

## DBA/ORACLE 12C / TP4 le fichier de contrôle

## Rappel

PR.SANAA EL FILALI

Smi / s6

Année universitaire : 2021/2022

### Mode d'ouverture de la base

1-Énoncez les différents modes d'ouverture de la base :

- **READ-WRITE** : Base de données ouverte en lecture-écriture
- **READ-ONLY** : Base de données ouverte en lecture-Seul
- **Mounted** : Base de donnée montée .

### Mode d'ouverture de la base

2- Associez chaque tâche au mode d'ouverture correspond :

- **INSERT INTO regions VALUES (5, 'Mars');**

**READ-WRITE** : car la base est ouverte en mode lecture-écriture :

**READ-ONLY** : pas possible la base est ouverte que pour la lecture .

**MONTÉE** : pas possible car la base est à l'état mount ( on n'as pas accès au fichier de données ) .

## Mode d'ouverture de la base

Associez chaque tâche au mode d'ouverture correspond :

- Select name from v\$database

READ-WRITE : **Lecture** – Ecriture : possible .

READ-ONLY : possible la base est ouverte pour la lecture .

~~MONTÉE~~ → pas possible car la base est à l'état mount ( on n'as pas accès au fichier de données ) .

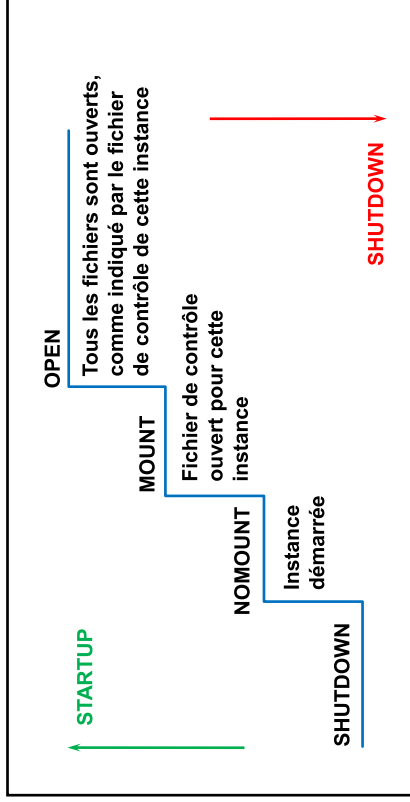
## Mode d'ouverture de la base

Associez chaque tâche au mode d'ouverture correspond :

- Alter database open read –only .

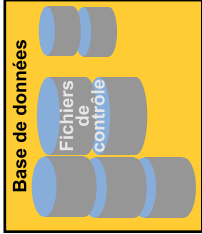
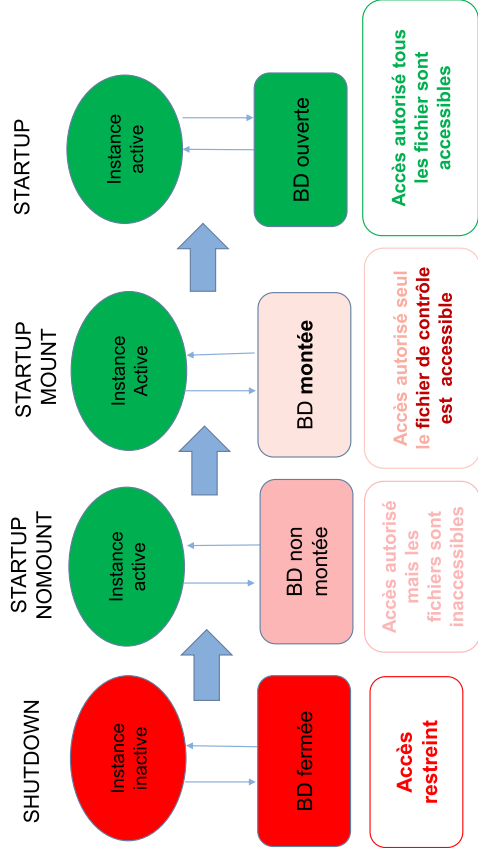
Cette requete va s'exécuter qu'à l'état mounted , car on doit changer le mode d'ouverture en lecture seul avant d'ouvrir la base ( le fichier de données et le fichier journal ) .

## Démarrer une base de données en mode OPEN



## Les différents mode de démarrage

# Démarrer une base de données en mode OPEN



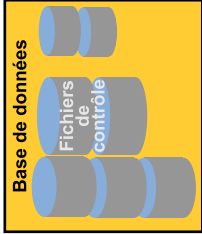
## Fichiers de contrôle -

- Il s'agit d'un petit fichier binaire.
- Il définit l'état actuel de la base de données physique.
- Il assure l'intégrité de la base de données.
- Il est requis :
  - lors de l'étape MOUNT lors du démarrage de la base de données,
  - pour le fonctionnement de la base de données.
- Il est lié à une seule base de données.
- La perte de ce fichier peut nécessiter la récupération de la base de données.

## Introduction au fichier de contrôle

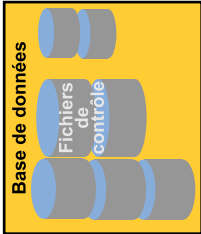
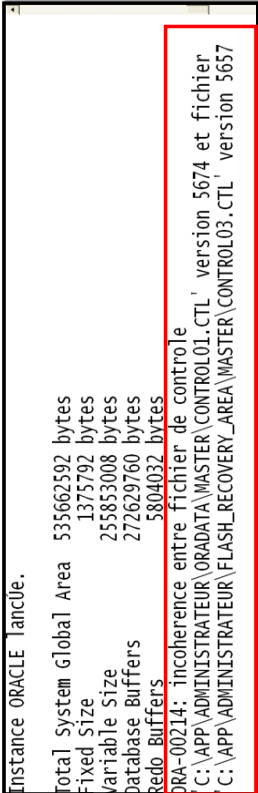
### Contenu du fichier de contrôle

- Le fichier de contrôle contient les entrées suivantes :
  - Nom et identificateur de la base de données
  - Nom des tablespaces
  - Nom et emplacement des fichiers de données et des fichiers de journalisation (redo log)
  - Numéro de séquence du fichier de journalisation en cours
  - Informations sur les points de reprise (checkpoints)
  - Informations sur l'archivage des fichiers de journalisation
  - Informations sur les sauvegardes



## Cas de perte du fichier de contrôle :

Exemple : On supprime le fichier de contrôle et on tente d'ouvrir la base par STARTUP .



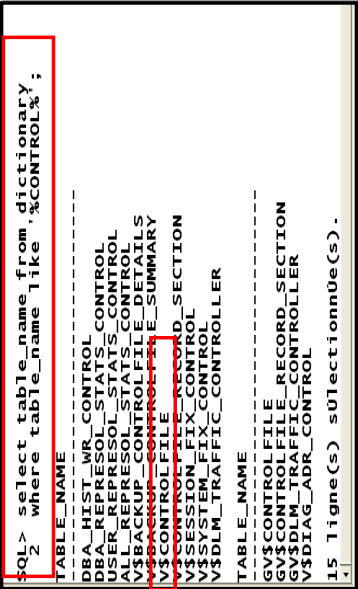
On ne peut pas ouvrir la base sans fichier de contrôle , on est dans l'état NOMOUNT .

Pour la sécurité de la base , On doit avoir plusieurs copie du fichier de contrôle .

## Fichiers de contrôle

1-Énoncez plusieurs méthodes pour identifier le nombre et le nom de chaque fichier de contrôle actif. (méthode 1)

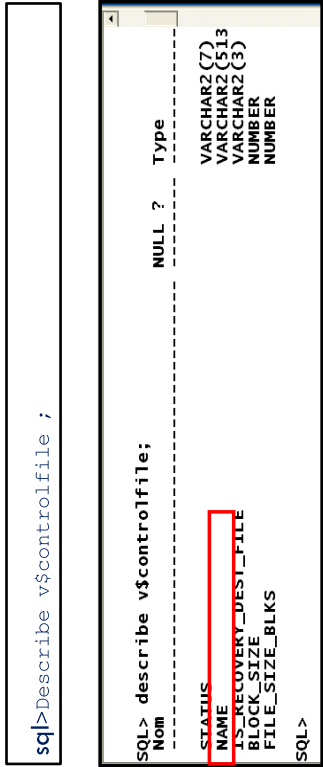
Quelle est la vue à interroger ?



## Fichiers de contrôle - fichiers journaux – fichiers de données

1-Énoncez plusieurs méthodes pour identifier le nombre et le nom de chaque fichier de contrôle actif. (méthode 1)

Avant de sélectionner , il faut décrire les champs de la vue , à travers la commande describe .



## Fichiers de contrôle - fichiers journaux – fichiers de données

1-Énoncez plusieurs méthodes pour identifier le nombre et le nom de chaque fichier de contrôle actif. (méthode 1)

Une fois les champs sont déterminée , on sélectionne le champ name .

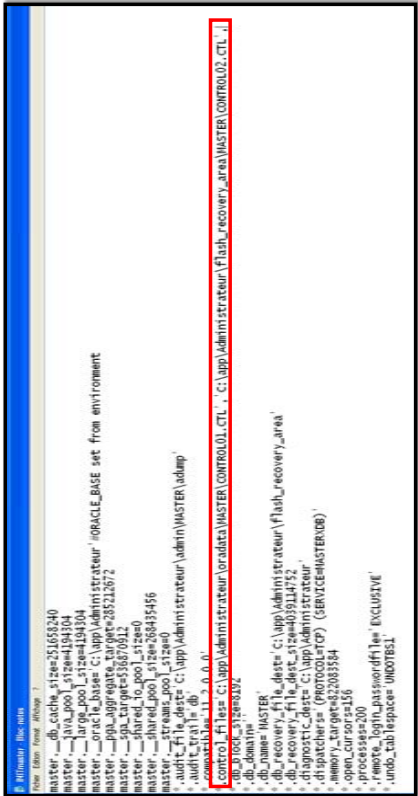
```
sql> Select name from v$controlfile ;
```

```
SQL> select name from v$controlfile ;

NAME
-----
C:\APP\ADMINISTRATEUR\ORADATA\MASTER\CONTROL01.CTL
C:\APP\ADMINISTRATEUR\FLASH_RECOVERY_AREA\MASTER\CONTROL02.CTL
```

## Fichiers de contrôle -

1-Énoncez plusieurs méthodes pour identifier le nombre et le nom de chaque fichier de contrôle actif. (méthode 3)



On peut visualiser à travers le fichier pfile le nombre ainsi que le chemin des contrôle file qu'on a .

## Fichiers de contrôle -

1-Énoncez plusieurs méthodes pour identifier le nombre et le nom de chaque fichier de contrôle actif. (méthode 2)

```
sql>show parameter control_files
```

NAME	TYPE	VALUE
control_files	string	C:\APP\ADMINISTRATEUR\ORADATA\MASTER\CONTROL01.CTL, C:\APP\ADMINISTRATEUR\FLASH_RECOVERY_AREA\MASTER\CONTROL02.CTL

## Multiplexer le fichier de contrôle

## Multiplexer le fichier de contrôle

1- Sachant qu'on possède déjà deux fichier de contrôle , control01.CTL et control02.CTL.



Le but est de créer une troisième copie , pour le faire on suit les étapes suivantes :

- **Création physique** : Copier un des fichier et créer un troisième fichier nommé control03.CTL .
- **Création logique** : Pour que le changement s'applique au niveau du système , on Ajoute le chemin du fichier ajouté ( control03.CTL ) au niveau du fichier de paramètres .

## Multiplexer le fichier de contrôle :

lorsqu'un fichier **PFILE** est utilisé

1. Arrêtez la base de données :

```
shutdown immediate
```

2. Créez un fichier de contrôle supplémentaire :

```
Créer une copie physique du fichier control03.CTL à partir d'un des deux
fichier (control01.CTL / control02.CTL) .
cp $HOME/ORADATA/u01/ctrl01.ctl
$HOME/ORADATA/u02/ctrl03.ctl
```

3. Ajoutez les noms des fichiers de contrôle au fichier PFILE :

```
CONTROL_FILES = (/DISK1/control01.ctl,
                 /DISK3/control02.ctl), /DISK3/control03.ctl) )
```

4. Démarrez la base de données :

```
startup
```

## Multiplexer le fichier de contrôle :

lorsqu'un fichier **SPFILE** est utilisé

1. Modifiez le SPFILE :

```
ALTER SYSTEM SET control_files =
'$HOME/ORADATA/u01/ctrl01.ctl',
'$HOME/ORADATA/u02/ctrl02.ctl',
'$HOME/ORADATA/u02/ctrl03.ctl'
SCOPE=SPFILE;
```

2. Arrêtez la base de données :

```
shutdown immediate
```

3. Créez des fichiers de contrôle supplémentaires :

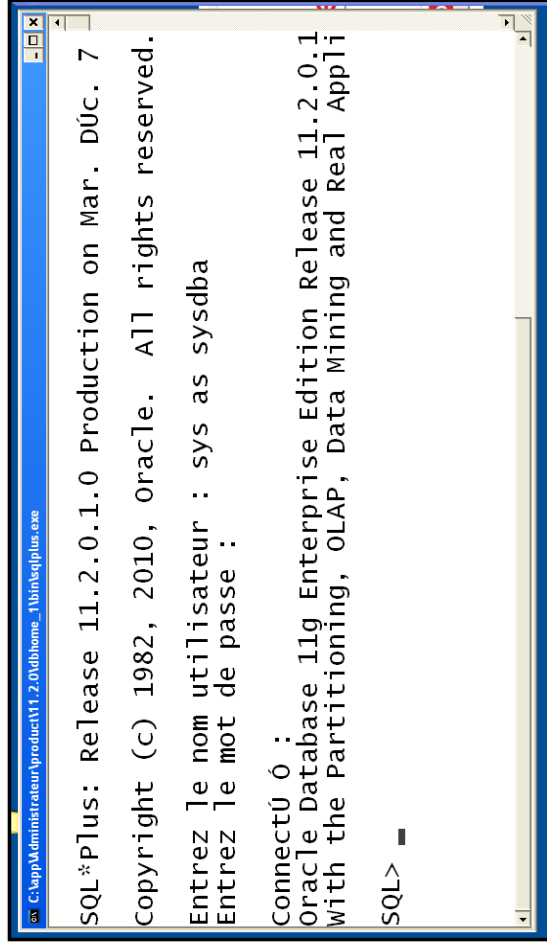
```
cp $HOME/ORADATA/u01/ctrl01.ctl
$HOME/ORADATA/u02/ctrl02.ctl
```

4. Démarrez la base de données :

```
startup
```

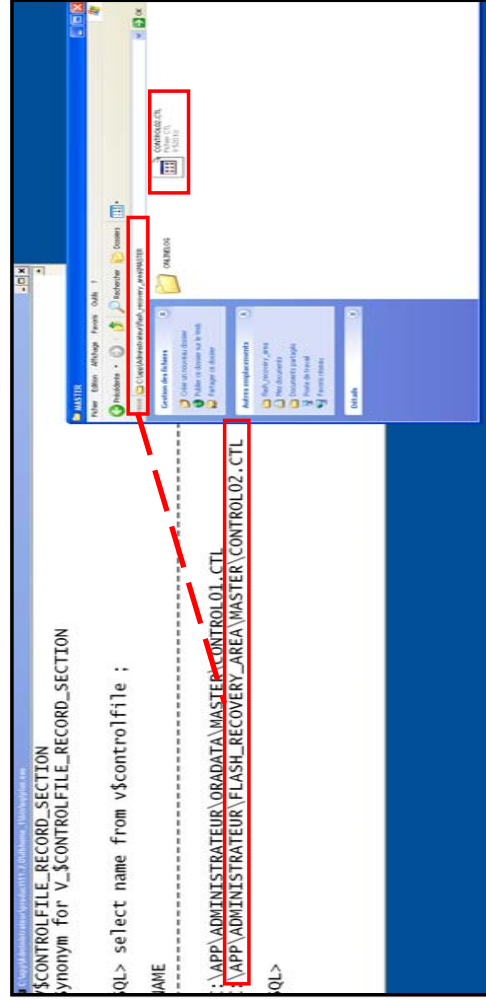
lorsqu'un fichier **SPFILE** est utilisé

1- Connectez vous à la base avec user sys :



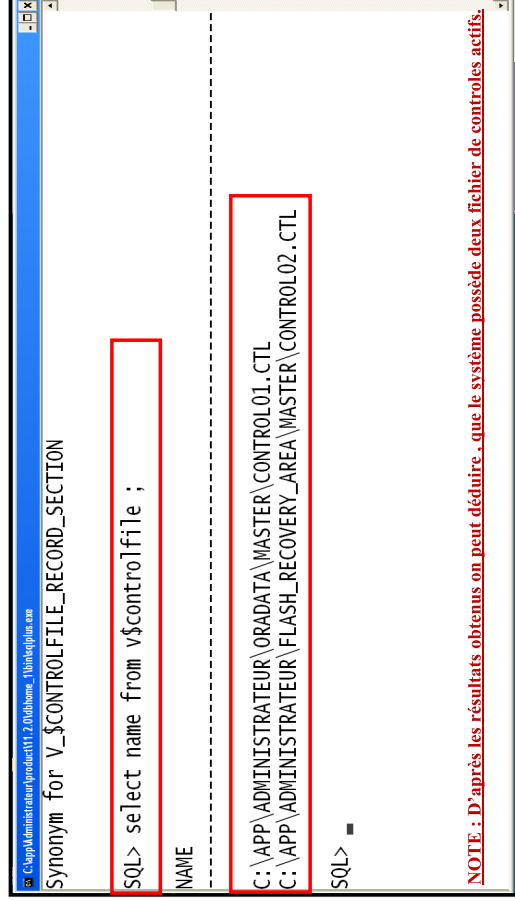
3- créez un nouveau fichier control03.CTL :

On accède à un des fichiers :



1- Afficher les fichiers de contrôles que vous disposez :

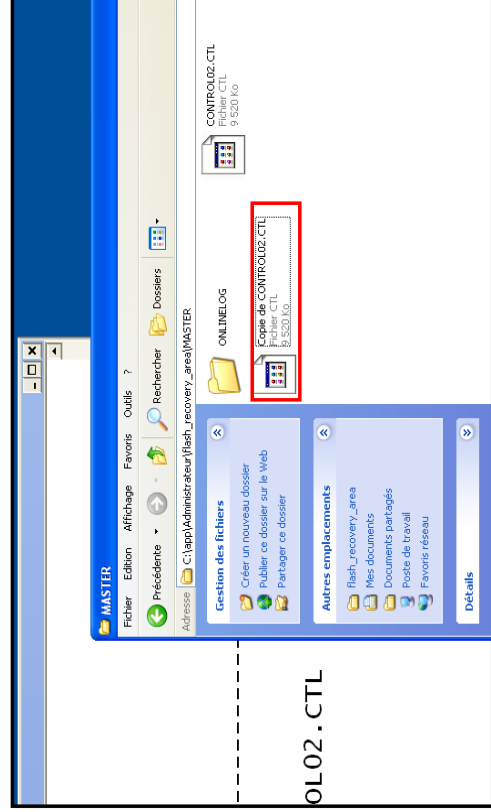
On passe par la première méthode et on interroge la vue V\$CONTROLFILE.



NOTE : D'après les résultats obtenus on peut déduire, que le système possède deux fichiers de contrôles actifs.

3- créez un nouveau fichier control03.CTL :

On crée une copie :

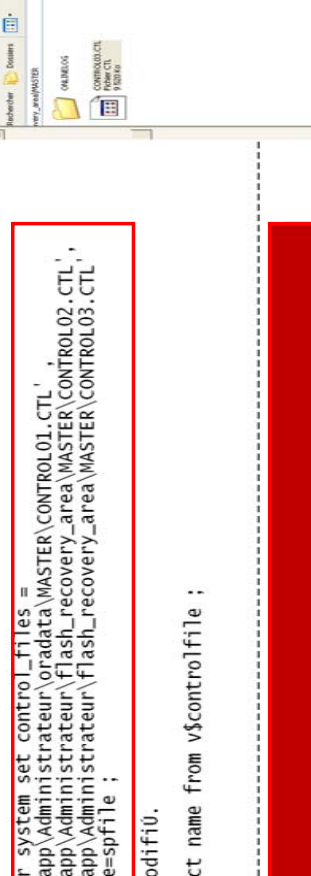






4- Une fois que le nouveau chemin est ajouté.

A votre avis si on teste encore une autre fois combien de fichier le système va afficher :



The screenshot displays a Windows XP desktop environment. In the foreground, a command prompt window is open, showing the following SQL commands and their output:

```
SQL> alter system set control_files =
2 'c:\app\Administrateur\oradata\MASTER\CONTROL01.ctl',
3 'c:\app\Administrateur\flash_recovery_area\MASTER\CONTROL02.ctl',
4 'c:\app\Administrateur\flash_recovery_area\MASTER\CONTROL03.ctl'
5 scope=spfile;

Systeme modifi .

SQL> select name from v$controlfile;
```

The output of the last command is partially obscured by a redacted area, but the column header 'NAME' is visible.

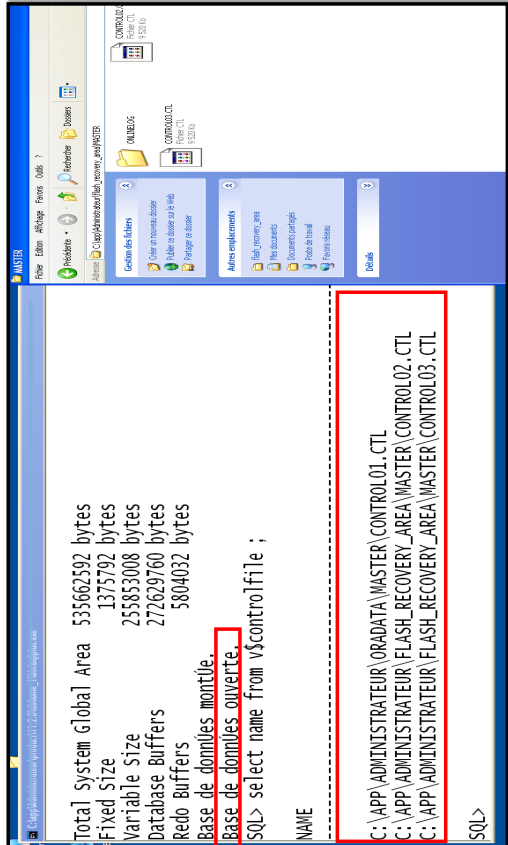
In the background, a Windows Explorer window is open, showing the contents of the file 'C:\app\ADMINISTRATEUR\flash\_recovery\_area\MASTER\CONTROL02.ctl'. The file is a binary file, and its contents are displayed as a series of hexadecimal and ASCII characters.

**ATTENTION!**

Total System Global Area 535662592 bytes  
 Fixed Size 1375792 bytes  
 Variable Size 255853008 bytes  
 Database Buffers 272629760 bytes  
 Redo Buffers 5804032 bytes  
 Base de données ouverte.  
 SQL> select name from v\$controlfile;

Les modifications au niveau de **SPFILE** sont prises en compte après **redémarrage**.

5-redémarrez la base , et testez .



## Fichiers de contrôle - fichiers journaux – fichiers de données

3-Quelle est la taille initiale de la section relative aux fichiers de données dans le fichier de contrôle ?

```
sql>SELECT records_total
      FROM v$controlfile_record_section
     WHERE type = 'DATAFILE';
```

```
SQL> SELECT records_total
2      FROM v$controlfile_record_section
3      WHERE type = 'DATAFILE';

RECORDS_TOTAL
-----
          1024
```