

21世纪高等教育计算机规划教材

COMPUTER

# Oracle 11g 数据库 管理与开发基础教程

Database Management  
and Development of Oracle 11g

袁鹏飞 主编

袁鹏飞 杨艳华 编著

结合作者多年的数据库管理、开发和教学经验

内容精炼，循序渐进，深入浅出

提供全面的 Oracle 入门指导



人民邮电出版社  
POSTS & TELECOM PRESS

21世纪高等教育计算机规划教材

COMPUTER

# Oracle 11g 数据库 管理与开发基础教程

Database Management  
and Development of Oracle 11g

■ 袁鹏飞 主编

■ 袁鹏飞 杨艳华 编著



人民邮电出版社

北京

## 图书在版编目 (C I P) 数据

Oracle 11g数据库管理与开发基础教程 / 袁鹏飞主编 ; 杨艳华编著. — 北京 : 人民邮电出版社, 2013. 2  
21世纪高等教育计算机规划教材  
ISBN 978-7-115-30403-2

I. ①0… II. ①袁… ②杨… III. ①关系数据库系统—高等学校—教材 IV. ①TP311.138

中国版本图书馆CIP数据核字(2012)第304081号

## 内 容 提 要

Oracle DataBase 是目前最为流行的 RDBMS 产品之一, 拥有众多的高端用户。它已成为大型数据库应用的首选平台, 自然也成为大学“大型数据库技术”课程的首选内容。

本书较为全面地介绍 Oracle DataBase 11g 的基本管理操作和应用开发方法。全书共分 18 章, 介绍 Oracle 数据库服务器环境的建立与日常管理操作、常见对象管理、游标和动态 SQL 技术, 以及 Oracle DataBase 对面向对象技术的支持。

本书可作为大学本科有关课程的教材, 也可供广大 Oracle 数据库管理员和数据库应用程序开发人员参考。

21 世纪高等教育计算机规划教材

### Oracle 11g 数据库管理与开发基础教程

- 
- ◆ 主 编 袁鹏飞  
编 著 袁鹏飞 杨艳华  
责任编辑 邹文波
  - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号  
邮编 100061 电子邮件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
北京昌平百善印刷厂印刷
  - ◆ 开本: 787×1092 1/16  
印张: 19.75 2013 年 2 月第 1 版  
字数: 532 千字 2013 年 2 月北京第 1 次印刷

---

ISBN 978-7-115-30403-2

定价: 39.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223  
反盗版热线: (010)67171154

# 目 录

## 第一部分 Oracle 服务器管理

第 1 章 建立 Oracle 数据库环境 .....	3
1.1 Oracle Database 11g 数据库产品 .....	3
1.2 Oracle 数据库体系结构 .....	3
1.2.1 Oracle 数据库物理存储结构 .....	4
1.2.2 Oracle 数据库逻辑存储结构 .....	5
1.2.3 Oracle 数据库实例 .....	7
1.2.4 连接模式与服务器进程 .....	11
1.3 Oracle 数据库服务器软件安装 .....	13
1.4 数据库创建 .....	16
1.4.1 用 DBCA 创建数据库 .....	16
1.4.2 手工创建数据库 .....	19
1.4.3 打开数据库 .....	23
1.4.4 关闭数据库 .....	24
1.5 Oracle Net 配置 .....	24
1.5.1 服务器端监听配置 .....	25
1.5.2 客户端配置与数据库连接测试 .....	29
本章小结 .....	32
习题 .....	33
第 2 章 常用 Oracle 管理工具 .....	34
2.1 SQL*Plus .....	34
2.1.1 SQL*Plus 的启动和关闭 .....	34
2.1.2 SQL*Plus 变量与参数配置 .....	35
2.1.3 编辑执行命令 .....	37
2.2 SQL Developer .....	39
2.2.1 建立数据库连接 .....	40
2.2.2 管理数据库对象 .....	41
2.2.3 使用 SQL 工作表 .....	42
2.3 Oracle 企业管理器 .....	44
2.3.1 配置 Database Control .....	44
2.3.2 启动和停止 Database Control .....	45
2.3.3 Database Control 功能概述 .....	47
本章小结 .....	47
习题 .....	47
第 3 章 静态数据字典与动态性能视图 .....	49
3.1 静态数据字典 .....	49
3.1.1 3 组常用数据字典视图 .....	49
3.1.2 其他特殊数据字典视图 .....	52
3.2 动态性能视图 .....	53
3.2.1 动态性能视图的创建和填充 .....	53
3.2.2 常用动态性能视图 .....	54
本章小结 .....	56
习题 .....	57
第 4 章 初始化参数文件与控制文件 .....	58
4.1 初始化参数文件 .....	58
4.1.1 初始化参数 .....	58
4.1.2 初始化参数文件 .....	59
4.1.3 设置初始化参数 .....	60
4.1.4 查看初始化参数 .....	62
4.1.5 用 OEM 管理初始化参数 .....	62
4.2 控制文件 .....	63
4.2.1 控制文件结构 .....	64
4.2.2 查看控制文件 .....	65
4.2.3 控制文件的多路存储 .....	65
4.2.4 控制文件的备份、恢复与重新创建 .....	65
4.2.5 用 OEM 管理控制文件 .....	68
本章小结 .....	69
习题 .....	69
第 5 章 重做日志管理 .....	70
5.1 重做日志的基本概念 .....	70
5.1.1 重做日志的内容 .....	70
5.1.2 重做日志的写入方式 .....	70

5.2 管理联机重做日志文件组及成员 .....	74	6.4 数据文件管理 .....	103
5.2.1 查看重做日志文件信息 .....	74	6.4.1 为表空间添加数据文件 .....	104
5.2.2 管理重做日志文件 .....	76	6.4.2 调整数据文件的大小 .....	105
5.3 管理归档重做日志 .....	79	6.4.3 改变数据文件的可用性 .....	107
5.3.1 设置归档位置 .....	79	6.4.4 重命名和移动数据文件 .....	108
5.3.2 设置归档日志文件命名格式 .....	81	6.4.5 删除数据文件 .....	109
5.3.3 调整归档进程数量 .....	81	本章小结 .....	110
5.3.4 改变归档模式 .....	81	习题 .....	110
5.3.5 查新归档重做日志相关的信息 .....	83		
本章小结 .....	85	<b>第 7 章 安全管理</b> .....	111
习题 .....	85	7.1 用户管理 .....	111
<b>第 6 章 表空间与数据文件</b> .....	86	7.1.1 用户身份验证 .....	112
6.1 管理永久表空间 .....	86	7.1.2 用 OEM 管理用户 .....	116
6.1.1 表空间的分类 .....	86	7.1.3 删除用户 .....	117
6.1.2 创建表空间 .....	86	7.2 概要文件 .....	118
6.1.3 区分配管理 .....	88	7.2.1 用概要文件管理资源 .....	118
6.1.4 段空间管理 .....	89	7.2.2 用概要文件控制口令设置 .....	119
6.1.5 数据块大小与数据缓冲区设置 .....	90	7.2.3 使用概要文件 .....	120
6.1.6 改变表空间的可用性 .....	90	7.2.4 用 OEM 管理概要文件 .....	123
6.1.7 设置表空间的读写属性 .....	92	7.3 权限管理 .....	124
6.1.8 重命名和删除表空间 .....	92	7.3.1 系统权限管理 .....	124
6.1.9 设置数据库默认表空间 .....	93	7.3.2 对象权限管理 .....	128
6.1.10 查询表空间相关的信息 .....	94	7.4 角色管理 .....	131
6.2 管理临时表空间 .....	96	7.4.1 创建角色 .....	132
6.2.1 创建临时表空间 .....	96	7.4.2 为角色授权 .....	133
6.2.2 设置默认临时表空间 .....	96	7.4.3 管理用户角色 .....	133
6.2.3 临时表空间内的空间分配 .....	97	7.4.4 查询角色信息 .....	136
6.3 管理还原数据 .....	97	7.4.5 修改和删除角色 .....	137
6.3.1 Undo 的作用 .....	98	7.4.6 预定义角色 .....	138
6.3.2 管理 Undo .....	101	本章小结 .....	139
6.3.3 管理 Undo 表空间 .....	102	习题 .....	139

## 第二部分 Oracle 数据库应用开发

<b>第 8 章 序列和同义词</b> .....	143	8.2 同义词 .....	146
8.1 序列 .....	143	本章小结 .....	147
8.1.1 创建序列 .....	143	习题 .....	147
8.1.2 使用序列 .....	144	<b>第 9 章 表</b> .....	149
8.1.3 修改序列 .....	146	9.1 创建表 .....	149
8.1.4 删除序列 .....	146	9.1.1 表的类型 .....	149

9.1.2 表的特性	150	第 11 章 视图	188
9.1.3 表的创建	151	11.1 创建视图	188
9.2 修改表	155	11.2 修改视图	190
9.2.1 列的添加、删除和修改	155	11.3 查看视图定义	190
9.2.2 重命名表	158	11.4 视图的 DML 操作	190
9.2.3 改变表的特性	158	11.5 删除视图	191
9.2.4 添加注释	159	11.6 其他视图	192
9.3 删除和查看表	159	11.6.1 内嵌视图	192
9.3.1 删除表	159	11.6.2 对象视图	192
9.3.2 查看表结构	160	本章小结	193
9.4 数据完整性约束	161	习题	194
9.4.1 约束的类别	161	第 12 章 PL/SQL 基础	195
9.4.2 定义约束	161	12.1 PL/SQL 基础	195
9.4.3 添加和删除约束	163	12.1.1 程序结构	195
9.4.4 约束的状态和延迟检查	165	12.1.2 数据类型	196
9.5 分区表	167	12.1.3 声明变量与常量	197
9.5.1 创建分区表	168	12.1.4 变量的赋值	197
9.5.2 维护分区表	172	12.1.5 PL/SQL 中的 SQL 语句	198
9.6 外部表	173	12.2 PL/SQL 控制结构	199
9.6.1 创建外部表	173	12.2.1 条件结构	199
9.6.2 用外部表导出数据	176	12.2.2 循环结构	202
9.6.3 维护外部表	176	12.2.3 GOTO 语句和 NULL 语句	203
本章小结	177	12.3 集合与记录	204
习题	178	12.3.1 联合数组	204
第 10 章 索引	181	12.3.2 嵌套表	207
10.1 概述	181	12.3.3 变长数组	210
10.2 创建索引	181	12.3.4 集合类型的应用	212
10.2.1 B-树索引	182	12.3.5 记录类型	215
10.2.2 位图索引	182	12.4 异常处理	216
10.2.3 基于函数的索引	183	12.4.1 异常概述	216
10.2.4 反向键值索引	183	12.4.2 异常处理过程	218
10.2.5 域索引	183	12.4.3 SQLCODE 和 SQLERRM 函数	221
10.3 修改索引	183	本章小结	222
10.3.1 合并索引	183	习题	222
10.3.2 重构索引	184	第 13 章 游标、存储过程和函数	224
10.3.3 重命名索引	184	13.1 游标	224
10.4 删除索引	184	13.1.1 游标的概念	224
10.5 索引的监视和查询	185	13.1.2 显式游标	224
10.5.1 监视索引	185	13.1.3 隐式游标	229
10.5.2 查询索引信息	185	13.1.4 游标变量	230
本章小结	186		
习题	187		

13.2 存储过程.....	232	16.3.2 PL/SQL 中使用对象.....	272
13.2.1 创建和调用存储过程.....	233	16.4 继承与重载.....	273
13.2.2 修改、查看和删除存储过程.....	235	16.4.1 对象继承.....	273
13.2.3 用 Java 编写存储过程.....	236	16.4.2 方法重载.....	275
13.3 函数.....	239	本章小结.....	276
13.3.1 创建和调用函数.....	239	习题.....	276
13.3.2 修改、查看和删除函数.....	240	<b>第 17 章 包</b> .....	278
本章小结.....	240	17.1 包的创建.....	278
习题.....	240	17.1.1 规范.....	278
<b>第 14 章 触发器</b> .....	242	17.1.2 包体.....	279
14.1 触发器的基本概念.....	242	17.1.3 初始化.....	279
14.1.1 触发事件.....	242	17.2 包的调用.....	280
14.1.2 触发器分类.....	242	17.3 包的管理.....	281
14.1.3 触发时序.....	243	17.3.1 查看包的信息.....	281
14.2 DML 触发器.....	243	17.3.2 修改包.....	281
14.3 INSTEAD OF 触发器.....	247	17.3.3 删除包.....	281
14.4 系统触发器.....	249	17.4 Oracle Database 11g 中的预定义包.....	281
14.4.1 创建系统事件触发器.....	249	17.4.1 DBMS_OUTPUT.....	282
14.4.2 事件属性函数.....	250	17.4.2 DBMS_ALERT.....	283
14.4.3 创建 DDL 事件触发器.....	251	17.4.3 DBMS_JOB.....	285
14.5 组合触发器.....	253	本章小结.....	288
14.6 管理触发器.....	254	习题.....	289
14.6.1 禁用与启用触发器.....	255	<b>第 18 章 Java 开发中的应用</b> .....	290
14.6.2 修改与删除触发器.....	255	18.1 开发环境配置.....	290
14.6.3 重新编译触发器.....	255	18.1.1 配置计算机.....	290
本章小结.....	255	18.1.2 Oracle JDBC 驱动程序.....	292
习题.....	255	18.1.3 导入 JDBC 包.....	292
<b>第 15 章 动态 SQL 操作</b> .....	257	18.1.4 注册 Oracle JDBC 驱动程序.....	293
15.1 动态 SQL.....	257	18.1.5 连接数据库.....	293
15.2 本地动态 SQL.....	257	18.2 创建 JDBC PreparedStatement 对象.....	294
15.2.1 动态 SQL 处理方法一.....	258	18.3 查询数据.....	294
15.2.3 动态 SQL 处理方法二.....	259	18.4 添加数据行.....	295
15.3 DBMS_SQL 包.....	261	18.5 删除数据行.....	296
本章小结.....	263	18.6 更新数据行.....	296
习题.....	264	18.7 通过 Hibernate 操作 Oracle 数据库.....	301
<b>第 16 章 对象</b> .....	265	18.7.1 配置.....	301
16.1 对象概述.....	265	18.7.2 利用 Hibernate 查询数据.....	303
16.2 创建对象类型.....	265	18.7.3 利用 Hibernate 插入数据.....	306
16.3 对象的使用.....	269	18.7.4 利用 Hibernate 更新数据.....	306
16.3.1 数据库表中对象.....	269	本章小结.....	307
		习题.....	307

# 第一部分

## Oracle 服务器管理

- 建立 Oracle 数据库环境
- 常用 Oracle 管理工具
- 静态数据字典和动态性能视图
- 初始化参数文件与控制文件
- 重做日志管理
- 表空间与数据文件
- 安全管理





# 第 1 章

## 建立 Oracle 数据库环境

本章首先简要介绍 Oracle 数据库产品版本以及体系结构, 然后介绍 Oracle 数据库服务器软件安装和客户端配置、数据库的创建和配置方法, 初步建立起 Oracle 数据库环境。

### 1.1 Oracle Database 11g 数据库产品

Oracle 数据库是关系型数据库, 并且支持面向对象和 XML (Extensible Markup Language, 可扩展标识语言) 功能。它是市面上最为流行的大型数据库产品之一, 广泛支持目前流行的操作系统平台, 如各种 UNIX、Linux 操作系统, 以及 Microsoft Windows (32 位和 64 位版本) 操作系统, 并且在各种操作系统下的操作和界面基本一致。

在开发数据库应用程序时, 如果需要支持不同的平台, Oracle 是一个很好的选择。除此之外, 在实际开发阶段, 能够用数据库实现的功能, 要尽可能在数据库中实现, 而不是自行实现。这样就可以把所开发的 Oracle 数据库应用部署或迁移到其他操作系统平台上。正如 Java 应用具有良好的可移植性得益于 Java 虚拟机的支持一样, Oracle 的这一特点, 为我们跨操作系统平台和硬件平台部署或迁移数据库应用提供了可能, 为数据互操作提供了支持。

为了满足不同层次、不同投资规模用户的需求, Oracle Database 11g 提供企业版、标准版、标准版 1 和个人版 4 种安装类型。

- 企业版: 这种安装类型安装的是 Oracle 数据库产品的全部功能。它为企业级应用提供数据管理, 适用于联机事务处理和数据仓库中高性能、高可用性和高安全性要求环境, 能够满足关键任务应用程序需求。

- 标准版: 这种安装类型适用于工作组和部门级应用, 以及中小规模企业。它提供关系数据库管理核心服务和选项, 以及构造关键业务应用所需的功能, 其中包含针对企业级可用性的 RAC (Oracle Real Application Clusters; Oracle 真正应用集群), 它提供了完整的集群件和存储管理功能。

- 标准版 1: 这种安装类型适用于工作组、部门或 Web 应用。它为单服务器环境或高度分布的部门环境提供关系数据库管理核心服务。该安装类型包含构造关键业务应用所需的功能, 是小型企业构造数据管理解决方案的理想选择。

- 个人版: 这种安装类型只用于 Windows 操作系统, 它安装的软件与企业版完全相同, 但只支持单用户开发和部署环境。

### 1.2 Oracle 数据库体系结构

Oracle 数据库服务器主要包含以下元素。

- 安装在计算机上的 Oracle 数据库系统管理软件：用 Oracle 通用安装程序（Oracle Universal Installer, OUI）把光盘介质上的 Oracle 数据库软件安装到数据库服务器。Oracle 数据库软件的安装路径被称作 Oracle 主目录，它通常存储在环境变量 ORACLE\_HOME 中。

- Oracle 实例（instance）：也称作数据库实例，它由安装在计算机上的 Oracle 数据库软件创建。实例由内存结构和后台进程组成。后台进程是操作系统进程或线程，它们执行对数据库的访问、存储、监视、备份、恢复等管理操作。内存结构主要用于缓存信息，以减少物理磁盘 I/O 次数，提高数据库系统的性能。Oracle 实例内存结构中的信息为所有后台进程和服务器进程所共享。该内存结构被称作系统全局区（System Global Area, SGA），由于它是一个共享区域，所以也被称作共享全局区（Shared Global Area）。

- 数据库：Oracle 实例创建的磁盘上一系列物理文件的集合，它们存储用户数据、元数据和控制结构。Oracle 利用元数据（描述数据的数据，如数据字典，它包含数据库的结构、配置和控制信息）管理用户数据。控制结构（如控制文件和联机重做日志文件）用于保证用户数据的完整性、可用性和可恢复性。

- 服务器进程：Oracle 数据库服务器上为连接用户创建的进程，用于实现用户进程和 Oracle 数据库实例之间的通信，它解释和执行连接用户或应用程序调用的 SQL 语句，并检索和返回结果。Oracle 创建服务器进程时要为其分配一个内存区域。这个区域是服务器进程的私有内存区域，被称作程序全局区（Program Global Area, PGA）。PGA 中存储的信息为单个服务器进程所专用，所以该区域又被称作进程全局区（Process Global Area）。

- Oracle Net：这是一个通信组件，它使客户端应用程序与 Oracle 数据库通过网络进行通信。

## 1.2.1 Oracle 数据库物理存储结构

Oracle 数据库文件的存储可以采取以下几种方式。

- 文件系统：这是最常用的文件存储方式，Oracle 数据库内容存储在一个个操作系统文件中。这种存储方式的优点是可以采用大家熟悉的操作系统的文件管理工具和命令来查看、复制数据库文件。

- 自动存储管理（Automatic Storage Management, ASM）：这是 Oracle 设计的一个磁盘卷管理器和文件系统，ASM 是 Oracle 推荐的存储管理解决方案，以替代传统的卷管理器、文件系统和裸设备。

- 集群文件系统（Oracle Cluster File System, OCFS）：这是 Oracle 为集群环境设计的一种共享文件系统，它类似于传统的操作系统文件系统，但可为多个节点所共享。目前 OCFS 只支持 Windows 和 Linux 操作系统。

- 原始分区（Raw Partition，也称作裸设备）：这是原始分区，而不是文件。Oracle 直接在原始分区上存取数据，而不通过操作系统的文件管理系统，所以裸设备不是缓冲设备。所有 I/O 操作都是直接输入/输出，数据没有经过操作系统的缓冲，因此性能会得到改善。

从物理存储来看，Oracle 数据库的存储结构由数据文件、控制文件和重做日志文件组成。

### 1. 数据文件

数据文件包括存储表和索引数据，以及排序和散列等操作的中间结果。一个数据文件只能属于一个数据库，而一个数据库可以包含一个或多个数据文件。

### 2. 控制文件

控制文件是 Oracle 为管理数据库的状态而维护的一个文件。这个文件很小，只有 64MB 左右，它记录数据库的物理存储结构和其他控制信息，如数据库名称、创建数据库的时间戳、组成数据库的各个数据文件和重做日志文件的存储路径及名称、系统的检查点信息等。

Oracle 打开数据库时,必须先打开控制文件,从中读取数据文件和重做日志文件信息。如果控制文件损坏,就会使数据库无法打开,导致用户无法访问存储在数据库中的信息。

控制文件对检查数据库的一致性和恢复数据库也很重要。在实例恢复过程中,控制文件中的检查点信息决定 Oracle 实例怎样使用重做日志文件恢复数据库。

控制文件对数据库来说至关重要,所以 Oracle 支持控制文件的多路存储,也就是它能够同时维护多个完全相同的控制文件拷贝,建立其镜像版本。

一个 Oracle 数据库的控制文件数量、存储位置和名称由数据库的参数文件记录。但当控制文件采用多路存储时,如果其中任一个控制文件损坏,Oracle 实例就无法运行。

### 3. 重做日志文件

Oracle 的重做日志文件记录了数据库所产生的所有变化信息。在实例或者介质失败时,可以用重做日志恢复数据库。

重做日志文件组存储数据库的重做日志信息,这组重做日志文件被称作联机重做日志文件。每个数据库必须至少拥有两组重做日志文件。Oracle 实例以循环写入方式使用数据库的重做日志文件组,当第一组联机重做日志文件填满后,开始使用下一组联机重做日志文件,当最后一组联机重做日志文件填满后,又开始使用第一组联机重做日志文件,如此循环下去。

如果数据库运行在归档模式下,在发生日志文件切换后,填满的重做日志文件被复制到其他地方保存。这些日志文件副本被称作归档日志文件。

### 4. 其他文件

以上 3 种文件属于 Oracle 数据库文件。除此之外,Oracle 数据库管理系统在管理数据库时还使用其他一些辅助文件。这些文件包括(但不限于)以下几种。

- 参数文件:初始化参数文件中的参数定义实例属性,如说明实例使用的内存量、控制文件的数量、存储路径和名称等。在 Oracle 数据库实例启动时首先打开该文件并读取其中的初始化参数值。

- 口令文件:这是一个可选文件,用于存储被授予 SYSDBA、SYSOPER 和 SYSASM 权限的数据库用户及口令,以便在数据库还未打开时用它验证这些具有特殊权限的数据库管理员的身份。

- 警告日志文件:这是一个文本文件,其名称是 alertdb\_name.log (db\_name 是数据库名)。它相当于一个数据库的“编年体”日志,按照时间的先后顺序完整记录从数据库创建、运行到删除之前的重大事项,如可能出现的内部错误或警告、数据库的启动与关闭、表空间的创建、联机和脱机操作等信息。

- 跟踪文件:提供调试数据,其中包含大量的诊断信息。跟踪文件分为两种:一种是通过 DBMS\_MONITOR (Oracle 预定义包)启用跟踪产生的用户请求跟踪文件,DBA 用它可以诊断系统性能;另一种是发生内部错误时自动产生的。我们在通过 Oracle Support 请求解决遇到的严重错误时,需要上传这种跟踪文件。跟踪文件的存储路由 user\_dump\_dest、background\_dump\_dest 和 core\_dump\_dest 3 个初始化参数指定,它们分别存储专用服务器进程产生的跟踪文件,共享服务器进程和后台进程产生的跟踪文件,以及发生严重错误时产生的跟踪文件。

## 1.2.2 Oracle 数据库逻辑存储结构

Oracle 数据库使用一组逻辑存储结构,管理数据文件所组成的物理存储空间。这些逻辑存储结构包括表空间、段、区和数据块。Oracle 使用它们控制对物理存储空间的使用。

### 1. 表空间

每个 Oracle 数据库都由一个或多个表空间 (tablespace) 组成。表空间是一个逻辑存储容器,它位于逻辑存储结构的顶层,用于存储数据库中的所有数据。表空间内的数据被物理存放在数据

文件中，一个表空间可以包含一个或多个数据文件。

在其他数据库系统（如 Microsoft SQL Server）中，一个数据库实例可以管理多个数据库，而每个 Oracle 实例则只能管理一个数据库，但其中可以建立多个表空间。使用表空间主要有以下优点。

- 能够隔离用户数据和数据字典，减少对 SYSTEM 表空间的 I/O 争用。
- 可以把不同表空间的数据文件存储在不同的硬盘上，把负载均衡分布到各个硬盘上，减少 I/O 争用。

• 隔离来自不同应用程序的数据，能够执行基于表空间的备份和恢复，同时可以避免一个应用程序的表空间脱机而影响其他应用程序的运行。

- 优化表空间的使用，如设置只读表空间、导入/导出指定表空间的数据等。
- 能够在各个表空间上设置用户可使用的存储空间限额。

Oracle Database 11g 创建数据库时，将默认创建以下表空间。

- SYSTEM：系统表空间，主要用于存储整个数据库的数据定义信息。
- SYSAUX：SYSTEM 表空间的辅助表空间，用于存储一些组件和产品的数据，以减轻 SYSTEM 表空间的负载，如 Automatic Workload Repository、Oracle Streams、Oracle Text 和 Database Control Repository 等组件，都是用 SYSAUX 作为它们的默认表空间。

- TEMP：临时表空间，用于存储 SQL 语句处理过程中产生的临时数据。
- UNDOTBS1：还原表空间，Oracle 数据库用它存储还原信息，实现回滚操作等。
- USERS：用于存储永久用户对象和数据。

## 2. 段

段（segment）就是占用存储空间的数据库对象。如果我们把表空间看做与应用程序相关，用它们来存储和隔离不同应用程序的数据，那么段就是与数据库对象相关，用于存储和隔离不同数据库对象的数据。Oracle 数据库中的段分为以下 4 种。

- 表段：又称数据段，每个非簇表的数据存储在一个或多个表段内，而簇内所有表的数据则存储在一个表段中。

- 索引段：Oracle 数据库中的每个非分区索引都用一个段来存储其所有数据，而对于分区索引来说，其每个分区的数据则存储在单个索引段中。

- 回滚段：用于存储数据库的还原信息。
- 临时段：用于存储 Oracle 在执行 SQL 语句期间所产生的中间状态数据。

## 3. 区

区（extent）是 Oracle 数据库内存储空间的最小分配单位。一个段需要存储空间时，Oracle 数据库就以区为单位将表空间内的空闲空间分配给段。每个区必须是一段连续的存储空间，它可以小到只有一个数据块，也可以大到 2GB 的空间。

## 4. 数据块

区由数据块（data block）构成，数据块是 Oracle 数据库的 I/O 单位，也就是说，在读写 Oracle 数据库中的数据时，每次读写的数量必须至少为一个数据块大小。

不要把 Oracle 的数据块与操作系统的 I/O 块相混淆。I/O 块是操作系统执行标准 I/O 操作时的块大小，而数据块则是 Oracle 执行读写操作时一次所传递的数据量，Oracle 数据块大小必须是操作系统 I/O 块大小的整数倍。

Oracle 数据块的结构如图 1-1 所示，它由以下几部分组成。

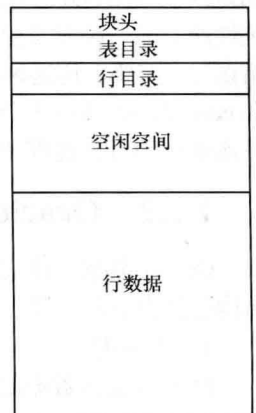


图 1-1 Oracle 数据块结构

- 块头：包含一般块信息，如块的磁盘地址及其所属段的类型（如表段或索引段）等。
- 表目录：说明块中数据所属的表信息。
- 行目录：说明块中数据对应的行信息。
- 空闲空间：数据块内还没有被分配使用的空闲空间。
- 行数据：包含表或索引数据，行数据可以跨越多个数据块。

Oracle 数据库支持的数据块大小包括 2KB、4KB、8KB、16KB 和 32KB 五种。在创建数据库时，初始化参数 `DB_BLOCK_SIZE` 指定数据块大小。该尺寸的数据块被称作数据库的标准块或默认块。数据库标准块大小一旦确定就无法改变，除非重新创建数据库。

在创建表空间时，如果不指定数据块的大小，所创建表空间的块大小将与标准块大小相同。但也可以使用 `BLOCKSIZE` 子句指定表空间的块大小。

数据库管理员（DBA，DataBase Administrator）在指定表空间的块大小时应考虑行数据的长度。虽然 Oracle 允许行数据的存储跨越多个数据块（称作行链接），但这样会降低检索性能，因为从多个数据块检索一行数据所需的 I/O 次数要比从一个数据块检索多，所以一个数据库内的行链接越多，查询的性能就会越低。因此，为了提高性能，DBA 应该根据应用中行数据的长度创建适当块大小的表空间。

### 5. Oracle 数据库物理存储结构和逻辑存储结构之间的关系

本小节的最后，让我们用一个图形总结一下 Oracle 数据库物理存储结构和逻辑存储结构之间的关系，如图 1-2 所示。

- 一个表空间可以包含一个或多个数据文件，在一个表空间内可以存储一个或多个段，所以段数据可以存储在一个数据文件上，也可以存储在一个表空间内的多个数据文件上。
- 每个段中包含一个或多个区，每个区由一个或多个数据块组成。
- 向段分配数据文件内的空闲空间是以区为单位。
- Oracle 数据块是操作系统 I/O 块的整数倍。一个表空间内的所有数据文件只能使用同样的块尺寸。

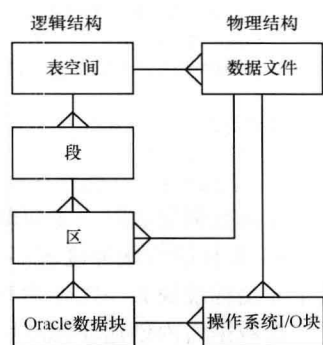


图 1-2 Oracle 数据库物理存储结构和逻辑存储结构之间的关系

## 1.2.3 Oracle 数据库实例

Oracle 实例由内存结构和后台进程组成。实例启动时会向操作系统申请内存，并启动其后台进程。每个实例只能管理一个 Oracle 数据库，但一个 Oracle 数据库可以由一个实例或多个实例（集群环境下）管理。

### 1. 实例内存结构

每个 Oracle 实例都用一个很大的内存结构来缓存数据，这样可以减少磁盘物理 I/O 次数，提高系统性能。Oracle 又把 SGA 分为更多的内存区域，以缓存不同类型的数据。SGA 中的主要区域包括以下几部分。

#### （1）固定 SGA

可以把它看做 SGA 的一个“自启”区域，Oracle 用这个区域来查找 SGA 中的其他区域。

#### （2）数据缓冲区缓存（data buffer cache）

为了减少数据库的物理 I/O 次数，提高性能，Oracle 在从磁盘数据文件检索数据之后或将数据块写入磁盘之前，都要将数据块缓存到数据缓冲区中。由于 Oracle 数据库除标准块（如 8KB）之外，还允许使用其他 4 种非标准块（2KB、4KB、16KB 和 32KB），所以数据缓冲区缓存也分

为标准块缓冲区缓存和非标准块缓冲区缓存。

- **标准块缓冲区缓存:** 通常, 一个默认的数据缓冲区缓存足以满足大多数系统的需要。但正如大家所熟知的, 数据库应用程序中不同表的数据使用频度是不同的, 有些表的数据使用频率极高, 需要长久缓存; 有些表的数据使用频率非常低, 使用之后即可从缓存中清除; 其余数据的使用频度则介于二者之间, 它们中的数据在缓存空间允许的情况下应尽可能长时间地缓存于缓冲区中。如果 Oracle 需要缓存新的数据, 则按照其内部算法把使用频度较低的数据从缓冲区缓存清除, 为其他数据块提供缓存空间。

针对这种情况, Oracle 将标准块缓冲区缓存划分为 3 种: 保持池、循环池和默认池, 分别用于缓存上述 3 种表的数据。这 3 种缓存的大小分别用初始化参数 `db_keep_cache_size`、`db_recycle_cache_size`、`db_cache_size` 设置。

配置多个缓存区可以更充分地发挥缓冲区缓存的效率。在创建表时, 使用 `STORAGE` 子句指定表中数据要使用哪种缓冲区缓存。例如, 下面语句创建的 `tkeep`、`trecycle` 和 `tdefault` 表将分别使用保持池、循环池和默认池。

```
CREATE TABLE tkeep (
  coll char )
  storage( Buffer_pool keep );

CREATE TABLE trecycle (
  coll char )
  storage( Buffer_pool recycle );

CREATE TABLE tdefault (
  coll char )
  storage( Buffer_pool default );
```

如果在创建表时, 未明确指出使用哪种缓冲区缓存, Oracle 则把它放在默认池中。

- **非标准块缓冲区缓存:** 非标准块缓冲区缓存的大小由初始化参数 `db_n_cache_size` 指定, 其中 `n` 是标准块大小之外的其他 4 种尺寸。

在数据库内创建非标准块表空间时, 必须先为这种尺寸的数据块分配缓冲区。例如, 下面的 SQL 语句为 16KB 数据块分配 50MB 的缓存空间。

```
ALTER SYSTEM SET db_16k_cache_size = 50M SCOPE=BOTH;
```

### (3) 重做日志缓冲区 (redo log buffer)

服务器进程把执行数据修改 (如插入、修改和删除等操作) 过程中产生的重做日志写入重做日志缓冲区, 然后由日志写入进程把日志缓冲区内的重做日志写入磁盘中的联机重做日志文件。

重做日志缓冲区的大小由初始化参数 `log_buffer` 指定。Oracle 内部把日志缓冲区看作一个环形区域。当日志写入进程把部分重做日志写入日志文件后, 服务器进程即可循环使用它, 用新的重做日志覆盖旧日志。

### (4) 共享池 (shared pool)

共享池是 SGA 中一个非常重要的区域, 它对 SQL 语句的执行性能有很大影响。共享池的大小由 `shared_pool_size` 参数指定, 它又分为以下几个主要子区域。

- **数据字典缓存:** 在首次执行 SQL、PL/SQL 代码时, 服务器进程首先要解析其代码, 生成执行计划。在解析过程中需要检索 SQL 语句操作的数据库对象及其定义、用户和权限等信息, 这些信息存储在数据库的数据字典内。数据字典缓存用于缓存这部分信息, 以减少解析代码时的磁盘 I/O 次数。

- **库缓存:** 用于缓存解析过的 SQL、PL/SQL 语句的执行计划。服务器进程在执行 SQL、PL/SQL 代码时, 首先从库缓存中查找其执行计划, 如果找到, 则重用该代码, 这称作软解析或

库缓存命中。否则，Oracle 必须生成该代码的执行计划，这被称作硬解析。

- 服务器结果缓存：用于缓存 SQL 语句的查询结果集合和 PL/SQL 函数的结果集。这与数据库缓冲区缓存不同，后者用于缓存数据块。

#### (5) 大型池 (large pool)、Java 池 (Java pool)

大型池是一个可选内存区域，它由 `large_pool_size` 参数设置，用于分配不适用于在共享池内分配的大块内存，如 RMAN 备份所需的缓冲区、语句并行执行所使用的缓冲区等。

Java 池用于存储与所有会话相关的 Java 代码和 Java 虚拟机 (JVM) 内的数据。它由 `java_pool_size` 参数设置。

#### (6) 流池 (stream pool)

Oracle Streams 是 Oracle 提供的一个组件，它允许在不同数据库和应用程序之间共享数据。流池专门为 Oracle Streams 组件所使用，用于缓存流进程在数据库间共享数据所使用的队列消息，它由 `streams_pool_size` 参数设置。

缓存是影响 Oracle 性能的主要因素之一。在服务器内存一定的情况下，合理分配缓存可以大大提高数据库的性能。对于有经验的 DBA 来说，可以采用手工分配方式自己分配各部分的内存量。而对于经验不足的 DBA 来说，则可以利用 Oracle 的自动内存管理方式让其代为管理各部分之间的内存分配。影响 Oracle 数据库内存自动分配的初始化参数如表 1-1 所示。

表 1-1 影响内存自动分配的初始化参数

初始化参数	作用
<code>memory_target</code>	设置 Oracle 系统可用的最大内存量。其值不为零时，Oracle 在运行过程中将根据需要增大或减小 SGA 和 PGA 的值，实现内存的自动管理
<code>memory_max_target</code>	可以指定给 <code>memory_target</code> 的最大值，如果该参数未设置，实例启动时将把它设置为与 <code>memory_target</code> 相同的值
<code>sga_target</code>	其值不为零时，Oracle 将实行 SGA 内存的自动管理，实现对以下内存区域的自动分配，而其他数据库缓冲区缓存、日志缓冲区、固定 SGA 和其他内部区域则不能实现内存的自动分配 标准块的默认池 (DB_CACHE_SIZE) 共享池 (SHARED_POOL_SIZE) 大型池 (LARGE_POOL_SIZE) Java 池 (JAVA_POOL_SIZE) 流池 (STREAMS_POOL_SIZE) 如果这些被自动调整内存池的初始化参数被设置为非零值，Oracle 将把它们用作可调整到的最小值
<code>sga_max_size</code>	指出实例中 SGA 可用的最大内存量。如果该参数未设置，而 <code>memory_target</code> 或 <code>memory_max_target</code> 参数已设置，实例将把 <code>sga_max_size</code> 设置为二者中较大的值
<code>pga_aggregate_target</code>	指出一个实例下所有服务器进程可用的 PGA 内存总量
<code>workarea_size_policy</code>	其值为 AUTO 时，进程所使用的各工作区的内存量将由系统根据 PGA 的总内存量自动调整；其值为 MANUAL 时，各工作区的内存量由 *_AREA_SIZE 参数指定

## 2. 后台进程

Oracle 数据库实例的后台进程是操作系统进程或线程，它们共同实现对 Oracle 数据库的管理功能。每个后台进程只完成一项单独的任务，这使 Oracle 实例具有较高的效率。Oracle 后台进程数量繁多，一些后台进程在每个实例中都会启动，而另一些后台进程则根据条件和配置启动。从



动态性能视图 `v$bgprocess` 可以查询有关后台进程信息，以及实例当前已经启动的后台进程。

一些常见的基本后台进程如下。

#### (1) 数据库写入进程 (database writer, DBWR)

数据库写入进程负责将 SGA 内数据库缓冲区缓存中修改过的数据块写入数据文件。如果需要，可以创建多个 DBWR 进程，让它们共同分担数据写入负载。实例启动的 DBWR 数量由初始化参数 `DB_WRITER_PROCESSES` 指定。在 Oracle Database 11g 中，一个实例最多可以启动 36 个 DBWR (依次编号为 `DBW0`, ..., `DBW9` 和 `DBWa`, ..., `DBWz`，因此，数据库写入进程又被统称为 `DBWn`)。由于 Windows 操作系统采用异步 I/O 方式，所以常常只配置一个数据库写入进程即可满足数据写入需要。

数据库写入进程在以下条件下将把脏数据块写入数据文件。

- 服务器进程找不到足够数量的可用干净缓冲区。
- 数据库系统执行检查点时。

但在写入脏数据块之前，如果 DBWR 发现与数据缓冲区内脏数据块相关的重做日志还没有写入磁盘，它将通知 LGWR，先把重做日志缓冲区内重做日志写入联机重做日志文件，并一直等到 LGWR 写完之后才开始把脏数据块写入磁盘，这称作前写协议 (Write-ahead protocol)。

#### (2) 日志写入进程 (log writer, LGWR) 和归档进程 (archiver, ARCH)

LGWR 负责把日志缓冲区内重做日志写入重做日志文件。发生日志文件切换时，如果数据库运行在归档模式下，将启动日志归档进程，把填写过的联机重做日志文件复制到指定位置进行归档。

#### (3) 检查点进程 (checkpoint process, CKPT)

在 Oracle 数据库内，检查点进程会定期启动，它把检查点信息写入控制文件和数据文件头部，并通知 `DBWn` 进程把脏数据块写入数据文件。`DBWn` 进程的运行又会启动 LGWR 进程将重做日志缓冲区内内容写入重做日志文件，这样就完全同步了数据文件、日志文件和控制文件的内容。

要理解检查点，首先应该了解 Oracle 数据库的事务提交方式。用户提交事务时，Oracle 分配一个系统修改号 (System Change Number, SCN) 用于标识该事务，LGWR 在重做日志缓冲区内添加一项提交记录，之后立即把重做日志缓冲区内数据写入联机重做日志文件。日志写入成功之后，通知用户或应用程序事务提交完成。这时候与该事务相关的数据可能还在数据缓冲区缓存内没有写入数据文件。这种提交方式被称作事务的快速提交。Oracle 之所以采用这种方式，是因为数据的写入是随机的，而日志的写入是顺序方式，所以日志的写入速度会比数据的写入速度快。

如果用户所提交事务的数据还未写入数据文件就遇到实例故障，在下次实例启动打开数据库过程中，它会发现数据库处于不一致状态，需要做实例恢复。此时，Oracle 会读出重做日志文件中的重做日志，再执行一遍 (这就是重做日志名称的由来)，即可恢复用户已经提交的所有事务的数据。所以，采用快速提交方式能够满足事务持久性的要求。

Oracle 数据库的检查点机制保证数据库文件在每个检查点处于同步状态，把所有已提交事务的数据写入数据文件 (这些事务对应的日志在提交事务时已经写入了重做日志文件)，因此实例恢复时只需恢复最后一个检查点后提交的事务，从而缩短实例恢复所需的时间，使数据库能够快速启动。但是，如果检查点进程启动过于频繁，它会大量增加系统的 I/O 次数，从而影响系统的运行性能。

与检查点频率设置相关的初始化参数包括 `LOG_CHECKPOINT_TIMEOUT`、`LOG_CHECKPOINT_INTERVAL` 和 `FAST_START_MTTR_TARGET`。初始化参数 `LOG_CHECKPOINTS_TO_ALERT` 的值决定是否把检查点启动事件写入数据库的警告日志文件。有关这些初始化参数的详细介绍，请