

# Chapitre 7

## Création d'une nouvelle base de données

### 1. Vue d'ensemble

#### 1.1 Étapes de création d'une nouvelle base de données pour une application

Le processus complet de création d'une nouvelle base de données pour une application comporte les grandes étapes suivantes :

##### Conception du modèle physique

- Définir tous les objets (Oracle) de l'application : tables, contraintes d'intégrité (clés primaires/uniques/étrangères), index, vues, programmes stockés (triggers, procédures/ fonctions stockées, packages).
- Étudier la volumétrie de l'application (nombre d'utilisateurs, nombre de lignes attendues dans les tables).

##### Création de la base proprement dite (ce chapitre)

- Créer une nouvelle instance.
- Créer une nouvelle base de données (fichiers de contrôle, fichiers de journalisation et fichiers de données des tablespaces "techniques" d'Oracle).
- Rendre le dictionnaire de données exploitable.
- À ce stade, la base de données peut être vue comme une "enveloppe" (une "boîte vide") dans laquelle des structures vont être créées pour une ou plusieurs applications.

### Création des structures de stockage adaptées (chapitre Tablespaces et des fichiers de données)

- Créer les tablespaces (avec leurs fichiers de données) destinés à stocker les données de l'application (tables et index).
- Les dimensionner en fonction de l'étude de volumétrie réalisée initialement.

### Création du compte Oracle qui va contenir les objets de l'application (chapitre Gestion des utilisateurs et de leurs droits)

- Créer le compte.
- Lui donner les privilèges suffisants pour créer les objets.
- L'autoriser à utiliser de l'espace dans les tablespaces de l'application.

### Création des objets de l'application dans ce compte Oracle (chapitre Gestion des tables et des index)

- Créer les objets Oracle de l'application (généralement sous la forme d'un ou de plusieurs scripts).

### Création des utilisateurs finaux de l'application (chapitre Gestion des utilisateurs et de leurs droits)

- Créer les utilisateurs.
- Leur donner des droits adaptés sur les objets de l'application (i.e. sur les objets créés précédemment dans le compte propriétaire de l'application).

### Sauvegarde de la base (chapitre Sauvegarde et récupération)

- Sauvegarde de référence de la base.

Comme vous pouvez le constater, la création de la base de données proprement dite présentée dans ce chapitre n'est qu'une petite étape du processus complet (mais une étape fondamentale).

## 1.2 Étapes de création de la base de données proprement dite

Les grandes étapes de la création de la base de données proprement dite sont les suivantes :

- Créer les répertoires sur les disques, si possible en respectant les recommandations du standard OFA.
- Préparer un nouveau fichier de paramètres texte, généralement par copie d'un ancien.

- Positionner la variable d'environnement `ORACLE_SID`.
- Créer le service associé à l'instance (plate-forme Windows) ou créer le fichier de mots de passe pour l'identification `SYSDBA` (plate-forme Unix ou Linux).
- Lancer SQL\*Plus et se connecter `AS SYSDBA`.
- Créer un fichier de paramètres serveur (pas obligatoire, mais vivement conseillé).
- Démarrer l'instance en état `NOMOUNT`.
- Créer la base de données (ordre `SQL CREATE DATABASE`).
- Finaliser la création du dictionnaire (quelques scripts à exécuter).
- Configurer Oracle Net pour la nouvelle base de données.
- Enregistrer la nouvelle instance dans le fichier `oratab` (plate-forme Unix ou Linux).
- Configurer EM Express.

La création d'une nouvelle base de données suppose l'installation préalable d'Oracle (chapitre Installation).

### ■ Remarque

*Si le serveur abrite déjà des bases de données Oracle, il est vivement conseillé d'effectuer une sauvegarde de ces bases de données avant de démarrer le processus de création.*

Après ces étapes, la nouvelle base de données est ouverte et contient :

- les tablespaces `SYSTEM` et `SYSAUX` avec leur(s) fichier(s) de données associé(s) ;
- éventuellement un tablespace d'annulation et un tablespace temporaire selon les options utilisées ;
- les fichiers de contrôle et de journalisation ;
- les deux comptes DBA standard (`SYS` et `SYSTEM`) ;
- le segment d'annulation `SYSTEM` ;
- le dictionnaire de données.

À ce stade, la base de données est prête pour accueillir des structures complémentaires qui vont constituer l'application.

## 1.3 Méthodes disponibles

La nouvelle base de données peut être créée à la main avec les outils du système d'exploitation et SQL\*Plus ; dans ce cas, il est très simple d'écrire ou de récupérer des scripts et de les réutiliser à chaque fois. Les étapes de création de la base de données proprement dite sont toujours les mêmes et dépendent (relativement) peu des caractéristiques de l'application (et en tout état de cause, des paramètres peuvent être ajustés ultérieurement en fonction des caractéristiques de l'application) ; utiliser des scripts "génériques" de création de bases est donc envisageable.

La nouvelle base de données peut aussi être créée à l'aide d'un assistant graphique, l'assistant **Configuration de base de données**. Cet assistant facilite la création de la base de données en offrant la possibilité d'utiliser des modèles de base de données prêts à l'emploi et/ou en permettant de définir très précisément les caractéristiques de la nouvelle base de données à l'aide de plusieurs écrans. Par ailleurs, il est possible de définir ses propres modèles de base de données, comprenant ou non des fichiers de données prêts à l'emploi puis de les utiliser lors de la création ultérieure d'une nouvelle base de données. L'assistant graphique offre aussi la possibilité de générer les scripts de création de la base de données sans créer la base de données ; c'est un bon moyen pour constituer nos scripts "génériques". Cet assistant peut aussi être utilisé en mode non interactif, avec ou sans fichier de réponse.

### ■ Remarque

*Utiliser l'assistant graphique est la méthode recommandée par Oracle pour créer une nouvelle base de données ; c'est la seule méthode qui sera présentée dans cet ouvrage.*

## 2. Les principaux paramètres d'initialisation

### 2.1 Préambule

Il y a plus de 400 paramètres documentés par Oracle ! Il n'est évidemment pas question de les spécifier tous ! Sur la totalité des paramètres, entre vingt et trente paramètres qu'il convient de connaître sont suffisants pour la plupart des bases de données.

Certains paramètres seront décrits brièvement dans cette partie puis présentés de manière plus détaillée dans des chapitres ultérieurs.

Lors de la création d'une base de données à l'aide de l'assistant graphique, certains de ces paramètres sont définis en fonction des valeurs saisies dans les différents écrans.

### 2.2 Description

Les paramètres ne sont pas listés dans un ordre alphabétique mais dans un ordre thématique.

#### DB\_NAME

Nom de la base (jusqu'à 8 caractères). Généralement DB\_NAME est égal au nom de l'instance (ORACLE\_SID).

#### Exemple

```
DB_NAME = hermes
```

#### DB\_DOMAIN

Localisation logique de la base sur le réseau (jusqu'à 128 caractères). Ce paramètre, associé au paramètre DB\_NAME, permet à Oracle de construire le nom global de la base de données, sous la forme DB\_NAME.DB\_DOMAIN. Ce paramètre est important si la base de données appartient à un système distribué (ou susceptible de l'être) ; sinon, il peut être ignoré.

#### Exemple

```
DB_DOMAIN = olivier-heurtel.fr
```

#### DB\_UNIQUE\_NAME

Nom unique de base de données (jusqu'à 30 caractères). Des bases de données ayant le même DB\_NAME au sein du même DB\_DOMAIN (par exemple une base de production et une base de test) doivent avoir un DB\_UNIQUE\_NAME différent. Ce paramètre est apparu en version 10. Il est, par défaut, égal à DB\_NAME.

Ce paramètre doit être spécifié si vous souhaitez ouvrir simultanément sur un serveur deux bases portant le même nom (le même DB\_NAME) ; il permet de les différencier.

#### Exemple

```
DB_UNIQUE_NAME = hermes_demo
```

#### COMPATIBLE

Indique un numéro de version d'Oracle avec laquelle la base de données doit être compatible. Valeurs possibles : 11.2.0 jusqu'au numéro de la version actuelle (19.0.0). Valeur par défaut : 19.0.0.

Ce paramètre permet d'utiliser une nouvelle version d'Oracle en restant compatible avec une version plus ancienne, et donc sans avoir besoin de tester les nouvelles fonctionnalités sur la base de données. Certaines fonctionnalités de la nouvelle version peuvent être restreintes. La valeur du paramètre peut être augmentée ultérieurement, mais il est ensuite généralement impossible de redescendre (il faut repartir d'une sauvegarde antérieure au changement).

#### Exemple

```
COMPATIBLE = 19.0.0.
```

### CONTROL\_FILES

Emplacement des fichiers de contrôle de la base de données. Il est conseillé d'en spécifier au minimum 2, si possible sur des disques différents (dans l'idéal, 3 ou 4 sur des disques différents). La recommandation OFA pour le nommage du fichier est `controlN.ctl`, *N* étant un numéro d'ordre (1, 2, etc. ou 01, 02, etc.).

#### Exemple

```
CONTROL_FILES = ("f:\app\oracle\oradata\HERMES\control01.ctl",  
                 "g:\app\oracle\oradata\HERMES\control02.ctl")
```

### DB\_BLOCK\_SIZE

Taille de bloc "standard" en octets, utilisée par défaut pour les fichiers de données des tablespaces et pour l'organisation du cache de données (*buffer cache*). La valeur doit être comprise entre 2 Ko et 32 Ko (sauf restriction spécifique à la plate-forme) et être un multiple de la taille de bloc du système d'exploitation. Ce paramètre ne peut pas être modifié ultérieurement sans recréer la base de données.

Valeur par défaut = 8192 (8 Ko).

La taille de bloc peut avoir un impact plus ou moins important sur les performances. L'exposé des avantages et inconvénients respectifs des "petits" blocs et des "grands" blocs sort du cadre de cet ouvrage. En résumé, les recommandations d'Oracle sont les suivantes :

- Pour un système plutôt transactionnel (généralement caractérisé par des petites requêtes de lecture et de mises à jour), utilisez des "petits" blocs (4 Ko ou 8 Ko).
- Pour un système plutôt décisionnel (généralement caractérisé par des grosses requêtes de lecture), utilisez des "gros" blocs (16 Ko ou 32 Ko).
- Pour les systèmes mixtes, ou dans le doute, utilisez une taille de bloc de 8 Ko (valeur par défaut).

#### Exemple

```
DB_BLOCK_SIZE = 8192
```

### MEMORY\_MAX\_TARGET

Taille maximum de la mémoire utilisable par l'instance. Peut être spécifiée en octets, en Ko (symbole K), en Mo (symbole M) ou en Go (symbole G). Si ce paramètre n'est pas spécifié, il est égal à la valeur du paramètre MEMORY\_TARGET.

N'oubliez pas que les modifications dynamiques de la mémoire s'effectuent dans la limite de la valeur du paramètre MEMORY\_MAX\_TARGET, qui lui n'est pas dynamique (cf. chapitre Les bases de l'architecture Oracle, section L'instance).

#### Exemple

```
MEMORY_MAX_TARGET = 2G
```

### MEMORY\_TARGET

Taille de la mémoire allouée à l'instance. Peut être spécifié en octets, en Ko (symbole K), en Mo (symbole M) ou en Go (symbole G). Valeur par défaut : 0. Valeur minimale : 152 Mo. La valeur peut être arrondie par Oracle au granule supérieur. Ce paramètre est apparu en version 11.

Si ce paramètre a une valeur différente de zéro, la gestion automatique de la mémoire (*Automatic Memory Management - AMM*) est activée. Dans ce cas, Oracle dimensionne automatiquement la SGA et la PGA en fonction de leurs besoins respectifs et de la charge du système (cf. section L'instance du chapitre Les bases de l'architecture Oracle).

#### Exemple

```
MEMORY_TARGET = 2G
```

### SGA\_MAX\_SIZE

Taille maximale de la SGA. Peut être spécifiée en octets, en Ko (symbole K), en Mo (symbole M) ou en Go (symbole G). Si ce paramètre n'est pas spécifié, Oracle lui donne la valeur du paramètre MEMORY\_MAX\_TARGET s'il est défini ou la taille de la SGA au démarrage de l'instance.

N'oubliez pas que les modifications dynamiques de la SGA s'effectuent dans la limite de la valeur du paramètre SGA\_MAX\_SIZE, qui lui n'est pas dynamique (cf. section L'instance du chapitre Les bases de l'architecture Oracle).

#### Exemple

```
SGA_MAX_SIZE = 1G
```