur, instance, base de données) les éléments suivants :

- structures mémoire de l'instance
 1. SGA
 2. shared pool
 3. database buffer cache

- 4. redo log buffer
- 5. library cache
- 6. dictionary cache
- fichiers de la base de données
 - 1. fichiers de données
 - 2. fichiers de contrôle
 - 3. fichiers journaux
- processus
 - 1. db writer
 - 2. log writer
 - 3. chkpt (checkpoint)
 - 4. smon
 - 5. pmon

1.2 Superviser les structures et éléments mis en jeu

La même figure vous est à nouveau proposée. Vous indiquerez maintenant les vues du méta-schéma qui servent à renseigner sur les différentes structures mémoire et processus d'arrière plan (ou autres processus). Pour vous faciliter la tâche, une liste de vues vous est fournie.

- 1. v\$instance : vue portant sur ...
- 2. v\$database : vue portant sur ...
- 3. v\$process : vue portant sur ...
- 4. v\$bgprocess : vue portant sur ...
- 5. v\$sga : vue portant sur ...
- 6. v\$sgainfo : vue portant sur ...
- 7. v\$sgastat : vue portant sur ...
- 8. v\$thread : vue portant sur ...
- 9. v\$log : vue portant sur ...
- 10. v\$sql, v\$sqlarea, v\$sqltext : vues portant sur ...
- 11. v\$datafile : vue portant sur ...
- 12. v\$logfile : vue portant sur ...
- 13. v\$controlfile : vue portant sur ...
- 14. v\$bh : vue portant sur ...
- 15. v\$rowcache : vue portant sur ...

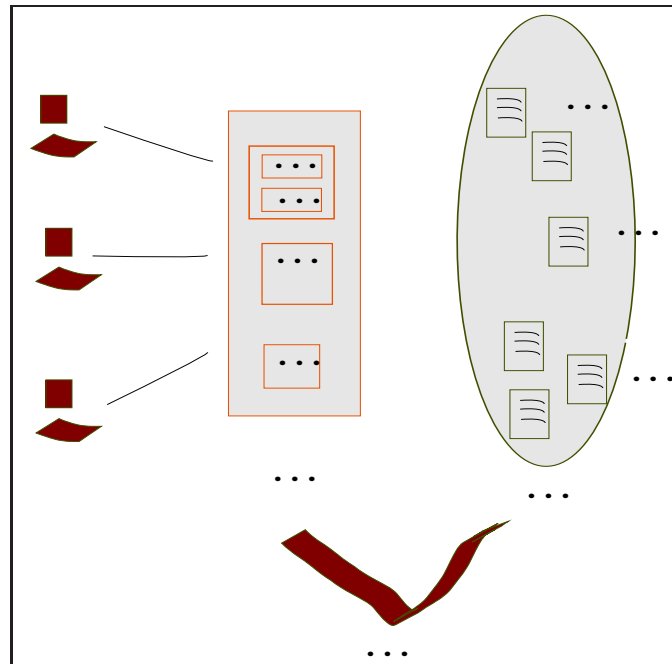


FIGURE 2 – Architecture physique Oracle (vues du méta-schéma associées)

1.3 Vues statiques et dynamiques du dictionnaire de données

Vous expliquerez, dans un premier temps, les différences entre les vues statiques et les vues dynamiques. Pour ce qui concerne les vues statiques, vous expliquerez dans un second temps, la différence entre les vues préfixées respectivement par `USER_`, `ALL_` et `DBA_`.

- Vous souhaitez connaître les tables qui vous sont disponibles en consultation. Quelle est la vue qu'il vous faut interroger à ce sujet ?
 1. `USER_TABLES`
 2. `DBA_TABLES`
 3. `ALL_TABLES`
- Vous souhaitez connaître les tables qui vous sont accessibles en consultation mais qui ne vous appartiennent pas. Donner un exemple de requête qui vous permettrait d'avoir cette information
- A quoi correspond la notion de session ? Quelles sont les notions qui vous semblent associées à session ?
- Que retourne la requête suivante ?


```
select status, count(*) as cxns from V$SESSION group by status;
```

1.4 Questions d'ordre général sur l'architecture

Exploitez les vues du dictionnaire de données vous permettant de répondre aux questions suivantes :

1. sur quelle instance suis je connecté(e) ?
2. Que donne comme informations complémentaires la requête suivante ?


```
select instance_name, host_name, version, startup_time from v$instance;
```
3. Sur quelle base de données pointe l'instance ? et est ce que le mode archivage est activé

4. Que donne comme informations complémentaires la requête suivante ?

```
alter session set NLS_DATE_FORMAT = 'mm-dd-yyyy HH24:mi:ss';
select name, created, log_mode, SCN_TO_TIMESTAMP(current_scn)
from v$database;
```

5. Traduisez en langue naturelle la requête suivante

```
select addr, username, program from v$process
where addr in (select paddr from v$bgprocess);
```

6. quels sont les nom et taille des principales structures mémoire de la SGA rappeler brièvement les rôles dévolus à chacune de ces structures mémoires
7. chassez l'intrus : laquelle de ces vues n'est pas associée à la "shared pool" Les vues dynamiques de performance associées à la mémoire partagée sont : v\$rowcache, v\$librarycache, v\$sqlarea et v\$lock
8. Comment afficher la taille du bloc de données ?

1.5 Organisation logique

1. A quoi correspond la notion d'espace de table (tablespace) ? En quoi se décompose t'elle ?
2. Quelle est la structure mémoire qui est la plus impactée par l'organisation en tablespaces ?
3. Comment connaître le nom de l'espace de table dans lequel les tables de votre schéma utilisateur sont organisées ? idem pour les index ?
4. Quels sont les types de segments les plus habituels ?
5. Comment savoir à quel fichier de données correspond un espace de table ?
6. Quel est le paquetage qui vous renseigne sur le bloc de données qui contient un tuple précis d'une table en particulier ?

1.6 Library Cache

1. Donnez votre compréhension de la requête qui suit

```
set linesize 300

SELECT
Sql_FullText, (cpu_time/100000) "Cpu Time (s)"
, (elapsed_time/1000000) "Elapsed time (s)"
, fetches, buffer_gets, disk_reads, executions
FROM v$sqlarea
WHERE
Parsing_Schema_Name = USER
AND ROWNUM < 50
ORDER BY 3 DESC;
```

2. Index

1. quel est le rôle d'un index ? Quelle est la structure d'index la plus exploitée dans les SGBDR ?

2. Vous expliquerez l'ordre suivant :

```
set verify off
select index_name from dba_indexes where upper(table_name) = upper(&matable);
```

3. Vous disposez d'une table EMP qui comporte un attribut NUM sur lequel est posé une contrainte de clé primaire.

(a) EMP se voit appliquer un index unique ? (justifier votre réponse)

i. OUI

ii. NON

(b) Des requêtes de consultation vous sont données. Lesquelles vont bénéficier de l'index ?

i. `select num from emp;`

ii. `select nom from emp where num = 20;`

iii. `select nom, fonction from emp;`

iv. `select nom from emp where num > 10;`

(c) remplacer EMP par une table indexée (IOT)

i. donner l'ordre de création de cette table IOT

ii. quelles seraient les requêtes précédentes qui seraient rendues encore plus efficaces par cette nouvelle organisation ?

3. Transactions

Vous travaillerez sur les transactions et les verrous actifs.

1. Donnez votre compréhension de la requête qui suit

```
select id1, id2, request,
decode(lmode,
 0, 'None(0)',
 1, 'Null(1)',
 2, 'Row Share(2)',
 3, 'Row Exclu(3)',
 4, 'Share(4)',
 5, 'Share Row Ex(5)',
 6, 'Exclusive(6)') lmode, sid from v$lock;
```

2. Expliquer la séquence suivante (placez vous en début de transaction)

```
select * from employe;
select username, v$lock.sid, id1, id2, lmode, request, block, v$lock.type
from v$lock, v$session
where v$lock.sid = v$session.sid and v$session.username = USER;
```

3. Expliquer la nouvelle séquence

```
insert into from employe (num) values (55555);
select username, v$lock.sid, id1, id2, lmode, request, block, v$lock.type
from v$lock, v$session
where v$lock.sid = v$session.sid and v$session.username = USER;
```

4. surcouche procédurale PL/SQL

4.1 Déclencheurs

Trois ordres de déclencheurs vous sont donnés. Un seul ordre est correct. Vous indiquerez lequel et vous expliquerez pourquoi les deux autres ne le sont pas.

```
create or replace trigger t1 before update
on commune
begin
:new.nom_com := lower(:old.nom_com) ;
end t1;
/
```

```
create or replace trigger t1 before update
on commune
for each row
begin
:new.nom_com := upper(:old.nom_com) ;
end t1;
/
```

```
create or replace trigger t1 after update
on commune
for each row
begin
:new.nom_com := upper(:old.nom_com) ;
end t1;
/
```

5. Fonctions et procédures

Expliquez la différence entre une procédure et une fonction ?

Répondez au préalable aux questions suivantes (plusieurs réponses possibles)

1. Les procédures et fonctions doivent retourner des valeurs
2. Les procédures et fonctions peuvent retourner des valeurs
3. Les procédures peuvent retourner des valeurs
4. Les fonctions doivent retourner des valeurs

6. Bases de données réparties (et solution Db Link Oracle)

6.1 Question 1

Expliquer les grands principes de l'approche ascendante. A cet effet, vous définirez la notion de schéma global et de vue intégrante (vous pouvez vous aider d'un schéma et/ou d'un exemple illustratif).

6.2 Question 2

Quelle est la différence entre une vue et une vue matérialisée ? Quel peut être l'apport d'une vue matérialisée dans le contexte des bases de données réparties ? Vous expliquerez les ordres suivants :

```
create database link lk_M connect to user1 identified by user1
      using 'licence';
```

```
create materialized view test as select * from commune@lk_M;
```

6.3 Question 3

Nous nous concentrons sur la base de données relative aux communes françaises déjà étudiée en TP. Les relations Commune, Département et Region sont à l'origine dans une seule base de données. Chaque nouvelle grande région française souhaite disposer des informations concernant les communes sous leur autorité. Vous ferez une proposition de fragmentation de la relation Commune dans un contexte d'approche descendante qui répond à cette relative autonomie (donnez le schéma de fragmentation).

7. Exercices sur l'énoncé : la parfumerie

7.1 Enoncé

La base de données considérée, s'attache à décrire le monde du parfum. La gestion qui en est faite, est très partielle, et porte sur les types d'entité suivants. Un type d'entité **Marque** est identifié par un nom de marque, et est caractérisé par un siège social et un attribut nommé couturier qui indique si la marque est aussi une maison de couture (valeurs parmi l'ensemble {o, n} pour oui ou non). Un type d'entité **Parfum** est identifié par les attributs nom de parfum et nom de marque, et est caractérisé par un nom de créateur, une date de création, un public cible (valeurs parmi l'ensemble {homme, femme, mixte}), et un type de parfum (valeurs parmi l'ensemble {boise, floral, oriental, hesperide, fougere, ambre, chypre, cuir}). Un type d'entité **Fragrance** (odeur agréable) est identifié par son nom et est caractérisé par une origine (valeurs parmi l'ensemble {animal, vegetal, synthétique}). Un parfum **se compose** de plusieurs fragrances.

7.2 Schémas EA et Relationnel

Les schémas conceptuel et relationnel vous sont donnés. Un jeu de tuples est fourni, en annexe, à titre illustratif.

Les attributs portant les contraintes de clés primaires sont en gras. Les contraintes de clés étrangères vous sont données sous la forme de contraintes d'inclusion. Les types des attributs vous sont également indiqués.

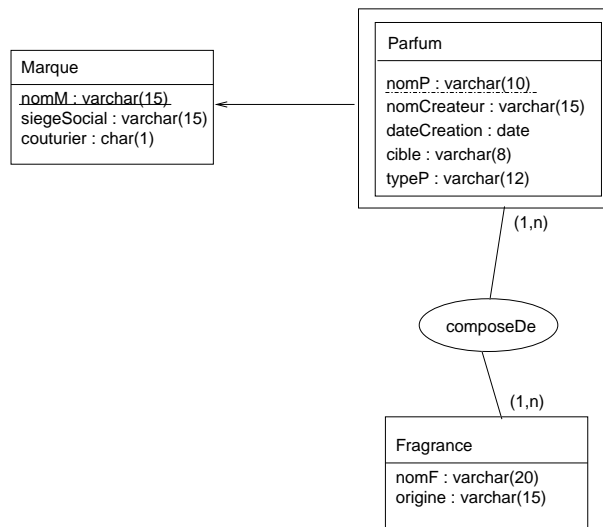


FIGURE 3 – Modèle entité association Parfumerie

- Marque(**nomM** varchar(15), siegeSocial varchar(15), couturier char(1))
- Parfum(**nomP** varchar(10), **nomM** varchar(15), nomCreateur varchar(15), dateCreation date, cible varchar(8), typeP varchar(12))
avec Parfum(nomM) ⊆ Marque(nomM)
- Fragrance(**nomF** varchar(20), origine varchar(15))
- ComposeDe(**nomP** varchar(10), **nomM** varchar(15), **nomF** varchar(20))
avec ComposeDe(nomF) ⊆ Fragrance(nomF)
avec ComposeDe(nomP, nomM) ⊆ Parfum(nomP, nomM)

7.3 Exercice optimisation

7.4 Index

Vous répondrez (avec une justification courte) aux questions suivantes :

1. quels sont les index déjà présents au sein du schéma de la base de données, et sur quels attributs, sont ils implicitement posés ?
2. sur quels autres attributs, poseriez vous d'autres index ? Donner les ordres de création d'index correspondants
3. expliquer la requête suivante et les résultats obtenus

```
select index_name, uniqueness, table_name, num_rows from user_indexes;
```

INDEX_NAME	UNIQUENESS	TABLE_NAME	NUM_ROWS
COMPOSEDE_PK	UNIQUE	COMPOSEDE	11
FRAGRANCE_PK	UNIQUE	FRAGRANCE	13
PARFUM_PK	UNIQUE	PARFUM	2
MARQUE_PK	UNIQUE	MARQUE	2

7.5 Plan d'exécution d'une requête

1. expliquer l'intérêt de la requête suivante et décrivez le plan d'exécution


```

explain plan for SELECT /*+ use_nl(p m) */ m.nomM
FROM marque m, parfum p
WHERE m.nomM=p.nomM;

select plan_table_output from table (dbms_xplan.display()) ;

Plan hash value: 3812789126

```

```

-----
| Id | Operation      | Name      | Rows | Bytes | Cost (%CPU)| Time |
-----
|  0 | SELECT STATEMENT |           |     2 |    36 |    2 (0)| 00:00:01 |
|  1 | NESTED LOOPS      |           |     2 |    36 |    2 (0)| 00:00:01 |
|  2 | INDEX FAST FULL SCAN| PARFUM_PK |     2 |    18 |    2 (0)| 00:00:01 |
|* 3 | INDEX UNIQUE SCAN | MARQUE_PK |     1 |     9 |    0 (0)| 00:00:01 |
-----

```

PLAN_TABLE_OUTPUT

Predicate Information (identified by operation id):

```

-----
3 - access("M"."NOMM"="P"."NOMM")

```

2. même question pour la requête suivante

```

explain plan for SELECT /*+use_hash(m p) */ m.nomM FROM marque m, parfum p
WHERE m.nomM=p.nomM;

select plan_table_output from table (dbms_xplan.display()) ;

Plan hash value: 1371628531

```

```

-----
| Id | Operation      | Name      | Rows | Bytes | Cost (%CPU)| Time |
-----
|  0 | SELECT STATEMENT |           |     2 |    52 |     3 (34)| 00:00:01 |
|* 1 | HASH JOIN        |           |     2 |    52 |     3 (34)| 00:00:01 |
|  2 | INDEX FULL SCAN  | MARQUE_PK |     2 |    26 |     1 (0)| 00:00:01 |
|  3 | INDEX FULL SCAN  | PARFUM_PK |     2 |    26 |     1 (0)| 00:00:01 |
-----

```

PLAN_TABLE_OUTPUT

Predicate Information (identified by operation id):

```

-----
1 - access("M"."NOMM"="P"."NOMM")

```

8. Exercice transaction

Vous répondrez avec brièveté aux questions suivantes :

1. Une séquence d'ordres SQL vous est fourni. Vous définirez le rôle de l'ordre rollback en général et en expliquerez les effets sur la séquence proposée. Quels sont les ordres SQL impactés et pourquoi ?

```
INSERT INTO Fragrance VALUES ('violette','vegetal');
INSERT INTO Fragrance VALUES ('verveine','vegetal');
ALTER TABLE FRAGRANCE ADD prix float;
INSERT INTO Fragrance VALUES ('musc nitre','animal',100);
UPDATE Fragrance SET origine='synthetique' WHERE nomF='musc nitre';
ROLLBACK;
```

2. quelles sont les propriétés caractéristiques d'une session dont le début est marqué par l'ordre SQL suivant :

```
ALTER SESSION SET ISOLATION_LEVEL = SERIALIZABLE;
```

3. A partir du schéma de l'examen, proposez un exemple d'étreinte mortelle (deadlock) pouvant survenir entre deux transactions concurrentes

9. Exercice PL/SQL

Vous donnerez l'écriture d'un déclencheur d'avertissement qui renvoie le message : *odeur trop forte* si un parfum se compose de plus de deux fragrances d'origine animale (déclencheur portant sur l'insertion et la mise à jour de tuples dans la table composeDe).