

ORACLE 数据库系统

——管理与应用

刘辰 高月秋 等编著

人民邮电出版社

内容简介

本书集中详细地介绍了 ORACLE 数据库的管理与应用基础, 主要内容包括: ORACLE 数据库的结构、安装、管理及维护, ORACLE 的网络结构、配置及分布式数据库的特性, ORACLE 数据库的存储过程、包和数据库触发器的定义、管理和应用。

本书不仅可以作为 ORACLE 数据库的教材, 也可以作为 ORACLE 数据库应用系统开发人员的详细参考资料, 还可以作为 ORACLE 数据库管理与维护人员的参考书。

ORACLE 数据库系统——管理与应用

◆ 编 著 刘 辰 高月秋 等

责任编辑 刘兴航

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号

北京鸿佳印刷厂印刷

新华书店总店北京发行所经销

◆ 开本: 787×1092 1/16

印张: 14.25

字数: 350 千字

1999 年 3 月第 1 版

印数: 1—3 000 册

1999 年 3 月北京第 1 次印刷

ISBN 7-115-07620-0/TP·989

定价: 31.00 元

前 言

ORACLE 数据库作为关系型数据库的先驱和基于标准 SQL 数据库语言的产品,自推出后,特别是近年来,在邮电、交通、石油、金融等各个领域获得了广泛应用。ORACLE 数据库系统使用方便、功能强大、便于维护、易于推广,比如针对不同的应用需求,ORACLE 分别提供了不同的开发工具,以辅助程序的生成,并且,ORACLE 应用程序具有很好的移植性,可方便地在不同的硬件平台、网络环境中移植。目前,ORACLE 系统已经拥有大量的用户,无论作为 ORACLE 应用的开发人员、管理维护人员、还是 ORACLE 用户等,均需了解掌握 ORACLE。为此我们编写了《ORACLE 数据库系统》这套书。本书系统性、实用性较强,不仅可以作为 ORACLE 数据库教材,也可以作为 ORACLE 数据库应用系统开发人员的全面、详细的参考资料,还可以作为 ORACLE 系统管理、维护人员的参考书。

这套书分为两册,即《ORACLE 数据库系统——基础与工具》和《ORACLE 数据库系统——管理与应用》。其中《ORACLE 数据库系统——基础与工具》按照 ORACLE 数据库系统基础、ORACLE 应用系统开发工具、ORACLE 产品工具的集成等,详细全面地介绍了 SQL 语言、SQL*Plus 环境及命令、PL/SQL 语言、Pro*C 应用、ORACLE*FORMS4.5、ORACLE*REPORTS2.5、ORACLE*GRAPHICS2.5,最后详细介绍了 ORACLE 各工具之间的集成。《ORACLE 数据库系统——管理与应用》按照 ORACLE 数据库的管理、ORACLE 分布式环境、ORACLE 过程化选件等,详细介绍了 ORACLE 数据库的结构、安装、管理及维护,ORACLE 的网络结构、配置及分布式数据库的特性,ORACLE 数据库的存储过程、包和数据库触发器的定义、管理和应用。

这套书由北京邮电大学和北方交通大学组成的编写组策划编写,具体分工是《ORACLE 数据库系统——基础与工具》第一、二、四章由李文生编写,第三、五章由冯凤娟编写,第六章由霍宇红编写,第七章由施海舟编写,全书由李文生统稿和编排。《ORACLE 数据库系统——管理与应用》第一、二章由高月秋编写,第三、四、五章由刘辰编写,全书由刘辰统稿和编排。作者均是 ORACLE 联合大学的教师,具有多年 ORACLE 教学经验,并且具有较丰富的 ORACLE 数据库应用系统开发经验。

本书在编写过程中得到了刘振田、徐晖、黄曼琳、赵海莲、张美祥、酒正超、黄靖宇、纪增良、陈玉琳等同志的大力支持和帮助,在此表示衷心感谢。

本书内容繁多,涉及面广,书中难免有缺点和错误,敬请广大读者批评指正。

作 者

1998 年 6 月

目 录

第一章 ORACLE基本概念和结构	1
第一节 ORACLE基本概念	1
一、RDBMS简介	1
二、数据库的结构	2
三、ORACLE系统结构	6
四、数据存取	9
第二节 数据库的结构	14
一、数据块、区、段	14
二、表空间和数据文件	23
第三节 ORACLE的系统结构	31
一、ORACLE例程和进程	32
二、ORACLE存储结构	37
三、ORACLE配置	42
第二章 数据库的系统管理	46
第一节 基本数据库管理	46
一、数据库管理员的职责	46
二、安装ORACLE	47
三、创建数据库	49
四、启动和关闭数据库	52
第二节 数据库的备份与恢复	55
一、选择数据库的操作方式	56
二、数据库的备份	56
三、数据库的恢复	64
第三节 数据库的安全管理	75
一、数据库安全领域设置和安全管理策略	75
二、数据库用户的管理	79
三、数据库资源的管理	86
四、控制用户特权与角色	93
五、数据库的审计	107
第四节 分布处理和分布式数据库	113
一、管理分布式数据库	114
二、管理表快照	117
第三章 分布式数据库系统	119
第一节 分布式数据库系统概述	119

一、分布式数据库系统的一般概念	119
二、分布式数据库系统的模式结构	120
三、分布式数据库管理系统(D-DBMS)	122
四、ORACLE分布式数据库系统	123
第二节 ORACLE网络配置	125
一、Client/Server、Server/Server模式	125
二、SQL-Net产品简介	126
三、SQL*Net V ₁ 与SQL*Net V ₂ 的比较	127
四、SQL*Net的安装种类	129
五、SQL*Net的配置及使用	132
第三节 分布式数据库系统的实现	149
一、路径	149
二、表复制	154
三、位置透明性	158
四、查询与优化	161
五、事务及其二阶段提交	168
六、并发控制	171
第四章 存储过程与存储函数	175
第一节 概述	175
第二节 存储过程	176
一、存储过程的创建与组成	176
二、存储过程中的参数模式	177
三、例外情况处理	178
四、存储过程的存储及调试	180
五、存储过程的调用	183
六、存储过程的依赖性管理	186
七、存储过程的删除	190
第三节 存储函数	190
第四节 包	191
一、包的定义与组成	191
二、包的创建、修改与删除	192
三、包中的元素及其调用	194
四、包的管理	194
五、包的优点	196
六、ORACLE系统包	196
第五章 数据库触发器	199
第一节 概述	199
第二节 数据库触发器的创建	201
一、建立触发器所需的权限	201
二、命令	201
第三节 数据库触发器的存储与管理	205

一、数据库触发器的存储	205
二、数据库触发器的管理	205
第四节 数据库触发器与存储过程及其与FORM中触发器的区别	209
一、触发器与存储过程的比较	209
二、数据库触发器与FORM中触发器的比较	209
第五节 数据库触发器的应用举例	210
一、数据的安全性控制	210
二、利用数据库触发器进行审计操作	211
三、保证数据的完整性	212
四、保证数据的一致性	213
五、实现表复制	214
六、计算获取数据	215
七、执行事件操作	216

第一章 ORACLE基本概念和结构

第一节 ORACLE基本概念

一、RDBMS简介

ORACLE数据库管理系统（DBMS）是一个关系数据库管理系统。在多用户环境下能管理大量的数据，允许多用户并发访问数据，并能够有效地防止数据的非法存取。自从ORACLE问世以来，尤其从ORACLE版本7 自1994年推出以来，广泛的计算机数据库开发方面的实践证明，ORACLE有其非常重要的特点。

首先，ORACLE能对超大容量的存储空间提供有效的管理和应用控制，能支持几百个GB的数据，在众多用户同时访问某一数据时，能够使数据的竞争达到最小。ORACLE具有较高的事务处理能力，使用户能高速地对数据库进行数据操作。它可以昼夜不停地运行，即使要进行数据库的备份操作，也可以使用户不停止应用。当计算机系统产生故障时，也不会中断数据库的使用。ORACLE能在数据库级或者子数据库级控制数据的应用，DBA可以根据特殊的需要使某些数据可以访问，而另一些数据不可以访问。

ORACLE在数据存储语言、操作系统、用户接口、网络通信协议等方面都依附于工业界可接受的标准，完全符合美国国家标准技术局ANSI/ISO SQL89标准。并且，ORACLE具有安全保障功能，能够防止数据库的非法操作，并能监视和限制数据的存取。

ORACLE还具有client/server（分布处理）应用环境，它提高了ORACLE在计算机网络上的应用能力，增强了分布处理的能力。它能把物理上位于不同计算机上的数据构成一个逻辑数据库，并能做到与非分布处理系统一样的数据一致性和用户透明性。它还具有兼容性、可移植性和可连接性，能在近百种平台上、在各种操作系统上进行数据库的应用和开发，可以使网络中不同的计算机与操作系统共享ORACLE数据库资源。

ORACLE是一个关系型数据库管理系统，关系模型是数据库设计最广泛应用的一种，它的特点是描述的一致性，对象及其联系均用关系描述。用这种关系模型构造的数据库为关系数据库。关系数据库的全部数据操作均是基于关系运算进行的。关系数据库管理系统使物理数据存储和逻辑数据库结构无关，对于所有数据的存取都非常简单。在数据库的设计上更灵活，简化了数据的存储和冗余。所以，一个ORACLE数据库是一个以被处理数据为单位的数据

集合，数据库的目的就是存储和检索相关的信息。

二、数据库的结构

一个ORACLE数据库可分为物理结构和逻辑结构。由于物理结构与逻辑数据库结构是分开的，所以数据的物理存储不会影响逻辑存储结构。

一个ORACLE数据库的物理结构是由建立数据库的操作系统文件决定。每个ORACLE数据库包括三种类型的文件：一个或者一个以上的数据文件；两个或者两个以上的重做日志文件；一个或者一个以上的控制文件。这些文件完成了对数据库信息的物理存储。

1. 逻辑数据库结构

ORACLE数据库的逻辑结构是由表空间和数据库的模式对象组成，即由表空间和视图、索引、聚簇、序列、存储过程等组成的。

表空间是数据库的逻辑存储单位，一个数据库可以有多个表空间。在数据库管理中，将相关的逻辑结构组成一个表空间。数据库与表空间、数据文件之间的关系如图1-1所示。

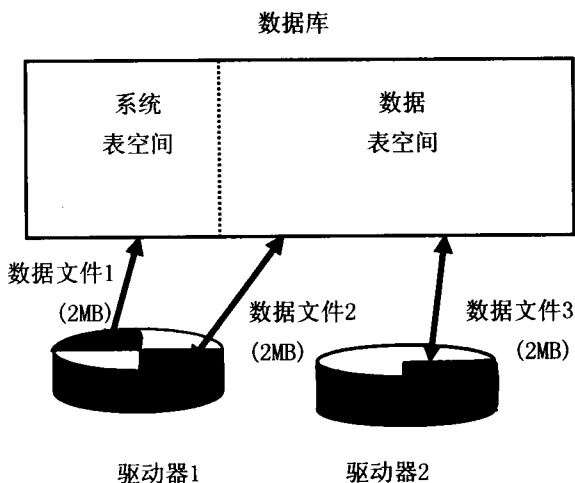


图1-1数据库与表空间、数据文件之间关系

从图中可以看出，一个数据库在逻辑上可以分成一个或者多个表空间。每个表空间上可以建立一个或者多个数据文件，数据文件实际地存储一个表空间中所有逻辑结构的数据。每个数据库表空间的存储容量总和为数据库的总存储空间。表空间可以是联机的，也可以是脱机的。表空间联机时，用户可以存取表空间的信息。然而，当表空间是脱机时，用户就不能存取表空间。有时候，有的表空间可以置成脱机，使数据库的某些数据不能被存取，而数据库的其余部分能够正常地被用户使用。

模式是对象的集合，模式对象是直接表示数据库数据的逻辑结构。模式对象包含的逻辑结构有：表、视图、序列、存储过程、同义词、索引、簇、数据库链路等。一个表空间和

一个模式之间没有关系；在同一模式中的对象可以在不同的表空间中，一个表空间也可以有不同模式的对象。

在ORACLE数据库中，表是数据存储的基本单位，所有数据库用户可存取的数据都存放在这些表中。表的数据是用行和列存储的，每个表都定义一个表名和列集。每一列都有一个列名，并要说明它的数据类型和宽度等。表一旦被建立，就可以每行插入数据，表的行可以进行查询、删除、修改。为使表数据符合某些规则，也可以为表定义完整性约束和触发器。

视图是从一个或者多个表中所取得的特定的数据，视图也可以看做“存储的查询”，视图并不包含数据或者存储数据，更确切地说，它们只是从表中导出数据，这种表叫做视图的基表。基表可以转换成表，也可以转换成视图。和表一样，视图可以被查询、修改、插入、删除；对视图的所有操作实际上是对它的基表的操作。

视图常常被用在限制对表中某些行、列数据的存取，相当于增加了表级的安全性。有时某些数据不希望被别人发现时，可以建立一个视图，将要保护的数据不包括的视图中。视图可以隐藏数据的复杂性，简化用户命令，并可以从不同的视角来表示表中的数据。

序列可以为表的列产生一串序列号，序列号能简化应用编程，序列可以用于同一个表或者多个表。如果同时有两个用户对同一个表插入数据，序列就能自动地为每个用户所要插入的数据行产生正确的序列号。

通常，存储过程、函数、程序包都称为“程序单位”。过程和函数是完成某个任务的SQL和PL/SQL语句。使用过程和函数编程，会使编程变得更加灵活和简单。过程和函数在本质上是相同的，只是函数给调用程序有一个返回值，而过程无返回值。程序包是将相关的过程、函数和其它的包放在一起，在数据库中做为一个单位。包使数据库管理员和应用开发人员便于管理和组织程序，提高了数据库的功能和特性。

同义词是为表、视图、序列、包等程序单位设置的别名。同义词可以用在屏蔽一个对象的所有者，为对象提供公共存取，为远程数据库的表、视图或者程序单位提供位置透明性，并且可以简化数据库用户的SQL语句。同义词可以是公用的或者是专用的。一个用户可以建立一个同义词供自己使用。通常，数据库管理员经常为基本的模式对象建立一个公共同义词，以便使数据库的所有用户都能使用这个同义词。

索引、聚簇、散列聚簇是与表有关的任选结构。这个结构的建立可以提高数据检索的性能和速度。索引能快速地定位所要检索的信息，ORACLE索引对表的数据提供了更快的存取路径，也可以使用部分索引或全部索引迅速定位所需的记录。

索引可以对表的一列或者是多个列来建立。索引建立以后，ORACLE能自动对其进行管理和维护。在对表数据进行操作时，会自动调整索引，并对用户是完全透明的。

索引在逻辑上和物理上都独立于数据，所以，索引可以随时建立和删除，而不会影响表或者其它的索引。如果一个索引被删除，所有的应用程序仍能继续运行，但是存取与本索引有关的数据时会降低速度。

聚簇是存储表数据的一个可选方法。聚簇是一组经常在一个或者多个表中被使用的公共列数据，或者是被多个应用程序共享的列数据。把这些相关的数据存储在磁盘上，可以减少存取磁盘的时间。在聚簇中，和各个表都相关的列称为聚簇键。聚簇键是建索引的，所以，就缩小了检索聚簇的输入输出时间。由于聚簇键的数据只存储一次，对于多个表构成的聚簇

来说, 利用聚簇来存储多个表要比每个表单独存储好得多。聚簇和非聚簇数据的存储情况如图1-2所示。

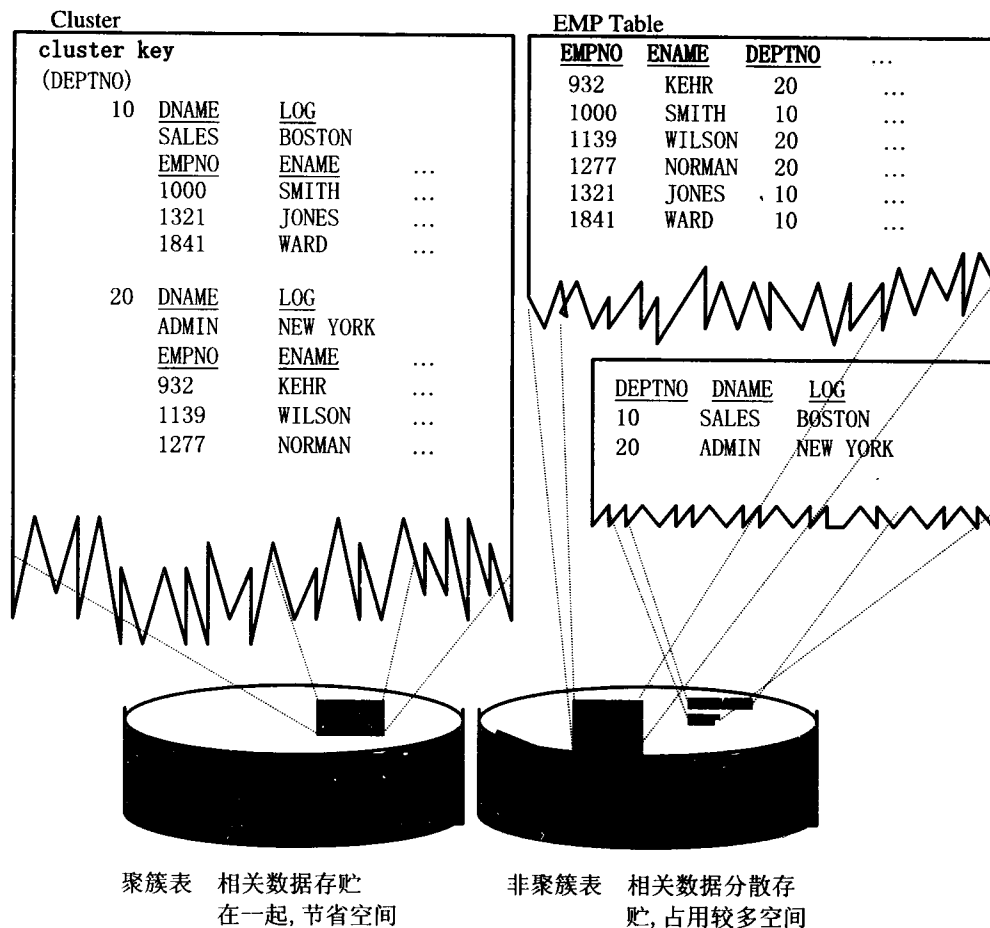


图1-2 聚簇与非聚簇数据的存储

聚簇能提高数据的检索性能, 尤其是进行多表连接时, 这一特点就更为突出。

散列聚簇也是将具有相同列的表数据存放在一起。但散列聚簇中存放的记录的关键值要施加一个散列函数。具有相同散列键值的记录在磁盘中存放在一起。当经常利用等式查询对某个表进行查询时, 使用散列聚簇比使用索引表、索引聚簇会更好更快。这种查询用指定的聚簇键值与散列函数运算, 得出的结果直接指向磁盘的存储区。

数据库链是一个命名的对象, 它描述了从一个数据库到另一个数据库的路径。在分布式数据库中将会使用数据库链。

ORACLE通过使用块、区、段这些逻辑存储结构来管理和控制磁盘空间的使用。一个数据块对应磁盘上的一个物理存储空间, 存储空间的字节数在建立数据库时指定。由若干个数据块构成一个区。若干个区又构成一个数据段。ORACLE中有四种类型的数据段, 这是根据它们

在数据库中的不同作用区分的。

数据段：每个非聚簇表有一个数据段，所有表中的数据按区存放在它的数据段中。每个聚簇有一个数据段，聚簇的每个表都存放在聚簇的数据段中。

索引段：每个索引有一个索引段，索引段中存放着索引的全部数据。

回退段：每个数据库都有一个或者几个回退段，回退段中记录着需要回退的事务的操作，它主要用于回退事务和数据库恢复，是实现读一致性的手段。

临时段：临时段是用来存放SQL语句执行时的临时信息，当SQL语句的操作执行完后，临时段空间返回给系统待将来使用。

ORACLE以区为单位给各种段分配空间，当某段的区用完后，ORACLE会根据需要为该段分配另外的区。所以，在磁盘上一个段的几个区并不一定是相邻的。

2. 数据库的物理结构

ORACLE数据库的物理结构包括数据文件、重做日志文件和控制文件。

每个ORACLE数据库都有一个或者多个物理数据文件，数据库的数据文件中包含了所有的数据库数据。逻辑数据库结构的数据，如表、索引等都物理地存放在数据文件中。一个数据文件只能与一个数据库相关联；一旦建立了一个数据文件，它的大小就不能改变。一个或者多个数据文件就构成了数据库的逻辑存储单位，即表空间。换句话说，一个表空间中有一个或者多个数据文件。

在数据库正常运行时，如果需要存取数据库表中的数据，就必须先将所需要的信息从相关的数据文件中读出，放入内存的高速缓冲区，才能进行存取。对数据文件中的数据进行修改或增加新数据时，并不是立即写入数据文件，而是先将这些数据放入到内存中，由ORACLE的后台进程确定后一次性写入磁盘，这样做就减少了磁盘的输入输出次数，提高了性能。

每个ORACLE数据库至少有两个重做日志文件。重做日志文件记录所有对数据库信息的修改。当数据库产生故障时，可以利用重做日志文件中的数据对数据库进行恢复。由于重做日志文件对保护数据库至关重要，所以，ORACLE提供了镜像功能，把一个或者是多个重做日志文件的拷贝分别存储在不同的磁盘上。

重做日志文件只用于系统故障或者是介质失败时，对数据库进行恢复。

每个ORACLE数据库有一个控制文件。控制文件中记录数据库的物理结构。数据库名、数据文件名和重做日志文件名以及数据库的建立时间等信息都包括在控制文件中。如果数据库的物理结构发生变化，ORACLE就会自动修改控制文件的内容。对数据库进行恢复时也要用到控制文件。

每个ORACLE数据库都有一个数据字典。数据字典是针对某一个数据库表和视图的一些信息，即数据库的逻辑结构和物理结构的信息。另外，在数据字典中也存储着ORACLE数据库合法用户信息，对表定义的完整性约束信息和模式对象的存储空间分配信息等。

数据字典是在数据库建立时创建的。在数据库运行期间，ORACLE会自动根据情况修改数据字典的内容。数据字典对数据库的运行非常重要，ORACLE要根据数据字典的信息，决定数据库的工作，要读取数据字典的信息，验证模式对象和用户对模式对象存取的合法性。

三、ORACLE系统结构

ORACLE系统结构主要是指管理数据库所使用的存储器结构和进程结构。ORACLE能支持多用户并发地访问一个数据库，并且在多用户、多应用程序并发操作的情况下能高效地工作，所以，ORACLE的核心就是通过存储器和进程来完成各种功能。我们所说的存储器，是存在于主存储器中。进程是计算机内存中工作的一些作业和任务。存储器和进程的示意图如图1-3所示。

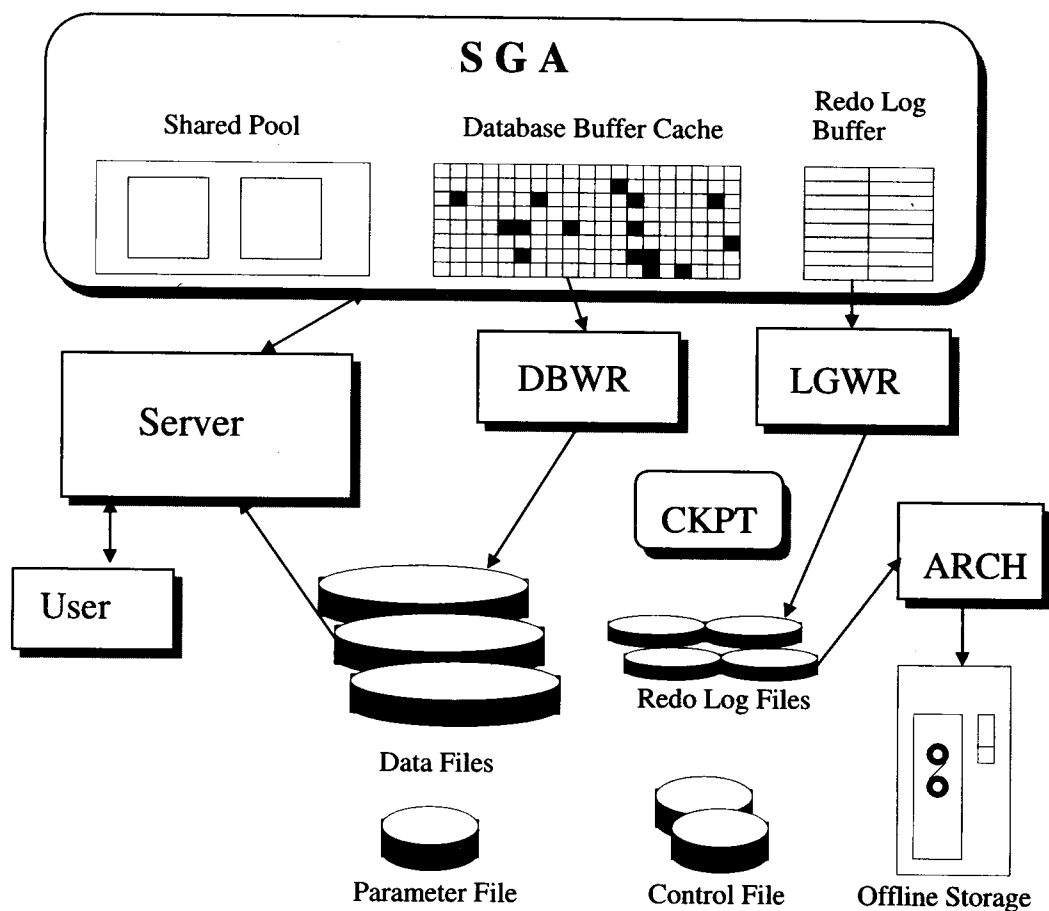


图1-3 存储器和进程示意图

1. 存储器结构

存储器主要由系统全局区、程序全局区组成。在存储器内存放程序代码和多用户共享数据。

系统全局区（SGA）是ORACLE分配的共享内存区，在这个存储区内存放一个ORACLE例程的控制信息和数据。每次ORACLE启动时，都将分配系统全局区（SGA）并启动ORACLE后台进

程。所以，一个SGA区和ORACLE的后台进程就叫做一个ORACLE例程（instance）。SGA区在例程启动时分配而在例程关闭时撤消。每个ORACLE例程都有一个SGA区。

SGA区中的数据供同时连接到数据库的所有用户共享。为了优化系统的功能，减少磁盘的输入输出操作，SGA区的容量应尽量大，以便存放更多的数据。根据存放在SGA区中的数据，SGA区又可以分为数据库缓存区、重做日志缓冲区和共享池。

数据库缓存区中存放近期经常使用的数据库的数据块。重做日志缓冲区中存放对数据库进行修改的记录。共享池是SGA的一部分，共享SQL区就包含在共享池中，在共享SQL区中存放着用于处理提交给数据库的SQL语句、语句的分析树和执行计划。一个共享SQL区可以供同时发出一条语句的多用户应用程序共享。

程序全局区（PGA）也是内存缓冲区，它存放着服务器进程的数据和控制信息。当一个服务器进程被启动时建立PGA，PGA中存放的信息与ORACLE的结构有关。

2. 进程

进程是操作系统中执行一组操作的机制。有些操作系统中将进程称为作业或者任务。一个进程在运行时有自己的存储区。

ORACLE有两种类型的进程，即用户进程和ORACLE进程。

用户进程的建立是为了执行应用程序的软件代码，如执行PRO*C程序和ORACLE工具。用户进程还通过程序接口，管理用户进程与服务器进程间的通信。

ORACLE进程是由完成特定功能的一组进程组成。根据每个进程所完成的特定操作，ORACLE进程可以分成许多类型的进程，其中包括服务器进程和一组后台进程。

服务器进程的建立，是为了处理从用户进程发来的请求，服务器进程还负责与用户进程间的通信，ORACLE系统与用户进程的交互对话等工作。如果用户需查询不在SGA缓冲区中的数据，服务器进程就会从数据文件中读出适当的数据块放入SGA缓冲区，提供给用户查询。

ORACLE对服务进程可配置多个用户进程。在专用服务器方式下，一个服务器进程只处理单一用户进程的请求。在多线索服务器方式下，允许许多用户进程共享少数的服务器进程。这样可以减少服务器进程的数量，充分利用系统的有效资源。

在有些系统中，用户进程和服务器进程是分开的，而有些系统中，用户进程和服务器进程又可能合成一个进程。多线索服务器方式下，用户进程和服务器进程是独立的。如果用户进程和服务器进程运行在不同的机器上，那么用户进程和服务器进程也必定是相互独立的。

ORACLE为每个例程建立了一组后台进程，每个后台进程对每个用户进程所执行的程序完成特定的功能，后台进程异步地执行输入输出操作和监控其它的ORACLE进程，提高并行性和可靠性。

每个ORACLE例程可以使用几个后台进程。这些后台进程分别为：DBWR、LGWR、CKPT、SMON、PMON、ARCH、RECO、Dnnn和LCKn。

DBWR进程是数据库写入器进程，它将已经修改的数据块从数据库缓冲区写到数据文件中。当事务被提交时，DBWR不需要进行写入操作。DBWR为减少磁盘的I/O操作，只是在有许多数据需要从SGA读出、数据库缓冲区中剩余空间太少时才进行磁盘写操作，首先将最近不经常使用的数据写入到数据文件中。

LGWR进程是将重做日志写入磁盘。重做日志数据在SGA的重做日志缓冲区中，当事务被

提交和重做日志缓冲区填满时, LGWR将重做日志数据写入到联机的重做日志文件中。

CKPT是检查点进程。在规定的时间内, DBWR进程将所有已修改的SGA中的数据写入到数据文件, 这个事件即为检查点。检查点进程担负着通知DBWR进程和修改所有的数据文件和控制文件的任务。CKPT是可选的, 如果CKPT不存在, 则由DBWR完成CKPT的功能。

SMON是系统监控进程。在ORACLE例程启动时, 完成例程的恢复。在多例程系统中, 一个例程的SMON也能够完成对另外已失败例程的恢复。SMON还能清除不常使用的临时段, 恢复由于刷新和例程恢复时产生的死事务。并能对存储空间进行有效的管理。

PMON是进程监控进程。当用户进程失败时, PMON完成进程恢复。释放失败的用户进程所占有的资源。PMON还监控调度进程和服务器进程, 一旦发现这两个进程失败, 则负责重新启动。

ARCH是归档进程。当联机的重做日志文件填满时, 归档进程拷贝联机重做日志文件去归档存储。归档进程只有在数据库的重做日志用在归档方式下时才有效。

RECO (Recoverer) 是恢复进程。RECO的作用是恢复由于网络故障或系统故障, 而在分布式数据库中挂起的分布事务。

Dnnn是派遣进程, 是可选的后台进程。在多线索服务器结构中才能用到此进程。对每一个通讯协议, 至少要建立一个派遣进程。每个派遣进程将用户进程发来的请求连接到相应的共享服务器进程, 并且将应答信息返回给用户进程。

LCKn是锁进程, 当使用ORACLE并行服务器时, 共有10个锁进程作为内部例程锁。

3. ORACLE例程

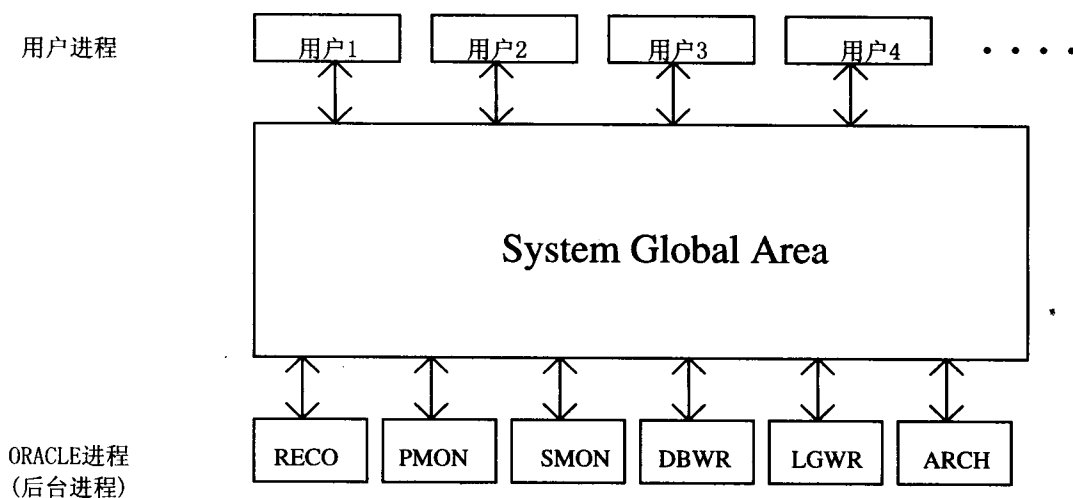


图1-4 多进程的ORACLE例程

每次ORACLE启动时, 将分配系统全局区并启动ORACLE后台进程。内存缓冲区和后台进程就称为一个ORACLE例程。一个多进程的ORACLE例程如图1-4所示。

在有些硬件结构中, 允许多个计算机共享数据、软件和外设, ORACLE具有这种先进的并行服务器结构的功能, 可以运行多个例程, 让这些例程共享一个物理数据库。并行服务器还

允许用户在多台机器上访问同一个数据库。

四、数据存取

1. SQL语句及SQL语句的处理

(1) SQL语句

SQL 是一种功能完善的数据库存取语言，是关系数据库管理系统的标准语言。ORACLE数据库的所有操作都是运用SQL语句执行的。根据SQL语句的功能，ORACLE的SQL语句可以分成五种类型：数据定义语句（DDL）、数据操纵语句（DML）、事务控制语句、会话控制语句、系统控制语句和嵌入语句。

①数据定义语句（DDL）

数据定义语句用来定义、修改模式对象的结构、删除模式对象。DDL语句所执行的主要操作有：

- 创建、修改、删除模式对象及数据库结构（CREATE, ALTER, DROP）。
- 改变模式对象名。
- 删除模式对象中的所有数据，不删除结构。
- 对模式对象做分类统计，列出被修改的行。
- 授予或者收回特权和角色。
- 跟踪统计。
- 向数据字典增加注释。

②数据操纵语句（DML）

DML 语句用于操纵数据库数据。对数据库进行查询、修改、插入、删除等操作。DML 所完成的主要操作有：

- 删除表或视图中的行。
- 查看SQL语句的执行计划。
- 向表或视图插入新的数据行。
- 封锁表或视图，临时限制其他的用户存取被封锁的表或视图。
- 从一个或者多个表中检索数据。
- 改变表或视图中现存行的列值。

③事务控制语句

事务控制语句管理由DML语句所产生的变化，允许用户或应用开发者将这些改变归为逻辑事务。事务控制语句完成的操作有：

- 使事务的改变成为永久。
- 回退整个事务或者从保留点后回退。
- 设置保留点。
- 建立事务特征。

④会话控制语句

会话控制语句允许用户控制当前会话的特性，包括角色和语言设置。会话控制语句有两个：ALTER SESSION 和 SET ROLE。

⑤系统控制语句

系统控制语句改变ORACLE服务器例程的特性。只有一个系统控制命令，即ALTER SYSTEM，该命令允许修改最小共享服务器数量，杀死一个会话和它的操作。

⑥嵌入SQL语句

嵌入SQL语句将DDL、DML、和事务控制语句组合在一起构成过程化语言程序。常见的嵌入SQL语句有OPEN、CLOSE、EXECUTE、FETCH。嵌入SQL语句能完成如下操作：

- 定义、定位和释放光标。
- 说明一个数据库名并与ORACLE联接。
- 指定变量名、初始化描述、指出错误和报警条件。
- 分析和执行SQL语句，从数据库中检索数据。

(2) SQL语句的处理

在用户进程中，由用户的应用程序发出SQL语句，向服务器进程发出请求。ORACLE的服务器进程对用户进程的SQL语句所做的工作如下：

①语句分析

语句分析是处理SQL语句过程的第一步。当应用程序发出一条SQL语句时，此应用程序就产生了一个分析调用给ORACLE的服务器进程。服务器进程执行下列操作：

- 检查语句的语法和语义的合法性。
- 判断发出这条语句的用户进程是否有权执行该条SQL语句。
- 为这条语句分配一个专用的SQL区。

ORACLE还要判断在共享SQL区中，是否已经存在这条语句的语句分析树，如果语句分析树存在，用户进程就使用这条语句的分析树，立即执行这条语句。如果语句分析树不存在，服务器进程就做以下的工作：

- 产生语句的语句分析树。
- 用户进程为SQL语句在高速缓冲区分一个SQL区，并将这条语句的语法分析树存放在共享SQL区中。

用户进程产生的SQL语句分析调用与ORACLE对SQL语句的实际语法分析是两个完全不同的操作。用户应用程序的分析调用是由应用程序为这条SQL语句分配一个专用SQL区。一旦一条语句有了专用的SQL区，这条语句就可以执行。当执行这条语句之后，如果再遇到相同语句，就不再产生分析调用。

由ORACLE的服务器进程完成的语句分析操作是为SQL语句分配一个共享SQL区，一旦为这个语句分配了共享SQL区，那么这条语句就可以被执行。该语句之后再出现相同的语句时就不再进行语句分析操作，而是直接执行这条语句。

服务器进程做完语法分析后，要查询数据字典，检查用户进程发出的SQL语句的实体是否存在，是否有权执行这条语句，并且决定执行语句的最有效查找路径。

当服务器进程确定了语句的分析树后，将分析树存放在共享SQL区中，并且制定该语句的执行计划。

②语句的执行

服务器进程将存放在共享SQL区中的分析树取出，放入数据库高速缓冲区，并且执行真正的数据库读写操作。如果应用程序所用的数据不在数据库高速缓冲区中，server就会将所需的数据块从磁盘中的数据文件读出，放入SGA区的数据库高速缓冲区，然后再执行语句的操作。

③返回执行结果

SQL 语句执行后，server进程根据用户进程的请求，将SQL语句的执行结果返给应用程序。

2. 事务及事务与数据的一致性

一个事务是由一个或者一串SQL语句组成。事务是一个逻辑工作单位。每一个事务的全部操作就是一个事务的整体。事务可以提交和回退。事务提交，就是将该事务中所有SQL语句对数据库所做的操作变成永久。一个事务提交后，另一个事务就开始了。

事务回退，就是撤消该事务的SQL语句对数据库所做的所有操作，使对数据库所做的改变无效，数据库中的数据仍保持原样不变。

对于包含很多SQL语句的长事务，可以在事务中使用几个保存点，对所进行的操作加以标记。这样，当事务的某些操作产生故障需要回退时，就可以从保存点之后进行回退，而保存点以前的正确操作仍然有效。特别是用户对数据库进行一系列修改操作时，使用保存点可以使修改操作产生某些错误时，不再从头执行每个SQL语句。

由于使用了事务这个工作单位，使数据库用户和应用开发者具有了保证数据一致性的能力。一个长的事务，包含很多的SQL语句，执行全部的SQL语句，才能完成这个事务的全部功能。所以，一个事务的SQL语句，或者全部执行，或者都不执行，所有SQL语句的操作是一个整体。在事务的执行过程中，个别语句也可能会破坏数据的一致性，但事务的执行结果是将数据库从一个一致状态转到事务执行以后的一致状态，数据库的数据永远保持一致性。

3. 数据的完整性和完整性约束

保证数据的完整性非常重要。维护数据的完整性就是保证数据符合一定的业务规则。业务规则是用户定义的完整性。如果在数据库运行中，用户操作产生错误或者系统产生故障，就会破坏数据的完整性，此时系统就会回退这个并且返回错误信息。为了管理数据库数据的完整性规则，ORACLE定义了一些完整性约束。

完整性约束是对表数据的约束。如果表中的数据不满足约束条件，那么这个完整性约束就不能被执行。完整性约束建立以后，如果任何的数据管理语句不符合这个完整性约束，那么这个语句将被回退并且给出错误信息。针对表建立的完整性约束的定义，作为表定义的一部分存储在数据库的数据字典中，以使所有的数据库应用都必须遵守相同的规则。如果修改约束规则，只需对数据库级进行修改，而不必对每个数据库应用进行修改。ORACLE支持的完整性约束有：

- NOT NULL 不允许表列值空
- UNIQUE 不允许一列或者几列的值重复
- PRIMARY KEY 不允许一列或几列的值重复或者空
- FOREIGN KEY 要求表的一列或几列的值与相关表中满足UNIQUE或者PRIMARY KEY的关