

# 3

## 数据库创建

ORACLE

# 目标

完成这一课的学习后，您应该能达到下列目标：

- 了解创建数据库的前提条件
- 使用 **Oracle Database Configuration Assistant** 创建数据库
- 手动创建数据库
- 使用 “**Oracle 管理文件**” (**Oracle Managed Files**) 创建数据库

# 管理和组织数据库

- 计划数据库是管理数据库系统的第一步
  - 确定数据库的用途
  - 确定数据库的类型
  - 概括数据库的体系结构设计
  - 选择数据库名称
- 创建数据库
- 使用 **Oracle Data Migration Assistant** 可以从较早的数据库版本进行移植

ORACLE

3-3

## 计划和组织数据库

计划数据库是组织和实施数据库系统的第一步。首先要确定数据库的用途，这就需要根据业务要求来确定应该创建哪种数据库类型。数据库类型包括数据仓库、用于高效的联机事务处理或用于通用目的的数据库。确定了用途和类型后，接下来就是概括要应用的数据库体系结构。例如：如何组织和存储数据文件、控制文件和重做日志文件？Oracle 的“最佳灵活体系结构” (Optimal Flexible Architecture) 可以帮助您安排数据库文件的结构和位置。定义了体系结构之后，您必须为新的数据库选择数据库名称和系统标识名。

数据库创建这项任务可用来准备一些操作系统文件，创建只需一次，这与数据库中的数据文件数量无关。

从较早版本的 Oracle 数据库进行移植时，除非需要一个全新的数据库，否则不必创建数据库。如果不必创建数据库，则可以使用移植实用程序。Oracle Data Migration Assistant 就是这样的工具，它可以协助您移植当前的数据库系统。

# 最佳灵活体系结构 (OFA)

- **Oracle 建议使用的标准数据库体系结构布局**
- **OFA 涉及三个主要规则：**
  - 建立一个目录结构，在该目录结构中，任何数据库文件都可以存储在任意磁盘资源上。
  - 将具有不同行为的对象分放到不同的表空间。
  - 通过将数据库组件分别安装到不同的磁盘资源上，使数据库获得最高的可靠性和最佳的性能。

ORACLE

3-4

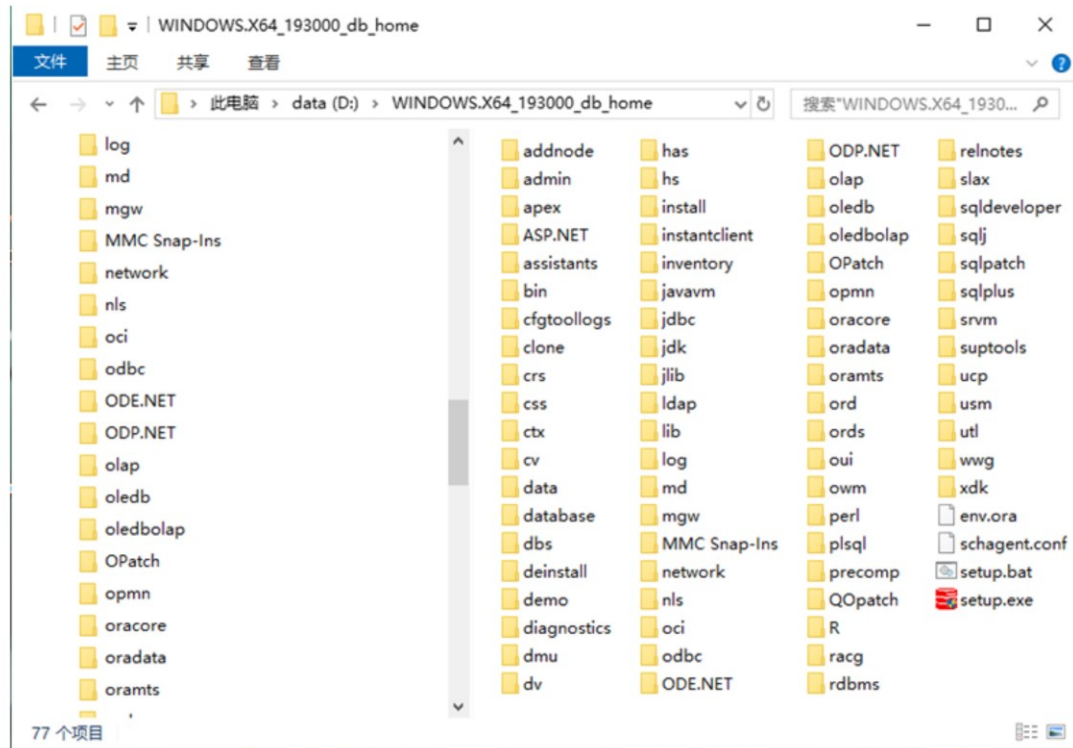
## 最佳灵活体系结构 (OFA)

所有支持的平台上的安装和配置都符合最佳灵活体系结构 (OFA)。OFA 按类型和用途来组织数据库文件。二进制文件、控制文件、日志文件和管理文件可分装在多个磁盘上。

一致的命名约定具有以下优点：

- 可以很容易地将数据库文件与其它文件区别开。
- 易于识别控制文件、重做日志文件和数据文件。
- 通过将文件分装在不同磁盘和目录中，对同一台计算机上的多个 Oracle 主目录的管理变得更加容易。
- 可实现更好的性能，因为数据文件、二进制文件和管理文件现在分别驻留在不同的目录和磁盘上，这样就减少了它们对磁盘的争用。

# Oracle 软件位置



3-5

ORACLE

## Oracle 软件的位置

上面的目录树是符合 OFA 的数据库示例。

### 最佳灵活体系结构：

安装和创建数据库过程中的另一个重要问题就是如何组织文件系统，以便于对数据库的增长进行管理。数据库的增长体现在向现有数据库添加数据、添加用户、创建新数据库、添加硬件、在众多的驱动器之间适当地分配输入输出 (IO) 负载等。

# Oracle 数据库文件位置

名称	修改日期	类型	大小
CONTROL01.CTL	2020-2-10 8:04	CTL 文件	10,352 KB
CONTROL02.CTL	2020-2-10 8:04	CTL 文件	10,352 KB
REDO01.LOG	2020-2-8 18:05	文本文档	204,801 KB
REDO02.LOG	2020-2-9 11:00	文本文档	204,801 KB
REDO03.LOG	2020-2-9 16:38	文本文档	204,801 KB
SYSAUX01.DBF	2020-2-10 8:08	DBF 文件	747,528 KB
SYSTEM01.DBF	2020-2-10 8:08	DBF 文件	911,368 KB
TEMP01.DBF	2020-2-9 16:37	DBF 文件	38,920 KB
UNDOTBS01.DBF	2020-2-10 8:08	DBF 文件	803,848 KB
USERS01.DBF	2020-2-9 16:38	DBF 文件	5,128 KB

ORACLE®

## 创建的前提条件

要新建数据库，您必须具备以下条件：

- 已授权的帐户，通过以下方式之一验证：
  - 操作系统
  - 口令文件
- 有足够的内存可用于启动例程
- 有足够的磁盘空间可用于计划要创建的数据库

ORACLE

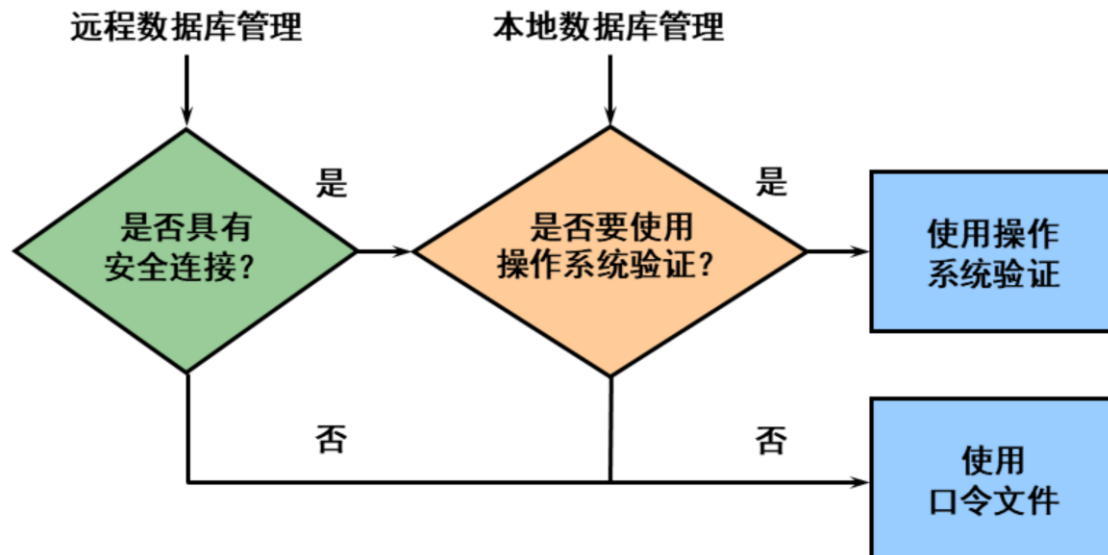
### 创建的前提条件

创建数据库需要具备 SYSDBA 权限。使用操作系统验证或口令文件验证即可授予这些权限。

创建数据库之前，确保有足够的内存可用于 SGA、Oracle 可执行程序 and 进程。请参考操作系统安装和管理指南。

计算数据库所需的磁盘空间，包括联机重做日志文件、控制文件和数据文件。

## 数据库管理员的验证方法



ORACLE

3-8

### 数据库管理员的验证方法

您可能需要从数据库所驻留的计算机上对数据库进行本地管理，或者需要从一个远程客户端管理许多不同的数据库服务器，根据实际情况，请选择使用操作系统还是口令文件来验证数据库管理员。

**注：**有关操作系统验证的信息，请参考针对各操作系统的手册。



## 使用口令文件验证

- 使用口令实用程序创建口令文件

```
D:> orapwd file=%ORACLE_HOME%\dbs\orapwU15  
password=admin entries=5
```

- 在初始化参数文件中设置  
`REMOTE_LOGIN_PASSWORDFILE=EXCLUSIVE`
- 向口令文件添加用户
- 向每个用户分配适当的权限

```
GRANT SYSDBA TO HR;
```

ORACLE

### 如何使用口令文件验证

Oracle 提供了一个口令实用程序 `orapwd` 来创建口令文件。使用 `SYSDBA` 权限进行连接时，您将以 `SYS` 方案而不是与您的用户名关联的方案进行连接。对于 `SYSOPER`，将连接到 `PUBLIC` 方案。

使用口令文件访问数据库的权限通过特权用户发出的特殊 `GRANT` 命令来提供。

**注：**有关授予权限的信息，请参考“管理权限”一课。

## 如何使用口令文件验证（续）

### 使用口令文件：

1. 使用口令实用程序 orapwd 创建口令文件。

```
orapwd file=filename password=password entries=max_users
```

其中：

filename: 口令文件的名称（必需）

password: SYSOPER 和 SYSDBA 的口令（必需）

entries: 允许作为 SYSDBA 或 SYSOPER 连接的不同用户的最大数目。如果超过该值，则必须创建一个新的口令文件。所以设置较大的值较为稳妥。等号 (=) 字符的两侧不应有空格。

示例: orapwd file=%ORACLE\_HOME%\dbs\orapwU15  
password=admin entries=5

其中：

filename: %ORACLE\_HOME%\dbs\orapwU15

password: admin

entries: 5

2. 将 REMOTE\_LOGIN\_PASSWORDFILE 参数设置为 EXCLUSIVE

其中：

EXCLUSIVE 表示只有一个例程可以使用口令文件，并且该口令文件包含 SYS 以外的名称。使用 EXCLUSIVE 口令文件可以向单个用户授予 SYSDBA 或 SYSOPER 权限。

3. 使用上面创建的口令文件连接数据库。

```
CONNECT sys/admin AS SYSDBA
```

### 口令文件位置：

UNIX: %ORACLE\_HOME%\dbs

NT: %ORACLE\_HOME%\database

### 维护口令文件：

使用操作系统命令删除现有口令文件，然后使用口令实用程序创建一个新的口令文件。

# 创建数据库

可通过以下方式创建 Oracle 数据库：

- **Oracle Universal Installer**
- **Oracle Database Configuration Assistant**
  - 图形用户界面
  - 基于 Java
  - 由 Oracle Universal Installer 启动
  - 可独立使用
  - 在 windows 下以管理员身份启动，否则权限不够
- **CREATE DATABASE 命令**

ORACLE

3-11

## 创建数据库

共有三种创建数据库的方式：使用 Oracle Universal Installer 在 Oracle19C 安装中自动创建；使用 Oracle Database Configuration Assistant (DBCA)；或使用 CREATE DATABASE 命令通过创建 SQL 脚本来创建数据库。

Database Configuration Assistant 是一个用来简化数据库创建操作的图形用户界面，它既能与 Oracle Universal Installer 交互使用，也可以独立使用。DBCA 基于 Java，可以从任何带 Java 引擎的平台启动。

安装 Oracle Server 的过程中，Oracle Universal Installer 会启动 DBCA，接着 DBCA 将自动创建一个初始数据库。对于用或不用 DBCA，创建或不创建初始数据库，以及是否在安装后作为独立的应用程序启动 DBCA 来创建数据库，您都有充分的选择自由。

如果使用的是较早版本的 Oracle 软件，还可以对现有数据库进行移植或升级。

# 环境变量

注册表编辑器

文件(F) 编辑(E) 查看(V) 收藏夹(A) 帮助(H)

计算机\HKEY\_LOCAL\_MACHINE\SOFTWARE\ORACLE\KEY\_OraDB19Home1

名称	类型	数据
(默认)	REG_SZ	(数据未设置)
MSHELP_TOOLS	REG_SZ	D:\WINDOWS.X64_193000_db_home\MSHELP
NLS_LANG	REG_SZ	SIMPLIFIED CHINESE_CHINA.ZHS16GBK
OLEDDB	REG_SZ	D:\WINDOWS.X64_193000_db_home\oledb\m...
OLEDDBOLAP	REG_SZ	D:\WINDOWS.X64_193000_db_home\oledbol...
ORA_CDB1_AUTOSTART	REG_EXPAND_SZ	TRUE
ORA_CDB1_SHUTDOWN	REG_EXPAND_SZ	TRUE
ORA_CDB1_SHUTDOWN_TIMEOUT	REG_EXPAND_SZ	90
ORA_CDB1_SHUTDOWNWNTYPE	REG_EXPAND_SZ	immediate
ORA_TEST1_AUTOSTART	REG_EXPAND_SZ	TRUE
ORA_TEST1_SHUTDOWN	REG_EXPAND_SZ	TRUE
ORA_TEST1_SHUTDOWN_TIMEOUT	REG_EXPAND_SZ	90
ORA_TEST1_SHUTDOWNWNTYPE	REG_EXPAND_SZ	immediate
ORACLE_BASE	REG_SZ	D:\app\111973004
ORACLE_BUNDLE_NAME	REG_SZ	Enterprise
ORACLE_GROUP_NAME	REG_SZ	Oracle - OraDB19Home1
ORACLE_HOME	REG_SZ	D:\WINDOWS.X64_193000_db_home
ORACLE_HOME_KEY	REG_SZ	SOFTWARE\ORACLE\KEY_OraDB19Home1
ORACLE_HOME_NAME	REG_SZ	OraDB19Home1
ORACLE_HOME_READONLY	REG_SZ	N
ORACLE_HOME_TYPE	REG_SZ	1
ORACLE_SID	REG_SZ	cdb1
ORACLE_SVCUSER	REG_SZ	NT AUTHORITY\SYSTEM
ORACLE_SVCUSER_PWDREQ	REG_SZ	0
ORACLE_SVCUSER_TYPE	REG_SZ	
RDBMS_ARCHIVE	REG_SZ	D:\WINDOWS.X64_193000_db_home\DATA...
RDBMS_CONTROL	REG_SZ	D:\WINDOWS.X64_193000_db_home\DATA...
SQLPATH	REG_SZ	D:\WINDOWS.X64_193000_db_home\dfs

# Database Configuration Assistant

Database Configuration Assistant 可用于：

- 创建数据库
- 配置现有数据库
- 删除数据库
- 管理可插入数据库
- 管理模板
  - 使用预定义模板设置创建新模板
  - 从现有数据库创建新模板
  - 删除数据库模板
- 管理可插入数据库

ORACLE

3-13

## Database Configuration Assistant

您可以利用一些预定义的模板，也可以将现有数据库用作副本来创建新的数据库或模板。数据库参数以 XML 格式存储。

使用模板的好处：

- 节省创建数据库的时间
- 可共享模板
- 可根据需要变换数据库选项

有关模板的详细信息，请参考 Oracle Database Configuration Assistant 联机帮助。

# 使用 Database Configuration Assistant 创建数据库

- 选择要从预定义模板创建的数据库类型
- 指定全局数据库名称和 SID
- 选择要在数据库中使用到的功能
- 确定数据库创建后要运行的任何脚本
- 选择数据库的运行模式

ORACLE

3-14

## 使用 Database Configuration Assistant 创建数据库

- 启动 Database Configuration Assistant: 程序 (Programs) > Oracle-OraDb19Home1 > 配置和移植工具 (Configuration and Migration Tools) > Database Configuration Assistant。
- 选择“创建数据库”(Create a Database) 选项。
- 从预定义模板列表中选择要创建的数据库类型。
  - 数据仓库
  - 通用
  - 新数据库
  - 事务处理

使用“显示详细资料”(Show Details) 选项查看要创建的数据库。创建的模板可带有数据文件，也可不带有数据文件。

- 不带有数据文件：只包含数据库的结构。可以指定和更改所有数据库参数。
- 带有数据文件：既包含数据库的结构也包含其中的物理数据文件。自动为数据库创建所有日志文件和控制文件，并且可添加\删除控制文件、日志组，还可以更改数据文件的目标位置和名称。无法添加或删除数据文件、表空间或回退段。无法更改初始化参数。

## 创建数据库（续）

- 指定“全局数据库名称”（Global Database Name）和 SID。
- 指定要在数据库中使用到的功能，如：
  - Oracle Spatial
  - Oracle OLAP services
  - 示例模式 (Example Schemas)

示例模式 (Example Schemas) 包含以下类型的表的脚本：

人力资源

订单输入

产品介质

销售历史记录

装运

- 确定数据库创建后要运行的任何脚本。
- 选择数据库的运行模式
  - 专用服务器模式
  - 共享服务器模式

## 使用 Database Configuration Assistant 创建数据库

- 指定内存、归档、数据库大小和文件位置等选项
- 定义数据库存储参数
- 按需更改文件位置变量
- 选择一个数据库创建选项完成数据库的创建

ORACLE

3-16

### 使用 Database Configuration Assistant 创建数据库

- 指定以下选项

- 内存

选择“典型”(Typical)或“自定义”(Custom)数据库

选择“典型”选项创建的数据库涉及的用户输入最少。选择此选项后，可以指定以下任一运行数据库的环境：联机事务处理 (Online Transaction Processing, OLTP)、多用途和数据仓库。

“自定义”选项允许您自定义数据库的创建。该选项仅供具有高级数据库创建经验的数据库管理员使用。

- 归档

此选项将数据库置于 ARCHIVELOG 模式，并在重新使用重做日志文件前对其进行归档。

- DB 大小

此参数用于定义数据库的块大小和排序区大小。只能在创建数据库时指定数据块的大小。SORT\_AREA\_SIZE 指可用于排序操作的最大内存空间。



## 创建数据库（续）

### - 文件位置

用于指定跟踪文件的位置，以及指定初始化参数文件的路径。

- 定义数据库存储参数。此页显示一个树列表以及一个摘要视图（多栏列表），允许您更改和查看以下对象：控制文件、表空间、数据文件、还原段和重做日志组。
- 单击“文件位置变量”（File Location Variables）按钮可更改任意文件位置变量。
- 选择一个数据库创建选项完成数据库的创建。
  - 创建数据库（Create Database）：此选项可用于立即创建数据库。
  - 另存为数据库模板（Save as a Database Template）：此选项可用于将数据库创建参数保存为模板。此模板随即将添加到可用模板列表中。
  - 生成数据库创建脚本（Generate Database Creation Scripts）：此选项可用于将数据库创建参数保存为脚本文件，供以后使用。
- 选择“完成”（Finish）。

# 手动创建数据库

- 为例程和数据库选择唯一的名称。
- 选择一个数据库字符集。
- 设置操作系统变量。
- 创建初始化参数文件。
- 创建服务（windows平台）
- 在 NOMOUNT 阶段启动该例程。
- 创建并执行 CREATE DATABASE 命令。
- 打开数据库。
- 运行脚本以生成数据字典并完成创建后的步骤。
- 根据需要创建其它表空间。

ORACLE

3-18

## 手动创建数据库

- 为例程和数据库选择唯一的名称。
- 选择一个数据库字符集。  
必须定义一个数据库字符集。同时还可选择定义一个国家字符集。例如：
  - 字符集 Character set AL32UTF16
  - 国家字符集 AL16UTF16 (National character set AL16UTF16)有关可供使用的各种字符集的信息，请参考“使用全球化支持”一课。
- 设置操作系统变量。  
需要设置四个环境变量：ORACLE\_HOME、ORACLE\_SID、PATH、LD\_LIBRARY\_PATH。
  - ORACLE\_HOME：安装 Oracle 19c 服务器的顶级目录。
  - ORACLE\_SID：可由用户定义的、分配给数据库例程的名称。用于区分运行在同一台机器上的不同数据库例程
  - PATH：定义操作系统查找可执行程序时要搜索的目录。
  - LD\_LIBRARY\_PATH：定义所需的库文件的存储目录。

## 手动创建数据库（续）

- 创建初始化参数文件。

初始化参数文件是通过随安装过程安装的 `init.ora` 示例文件而创建的。复制 `init.ora` 示例文件，将其命名为 `initSID.ora`。针对要创建的数据库的具体需要来修改该文件。如果要使用 `SPFILE`，则必须首先创建 `PFILE`。有关如何创建数据库特定的 `initSID.ora` 文件和 `SPFILE` 的指导说明，请参考“管理 Oracle 例程”一课。

- 创建服务（windows平台）

在windows平台，需要执行`oradim`命令来创建服务。一般用法为：

```
oradim -new -sid xxx -syspwd xxxxxx -pfile ...
```

- 在 NOMOUNT 阶段启动该例程。

以具有 `SYSDBA` 权限的用户 `SYS` 身份连接。要创建数据库，数据库必须处于 `NOMOUNT` 状态。有关如何让数据库处于 `NOMOUNT` 状态的指导说明，请参考“管理 Oracle 例程”一课。

- 创建并执行 `CREATE DATABASE` 命令。

- 创建包含 `CREATE DATABASE` 命令的 `SQL` 脚本。以具有 `SYSDBA` 权限的用户 `SYS` 身份连接到 `SQL*Plus`。当数据库处于 `NOMOUNT` 状态时，执行该脚本。
- 如果要创建的数据库是通过“Oracle 管理文件” (Oracle Managed Files, OMF) 来管理操作系统文件的，那么 `CREATE DATABASE` 命令的简化程度将非常明显。有关 OMF 的信息，请参考“管理 Oracle 例程”一课。

- 打开数据库。

必须首先打开数据库，然后再运行脚本，创建数据字典并完成创建后的步骤。有关如何从 `NOMOUNT` 状态下打开数据库的指导说明，请参考“管理 Oracle 例程”一课。

- 运行脚本。

- 创建数据库后必须运行两个脚本：`catalog.sql` 和 `catproc.sql`。这两个脚本都必须以具有 `SYSDBA` 权限的用户 `SYS` 身份运行。执行脚本前，数据库必须处于 `OPEN` 状态。
- `catalog.sql`：在基表和动态性能视图上创建视图及其同义词。它还启动其它脚本，为以下各项创建对象：
  - `PL/SQL` 基本环境，包括 `PL/SQL` 数据类型的声明、预定义异常、内置过程和函数、`SQL` 操作等
  - 审计
  - 导入/导出
  - `SQL*Loader`
  - 已安装选项

## 手动创建数据库（续）

- 运行脚本（续）
  - catproc.sql: 创建使用 PL/SQL 所需的程序包和过程。此外，此脚本还创建用于扩展 RDBMS 功能的若干 PL/SQL 程序包，以及用于预警、管道、logminer、大对象、对象、排队、复制和其它内置选项的程序包视图。
  - pupbld.sql: 创建名为“产品用户配置文件” (Product User Profile) 的表以及相关的过程。运行此脚本将在用户每次连接到 SQL\*Plus 时防止生成警告消息。

**注：**必须以用户 SYSTEM 的身份运行此脚本。

- 创建其它表空间。
  - 应该根据数据库的需要创建其它表空间。

**注：***附录 A* 提供了在 UNIX 环境中手动创建数据库的分步说明。此外，有关在特定的平台上创建数据库的信息，还可参考针对不同操作系统的 Oracle 文档。

# 创建数据库

```
CREATE DATABASE db01
LOGFILE
  GROUP 1 ('e:\db01\redo01.log') SIZE 50M,
  GROUP 2 ('e:\db01\redo02.log') SIZE 50M
MAXLOGFILES 20
MAXLOGMEMBERS 5
MAXDATAFILES 100
MAXINSTANCES 1
DATAFILE 'e:\db01\system01.dbf' SIZE 500M autoextend on
SYSAUX DATAFILE 'e:\db01\sysaux01.dbf' SIZE 200M autoextend on
UNDO TABLESPACE undotbs
DATAFILE 'e:\db01\undo01.dbf' SIZE 50M autoextend on
DEFAULT TEMPORARY TABLESPACE temp
TEMPFILE 'e:\db01\temp01.dbf' SIZE 50M autoextend on
CHARACTER SET ZHS16GBK
NATIONAL CHARACTER SET AL16UTF16;
```

ORACLE

3-21

## 创建数据库

若要创建数据库，请使用下列 SQL 命令：

```
CREATE DATABASE [database]
[CONTROLFILE REUSE]
[LOGFILE [GROUP integer] filespec]
[MAXLOGFILES integer]
[MAXLOGMEMBERS integer]
[MAXLOGHISTORY integer]
[MAXDATAFILES integer]
[MAXINSTANCES integer]
[ARCHIVELOG|NOARCHIVELOG]
[CHARACTER SET charset]
[NATIONAL CHARACTER SET charset]
[DATAFILE filespec [autoextend_clause]]
```



## 创建数据库（续）

```
filespec ::= 'filename' [SIZE integer][K|M] [REUSE]
        autoextend_clause ::=
        [AUTOEXTEND {OFF|ON [NEXT integer[K|M]]
                                [MAXSIZE {UNLIMITED|integer[K|M]} }}]

[DEFAULT TEMPORARY TABLESPACE tablespace filespec
 [temp_tablespace_extent_clause]
temp_tablespace_extent_clause::=
EXTENT MANAGEMENT LOCAL UNIFORM [SIZE integer][K|M] ]

[UNDO TABLESPACE tablespace DATAFILE filespec
 [autoextend_clause] ]

[SET TIME_ZONE [time_zone_region]]
}
]
```

其中：

- DATABASE：是要创建的数据库的名称（如果省略了数据库的名称，则使用初始化参数 DB\_NAME 的值）。
- CONTROLFILE REUSE：指定应重新使用参数文件中确定的现有控制文件
- LOGFILE GROUP：指定要使用的日志文件的名称及其所属的组
- MAXLOGFILES：指定可以为数据库创建的重做日志文件组的最大数量
- MAXLOGMEMBERS：指定日志文件组的日志文件成员的最大数量
- MAXLOGHISTORY：指定 Oracle Real Application Clusters 的自动介质恢复能够恢复的归档重做日志的最大数量

## 创建数据库（续）

- DATAFILE: filespec 指定要使用的数据文件
- AUTOEXTEND: 启用或禁用数据文件的自动扩展
- MAXDATAFILES: 指定在执行 CREATE DATABASE 或 CREATE CONTROLFILE 时控制文件的数据文件段的初始大小调整。如果尝试添加新文件（其数量大于 MAXDATAFILES，但小于或等于 DB\_FILES），将引起控制文件自动扩展，以便数据文件段能够容纳更多的文件。
- MAXINSTANCES: 是能够同时装载和打开数据库的例程的最大数量
- ARCHIVELOG: 指定重新使用重做日志前必须归档
- NOARCHIVELOG: 指定不必对重做日志的内容进行归档即可重新使用这些日志
- CHARACTER SET: 是数据库用来存储数据的字符集
- NATIONAL CHARACTER SET: 指定在定义为 NCHAR、NCLOB 或 NVARCHAR2 的列中存储数据时所使用的国家字符集。如果未指定，则国家字符集与数据库字符集相同
- DEFAULT TEMPORARY TABLESPACE: 为数据库创建缺省临时表空间。Oracle 会将所有未指定其它临时表空间的用户都分配到此临时表空间
- UNDO TABLESPACE: 创建还原表空间，并创建指定的数据文件，作为还原表空间的一部分
- SYSAUX :辅助表空间，减轻system表空间的压力。
- SET TIME\_ZONE: 设置数据库所在的时区



## 使用 “Oracle 管理文件” (OMF) 创建数据库

- 使用 **OMF** 可简化操作系统上的文件管理
- **OMF** 由 Oracle 服务器通过 **SQL** 命令创建和删除
- **OMF** 是通过设置以下两个参数来建立的：
  - **DB\_CREATE\_FILE\_DEST**: 设置该参数以提供数据文件的缺省位置
  - **DB\_CREATE\_ONLINE\_LOG\_DEST\_N**: 设置该参数以提供联机重做文件和控制文件的缺省位置
    - 最多可设置五个位置

ORACLE

3-24

### 使用 “Oracle 管理文件” (OMF) 创建数据库

OMF 免除了对 Oracle 数据库中的文件进行直接管理的必要，从而简化了文件管理。

OMF 按如下方式命名：

- 控制文件：ora\_%u.ctl
- 重做日志文件：ora\_%g\_%u.log
- 数据文件：ora\_%t\_%u.dbf
- 临时数据文件：ora\_%t\_%u.tmp

其中的一些字符定义如下：

- %u 是一个八个字符长的字符串，可以确保唯一性。
- %t 是表空间名，如有必要，可按照文件名的最大长度要求将其截断。将表空间名放在唯一性字符串之前，意味着表空间内的所有数据文件按照字母顺序排列显示。
- %g 是重做日志文件组号。
- 带 .dbf 扩展名的 ora\_ 表明该文件是 OMF。

## 使用 “Oracle 管理文件” (OMF) 创建数据库（续）

还原文件没有特殊的扩展名。

不必同时设置参数 DB\_CREATE\_FILE\_DEST 和 DB\_CREATE\_ONLINE\_LOG\_DEST\_N，可以使用其中任意一个，也可以同时使用。

## 使用 “Oracle 管理文件” (OMF) 创建数据库

- 在初始化参数文件中定义 **OMF** 参数。示例：
  - DB\_CREATE\_FILE\_DEST=\\$HOME\ORADATA\u05
  - DB\_CREATE\_ONLINE\_LOG\_DEST\_1=\\$HOME\ORADATA\u01
  - DB\_CREATE\_ONLINE\_LOG\_DEST\_2=\\$HOME\ORADATA\u02
- **CREATE DATABASE** 命令得到了简化：

```
@cddba01.sql  
> CREATE DATABASE dba01;
```

ORACLE

3-26

### 使用 “Oracle 管理文件” (OMF) 创建数据库

要使用 OMF 创建数据库，需要在初始化参数文件中定义参数 DB\_CREATE\_FILE\_DEST 和 DB\_CREATE\_ONLINE\_LOG\_DEST\_n。一旦定义了 OMF 参数，您就不必再定义文件名或位置，创建数据库的语法因而得到了简化。

上面的示例中创建的数据库使用了如下 OMF：

- 位于目录 \\$HOME\ORADATA\u05 中的 SYSTEM 表空间数据文件，大小为 100 MB，可无限地自动扩展。
- 两个联机重做日志组，组中各有两个大小为 100 MB 的成员，分别位于 \\$HOME\ORADATA\u01 和 \\$HOME\ORADATA\u02 中
- 如果启用了自动还原管理模式，目录 \\$HOME\ORADATA\u05 中还会有一个还原表空间数据文件，大小为 10 MB，可无限地自动扩展。同时创建一个名为 SYS\_UNDOTBS 的还原表空间。
- 如果未指定 CONTROL\_FILES 初始化参数，则会生成两个控制文件，分别位于 \\$HOME\ORADATA\u01 和 \\$HOME\ORADATA\u02 中。位于 \\$HOME\ORADATA\u01 中的控制文件是主控制文件。

## 使用 “Oracle 管理文件” (OMF) 创建数据库（续）

如果 CREATE DATABASE 命令失败，创建的所有 OMF 都将被删除。当用户从 DBA\_DATAFILES、V\$DATAFILE、V\$CONTROLFILE 和 V\$LOGFILE 中进行选择时，可看到内部生成的文件名。

可通过 ALTER SYSTEM SET 命令动态修改 DB\_CREATE\_FILE\_DEST 和 DB\_CREATE\_ONLINE\_LOG\_DEST\_N。

## 故障排除

如果存在以下情况，数据库创建将无法成功：

- **SQL** 脚本中存在语法错误
- 要创建的文件已经存在
- 出现操作系统错误，如文件或目录权限错误，或空间不足错误

ORACLE

3-28

### 故障排除

如果幻灯片中显示的三个问题中的任何一个问题发生，`CREATE DATABASE` 语句将以失败告终。应该删除 `CREATE DATABASE` 语句创建的所有文件，改正错误，然后尝试重新创建。

## 数据库的创建结果

数据库应包含：

- 数据文件、控制文件和重做日志文件
- 用户 **SYS**
- 用户 **SYSTEM**，手动创建初始口令为 **manager**
- 内部表（但没有数据字典视图）

ORACLE

3-29

### 数据库的创建结果

创建数据库后，该例程将运行起来，数据库也处于打开状态，可正常操作。数据库中包含用户 **SYS** 和 **SYSTEM**。根据数据库的创建方法，即使用 **DBCA** 创建或手动创建，可以相应地创建其他用户。创建数据库后应立即更改 **SYS** 和 **SYSTEM** 的口令。

可以查看动态性能视图，如 **V\$LOGFILE**、**V\$CONTROLFILE** 和 **V\$DATAFILE**，但没有创建数据字典视图。

## 小结

在这一课中，您应该能够掌握：

- 创建数据库的前提条件
- 使用 **Oracle Database Configuration Assistant** 创建数据库
- 手动创建数据库
- 使用 “**Oracle 管理文件**” (**Oracle Managed Files**) 创建数据库