

翻

ORACLE 关系数据库管理系统

用户手册

(7)

数据库管理员手册

《计算机技术》编辑部

目 录

第一部分：ORACERDBMS 介绍

第一章 什么是 ORACLE	(1)	四个后台进程	(5)
ORACLE 产品取名方法	(1)	提前异步读 (ARH)	(5)
ORACLE 产品系列	(1)	前映象写入 (BIW)	(5)
SQL 语言：所有实用程序的 基础	(1)	清理 (CLN)	(5)
ORACLE ROBMS	(2)	第二章 数据库管理员 (DBA) 的任务	(5)
“Easy”ORACLE 产品	(2)	SYS 和 SYSTEM: ORACLE	
“SQL”ORACLE 产品	(2)	DBA 用户	(6)
“Pro”ORACLE 产品	(3)	用户 SYS	(6)
ORACLE 版本编号方法	(3)	用户 SYSTEM	(6)
ORACLE 术语和概念介绍	(3)	DBA 特权	(7)
基本术语	(3)		

第二部分：使用 DBA 工具

第三章 利用 IOR 程序来启动和停止		INIT.ORA 参数细述	(11)
ORACLE	(8)	使用不同的参数文件	(16)
IOR 命令和参数	(8)	在 IOR 期间出现的错误信息	(16)
初始化 ORACLE (IOR INIT)	(9)	SGI 实用程序	(16)
暖启动 ORACLE (IOR WARM)	(9)	第四章 ORACLE 显示系统实用程序	
停止 ORACLE (IOR SHUT)	(9)	IOPSF	(19)
清理 ORACLE (IOR CLEAR)	(10)	调用 ODS	(19)
IOR 中的 DBA 选择项	(10)	使用 ODS 记录文件	(20)
IOR 中的 SHARED 选择项	(10)	ODS 显示屏幕	(20)
INIT.ORA 参数文件	(10)	前映象显示	(20)
调整 INIT.ORA 参数	(11)	I/O 显示	(22)
依赖于操作系统的参数	(11)	封锁显示	(23)
		进程显示	(24)
		摘要显示	(24)
		表显示	(25)
		用户显示	(26)
		第五章 后映象日志 (AIJ)	(28)

后映象日志的功能	(28)	CRT 使用的数据库表	(33)
后映象日志	(28)	表 SYSTEM.CRT	(34)
使用 AIJ 的前提	(28)	表 SYSTEM.CRTBOX	(34)
接通 AIJ	(28)	表 SYSTEM.CRT-PRO-	
影响日志的 INIT.ORA 参数		DUCTS	(36)
.....	(29)	表 SYSTEM.ESC	(36)
运用日志文件 (调用 AIJ)	(30)	表 SYSTEM.CRT-	
AIJ 命令及其参数	(30)	TYPE	(36)
在 AIJ 的第一趟和第二趟期间		表 SYSTEM.GOTO-LC	(37)
发生的事	(30)	增加新的终端定义	(37)
送到操作员控制台的讯息	(31)	利用 CRT 表	(38)
AIJ 错误信息	(31)	利用 SQL * PLUS	(40)
第六章 修改显示终端 (CRT 实用程序)		指定光标坐标	(40)
.....	(32)	修改现有的终端定义	(40)
什么是 CRT 实用程序	(32)	运行 CRT 程序	(40)
哪些 ORACLE 程序使用 CRT		安装 / 初始化 CRT (DBA 的职责)	
.....	(32)	(41)
CRT 所支持的终端	(33)	对使用 CRT 的几点提示	(41)
缺省的定義	(33)		

第三部分：ORACLE 数据库结构

第七章 ORACLE 数据字典	(42)	如何使用空间	(49)
什么是数据字典	(42)	空间如何回收	(50)
如何建立与更新数据字典	(44)	空间定义	(51)
对个别字典表格应当注意的事	(44)	空间定义 DEFAULT	(51)
增加新的数据字典项	(44)	空间定义 TEMPTABL 3	(51)
人工地更新数据字典	(44)	列出当前空间定义	(52)
第八章 数据是如何储存在数据库		创建一个空间定义	(52)
中	(44)	使用一个空间定义	(53)
数据库文件	(44)	当创建表时指定分区	(53)
分区和文件	(44)	改变空间定义	(54)
SYSTEM 分区	(45)	删去空间定义	(54)
增加分区和文件	(45)	当表超过最大空间 (EXTENT)	
实用 CCF 程序	(46)	(54)
分区满了时	(46)	ORACLE 数据类型	(55)
表	(47)	CHAR 数据类型	(55)
非聚集数据块的逻辑格式	(48)	NUMBER 数据类型	(55)
ORACLE 所需要的开销	(48)	内部数值格式	(55)

有关数值的比例因子及精度	(56)	单用户调整	(70)
DATE 数据类型	(56)	索引	(70)
LONG 数据类型	(56)	索引、键、外来键	(70)
RAW 和 LONG RAW 数据类型	(57)	建立索引	(70)
SQL/DS 兼容的数据类型	(57)	并置索引	(71)
数据类型 ROWID	(58)	压缩型和非压缩型索引	(72)
数据转换	(59)	写 SQL 语句以利用索引的好处	(73)
临时表	(59)	单个索引	(74)
需要临时表的操作	(60)	索引和空值	(75)
如何使用临时表	(60)	一个表上有多个索引	(75)
空间定义 TEMPTABLE	(60)	在多个索引中进行选择	(75)
第九章 一致性和并行性	(60)	抑制索引的使用	(75)
逻辑工作单位 (Transaction)	(60)	VALIDATE INDEX 命令	(76)
COMMIT WORK 语句	(61)	索引的内部结构	(76)
当你交付 (commit) 时发生的	(61)	簇 (clusters)	(77)
事	(61)	什么是簇	(77)
ROLLBACK WORK 语句	(61)	簇数据块的逻辑格式	(77)
调用 COMMIT 和 ROLLBACK	(61)	簇键	(77)
显式交付/回退	(61)	建立簇	(78)
隐式自动交付/回退	(61)	确定逻辑块的大小	(78)
前映象文件	(62)	簇和空间定义	(79)
监视前映象文件	(62)	使用簇时的注意事项	(79)
扩大前映象文件	(62)	簇的加载	(79)
读一致性	(63)	簇的删除	(79)
ORACLE 锁	(63)	ORACLE 上下文空间	(79)
锁的类型	(63)	操作符的优先次序	(80)
内部锁	(63)	各种 SQL 语句的优化	(81)
锁的类型比较	(64)	自动实施的优化	(81)
字典语法分析锁 (DDL 锁)	(65)	优化查询(SELECTS)	(81)
表/行锁 (DML)	(66)	优化的 NOT	(82)
在 SQL*Forms 中的锁	(68)	优化 OR	(82)
对加锁的限制	(69)	ORACLE 分类/合并例程	(83)
死锁检测	(69)	优化 ORDER BY	(83)
对加锁的一般性建议	(69)	优化 GROUP BY	(83)
第十章 调整系统性能	(69)	优化连接	(84)
		无索引连接	(84)
		索引连接	(84)

数组处理 (85)

第四部分 ORACLE RDBMS 安全性管理

第十一章 用户的注册和退出 (86)	表和视图的审计 (90)
用户的登记 (86)	设定系统审计选择 (91)
有 CONNECT 特权的用户	审计与数据字典 (92)
..... (86)	审计什么动作 (92)
有 RESOURCE 特权的用户	审计追踪表 (AUDIT-TRAIL)
..... (87) (92)
有 DBA 特权的用户 (87)	罗列当前的审计选择 (93)
自动注册 (87)	罗列 ORACLE 系统审计选择
修改口令 (88) (94)
用户的除名 (88)	函数 USERENV..... (95)
特殊“用户”PUBLIC (88)	附录 A. 保留字(略) (95)
公共同义词 (88)	附录 B. 数据字典表定义 (95)
LOGIC.SQL 文件 (89)	附录 C. 限制 (95)
第十二章 使用 ORACLE 审计手段	附录 D. INIT.ORA 参数一览(略) ... (96)
..... (89)	附录 E. 使用 CRT 的终端功能码 ... (96)
审计 (AUDiting) (89)	附录 F. Oracle 公司提供的终端
允许系统审计 (90)	定义 (97)

第一部分：ORACLE RDBMS 介绍

这一部分对 ORACLE 产品做了概述。它介绍了一些公共的 ORACLE 名词术语，概括了称之为数据库管理员 (DBA) 的人的职责。

第一章 什么是 ORACLE

本章介绍各种 ORACLE 产品，这些产品构成了 ORACLE 产品系列。对每一个产品作了简要的说明。它还介绍了 SQL 语言，后者乃是 ORACLE 的基础。定义了基本的 ORACLE 术语。

ORACLE 产品取名方法

ORACLE RDBMS 第五版发行时，提出了若干新的 ORACLE 产品。ORACLE 产品的取名是使用这样一种方法：指明该产品的类型 (type) 和级别 (LEVEL)。用一组前缀来指明级别，如：

EASY 指明这是全屏幕产品，提供了许多选择，以此来引导用户开展工作。通过菜单以及详细的联机求助手段和信息，来实现上述目的。主要对象是 ORACLE RDBMS 的新用户，不经常上机的用户，或不熟悉计算机的人。

SQL 指明这是交互式的即命令驱动的产品。因此对 SQL 和 ORACLE 产品要有较多的了解和专业知识。对象是对 ORACLE, SQL, 和数据库都有相当多经验的用户。

Pro 指明这是一个 ORACLE RDBMS 程序设计接口。Pro 接口不但要求对 SQL 和 ORACLE 有经验，而且要求有程序设计的专门知识。这些接口是为程序员提供的。

ORACLE 产品系列

本节介绍各种产品，它们被总称为

ORACLE 数据库管理系统。许多用户不需要使用全部的 ORACLE 产品；不过用户可以利用不同的产品，通过各种途径来完成各种任务。

SQL 语言：所有实用程序的基础

ORACLE RDBMS 的心脏是 SQL 语言，这是一种类英语数据库语言，发音时念作“scquel”。SQL 相当简单，故初学的用户可以在很短的时间内学会如何存取数据。另一方面 SQL 又相当有力，能为程序员提供他们所需要的全部能力和灵活性。

SQL 是 IBM 公司开发和定义的，已被美国国家标准局推荐作为关系数据库管理系统的标准语言的基础。

SQL 语言的语句，常常被划分为四类：

查询：这种语句根据某种组合，某个表达式，或某种次序，来减缩现存的数据。查询语句总是用保留字 SELECT 打头，后面是所想要的数据库，以及包含源数据的表或视图。如：SELECT ENAME, MGR FROM EMP;

查询不改变数据；它们只是检索数据。

DMI：数据操纵语言语句，用来以三种基本方法改变数据：

• INSERT 插入新的数据行到一个表

- UPDATE 更新现有的行中列的值
- DELETE 从表中删除行

DDL: 数据定义语言语句, 用来建立数据库实体, 并在这些实体不再被需要时, 撤除它们。DDL 语句包括 CREATE TABLE, CREATE VIEW, CREATE INDFX, CREATE SYNONYM, ALTER TABLE, 和相应的 DROP 语句。

DCL: 数据控制语言语句, 用来获得和放弃存取数据库的特权 (例如 GRANT CONNECT, RESOURCE, DBA), 以及看数据库数据 (例如 GRANT SELECT 和 REVOKE DELETE)。DCL 语句使一个用户能够让别的用户去看, 去改变, 去使用他的表里的数据, 甚至把他自己的特权授予别的用户 (GRANT SELECT...WITH GRANT)。

DCL 语句常与 DDL 语句组合起来用。数据控制语言还包括 COMMIT, ROLLBACK 和 LOCK TABLE, 它们控制数据操纵事务如何和何时出现。还包括 AUDIT 语句。

当你对 ORACLE RDBMS 比较熟悉后, 你会发现, DML, DDL 和查询语句对数据库而言具有不同的含义。因此, 假定你知道哪一些 SQL 语句属于哪一个范畴的话, 本书可以作为对这些语句类 (DML, DDL, 等等) 的临时的参考书。

ORACLE RDBMS

ORACLE RDBMS 是主要的 ORACLE 产品。它包括核心数据库管理程序以及若干功能部件, 后者用来帮助用户和 DBA 们维护、监视和使用数据。下面是 ORACLE RDBMS 产品中所包含的实用程序 (其中 DBA 要用的实用程序, 在本手册中加以介绍) :

IOR DBA 实用程序, 用来启动、停止和初始化 ORACLE 系统。

SGI DBA 实用程序, 用来描述 ORACLE 所用的共享存贮区域。

ODS DBA 实用程序 (ORACLE 显示系统), 用来监视用户和 ORACLE 处理。

CRT DBA 实用程序, 用来为全屏幕产品 (如 SQL * Forms) 调整屏幕显示特性, 或修改小键盘功能定义。

AIJ DBA 实用程序 (后映象日志), 用来记住对数据所做的改变, 使得在磁盘发生故障时能恢复数据。

Export / Import 用户实用程序, 用来把数据送到文件或从文件中取来。因此可以在操作系统之间或 ORACLE 数据库之间传送数据。

ODL 用户实用程序 (ORACLE Data Loader), 用来从 O/S 标准文件把数据加载到库。

"Easy" ORACLE 产品

Easy * ORACLE 产品为初学用户提供各种功能。并不是全部 "Easy" 产品都可以在每一个操作系统上使用。

• Easy * SQL

Easy * SQL ORACLE 是 Easy * SQL 的一个 MS-DOS 版本, 它只包含 RDBMS 和 Easy * SQL。

"SQL" ORACLE 产品

SQL * 产品是 ORACLE 产品系列中的主流产品, 提供了许多复杂的方法来存取数据。但不是所有 "SQL" 产品都可用于每一个操作系统。

SQL * Plus 这是一个交互式的命令驱动的 ORACLE 接口, 适用于特别的查询和报告生成。

SQL * Forms 一个全屏幕表接口, 允许用户建立, 修改, 和使用全屏幕的表。

SQL * Calc 一个完全与 ORACLE 数据库融为一体的电子表格接口。

SQL * Menu 是一个产品，用以构造某软件程序的菜单接口。它使你能够提供一种方法，在同一个“伞盖”下面执行不同的程序和操作系统命令。

SQL * Graph 这个产品能用图形来描绘数据库信息。

SQL * Report 这个产品用 ORACLE 数据来合并和格式化正文。

SQL * Net 这个产品存取别的计算机上的数据库，这些计算机不在本数据库的所在地。

SQL * Calc ORACLE 是 Easy * SQL 的一个 MS-DOS 版本，只包含 RDBMS 和 SQL * Calc。

“Pro”ORACLE 产品

Pro * ORACLE 产品是程序设计接口产品。它允许程序员使用高级语言来开发应用，而又使用 ORACLE 数据。程序员可以利用预编译程序以便在程序中嵌套 SQL 语句，或者利用一组预定义的例程调用，这些调用使用较低的调用来存取数据库中的数据。支持如下语言：

- Pro * C
- Pro * COBOL
- Pro * FORTRAN
- Pro * PL/I
- Pro * Pascal
- Pro * Ada

并不是所有的 ORACLE 操作系统版本都支持全部高级语言；细节请见针对你的操作系统的“安装和用户指南”。

ORACLE 版本编号方法

由于 ORACLE 产品总在不断开发和改变，所以任何时候产品都会有好几个版本在使用。需要三个号才能完全标别你那儿正在运行的产品的版本：版本号，发行号，修订级号。

版本号：版本号，例如 Version 5，是总的标识。当重要的新功能被加到一个

产品中去时，发行一个新版本。

发行号：发行号指出一个总的版本第几次发行，从 0 开始，例如 Version 4.0 或 Version 5.0。当对现有的程序作了修正或加入新的功能部件时，发行号加 1。

修订级号：修订级指出目标码的具体级别。这个号主要由 ORACLE 公司用来完全地标别一个 ORACLE 系统；一般来说，每个买主只会收到一种版本。在极少数情况下，会收到一个版本和两盘带，这种情况下修订级号不一定是连续的。

在 ORACLE 公司介绍新的产品或改进现有产品时，各个产品的版本号互相之间是无关系的。因此，你可以让一个 V5. 1. 12RDBMS 系统，同 SQL * Forms V2. 0. 3，SQL * Plus V1. 0. 9，以及 Pro * FORTRAN V1. 1. 2 联合运行（此处列举的号，只是为了举例方便而已）。

ORACLE 术语和概念介绍

本节简单介绍一下讨论 ORACLE RDBMS 时常用到的术语。同时也指出在本手册中何处可以找到更详细的说明。

基本术语

发行介质 由 Oracle 公司提供，包含 ORACLE RDBMS 或其它 ORACLE 公司产品的目标代码的磁带（或其它介质，如盒带或软盘）。在针对每个操作系统的安装和用户指南中，有读和加载这些配给介质的指令。

数据库文件 这些文件（又称 DB 文件）包含由一个 ORACLE RDBMS 所存储的或所产生的全部用户数据表和数据字典表。至少有一个 DB 文件。有关数据库文件的更多信息，参见对 INIT.ORA 参数 DATABASE 的说明和“数据库文件”一节。

前映象文件 这个文件（又称 BI 文件）包含数据被改变之前的映象。它被用

来保证，在事务尚未完成、尚未协调一致、用户尚未交付之前，不把该事务弄成永久的。当硬件或系统故障事件发生时，它用来实现向后 (rollback) 恢复。BI 文件还支持多个用户同时存取数据。每个系统只有一个 BI 文件。有关前映象文件的更详尽信息，参见对 INIT.ORA 参数 BEFORE-IMAGE 的说明以及“前映象文件”一节。

后映象文件 这些文件 (又称 AIJ 文件) 用来记录业已交付的事务，在介质故障事件发生时，用来实现向前 (roll forward) 恢复。有关后映象文件的更详尽信息，参见对 INIT.ORA 参数 AFTER-IMAGE 的说明以及“第五章后映象日志”。

数据库管理员 (DBA) 是负责维持一个 ORACLE RDBMS 平稳地运转的人。参见“第 2 章 数据库管理员 (DBA) 的任务”。

IOR 程序 这个程序被 DBA 用来启动 ORACLE 系统 (供用户使用) 或停止一个系统 (对用户关闭)。IOR 还用来初始化系统 (第一次启动它)。有关 IOR 程序的更详尽信息，参见“IOR 命令机器参数”一节。

INIT.ORA 文件 这个文件包含若干系统参数，当用 IOR 程序来启动数据库系统时，要读进和使用这些参数。可以调整这些参数，以便影响 RDBMS 运行的方式或改变某些限定。例如，每当 ORACLE 被启动时，INIT.ORA 文件被读，以便找到 DB 文件和 BI 文件的名字，看看允许多少个进程同时注册到 ORACLE。有关 INIT.ORA 的更多信息，参见“INIT.ORA 参数文件”一节。

SGA 系统全局区域 (SGA) 是主存或虚存中的一个共享存贮区域。用 IOR 程序来分配它，在 ORACLE 运行期间它是

数据活动的中心。SGA 的大小因 INIT.ORA 文件中所设定的参数而异。例如，SGA 包含下列这些：

- 数据缓冲区
- 锁的清单 (lock lists)
- 列缓存 (column caches)
- 表缓存 (Table caches)
- 用户缓存 (User caches)

有关 SGA 的更多信息，参见“SGI 实用程序”一节。

核心 ORACLE RDBMS 的这一部分，执行用户所要求的 SQL 操作。它执行对 DB 和 BI 文件的读写，维护 SGA 里的数据，协调多个并行用户之间的活动。

分离的进程 如果打算由多个用户使用同一个 ORACLE 系统，那么它应当被安装作为一个多用户系统。大多数系统都是多用户的，为了处理数据库数据的并行存取，ORACLE RDBMS 需要一些附加的“用户”，不同的操作系统用不同的术语来称谓它们 (如，VMS 中称之为分离的进程，IBM 的 VM 操作系统中称之为虚拟机)。本指南中，把它们叫做分离的或后台的进程。下面指名的进程，在这里的基本术语介绍之后，还会更详细地介绍的。

ARH 读数据库文件，供用户扫描表之用

BIW 写到前映象文件

BWR 写到数据库和日志文件

CLN 进程失败后清理

共享代码系统 这个术语的含义也因操作系统而异，不过一般说来，当 ORACLE 被安装作为一个共享的系统时，所有 ORACLE 用户和系统，共享着同样的物理目标代码。

共享分区系统 是这样一种 ORACLE 配置：一个单独的数据库 (分区) 和前映象文件可以被多个不相关的“系统” (分离进程组) 所共享。一个在 VAX 群集

上运行的 ORACLE 群集，是一个共享分区系统。

多用户系统 多用户系统是这样的系统：它被设计成可被若干用户同时存取。由于用户们可能需要同时存取同样的表或行，故 ORACLE 采用加锁法来协调多个用户的活动。

单用户系统 一个数据库不需要被多个用户存取，有种种原因一个人仅仅需要他自己的系统。可以用单用户方式或多用户方式来调用 ORACLE RDBMS。如果用单用户方式来调用，那么 RDBMS 的操作就简化了，因为多用户协调工作不需要了。但是，这不一定能在性能上获得好处。一个系统是单用户的还是多用户的，由 INIT.ORA 参数 SINGLE-USER 来设定。

数据字典 数据字典是一组表和视图。由 ORACLE RDBMS 建立并及时更新之。它包含有关用户、表和其它实体的管理信息，有关特权的信息，它是 ORACLE 操作的“心脏”。在“第 7 章 ORACLE 数据字典”中详细介绍。

四个后台进程

提前异步读 (ARH)

ARH 替正在做全表扫描的查询，从数据库文件中把块拷到 SGA，以便减少为了完成查询所需要的时间。概念上，在一个程序需要一块之前，ARH 就已经把这

一块读进 SGA。涉及 ARH 的 INIT.ORA 参数有：READ-REQUESTS, READ-BLKS-TOT 和 READ-BLKS-REQ；不过你最好不要去改它们（见“INIT.ORA 参数文件”一节）。ARH 读块时，与想要处理所检索数据的那个程序的执行是并行的。

前映象写 (BIW)

BIW 是唯一向前映象 (BI) 文件写数据的进程。BIW 把块从 SGA 中的前映象高速缓冲区拷贝到前映象文件。所拷贝的块将用于如下场合：事务回退，CPU 故障，或要求读一致性的查询。

缓冲区写 (BWR)

BWR 是唯一向数据库和后映象 (AIJ) 文件写数据的进程。当缓冲区中的空间被另一块所需要时，BWR 把修改了的块从 SGA 缓冲区中取出，把他们写到数据库中去，如果后映象日志已接通的话，还要把这些块写到 AIJ 文件。

清理 (CLN)

CLN 有两个功能：标识并断开一个已经异常终止的 ORACLE 进程，帮助有秩序地关掉数据库。CLN 周期性地“醒过来”，去扫描 SGA 进程信息。如果它发现进程已经异常终止，它便临时地恢复这个夭折进程的实体，先把它的悬而未决的事务加以回退，然后才代表该进程与 ORACLE 断开。

第二章 数据库管理员 (DBA) 的任务

本章简要地阐述了负责维护 ORACLE 数据库的人员——数据库管理员，即 DBA 的职责和他应关心的事。这里提到两个 ORACLE DBA 用户，即 SYS 和 SYSTEM。

Oracle 公司建议，每一个 ORACLE

装置要有一个数据库管理员 (DBA)，他将是与 Oracle 公司联络的主要人员，将是 ORACLE 数据库用户信息的主要来源。DBA 可以是安装 ORACLE 的人，也可以不是。不过，DBA 往往控制着整个装置。

严格地说，DBA 可以被认为是这样的用户：他（们）对 ORACLE 数据库拥有 DBA 特权。每一个 ORACLE 系统被安装时，有两个这样的用户（SYS 和 SYSTEM）；通过这两个用户中的任何一个，可以把更多的用户加入到 ORACLE 系统。

不严格地说，DBA 这个术语经常被用来指对数据库的概貌和细节都最为了解的人。例如，都有哪一些用户，这些用户贮存和存取什么样的数据，贮存和存取的频繁程度如何，目前有哪几种类型的事务，以及如何最大程度地提高性能等。

在本手册中，DBA 一词一般隐指上述两者，即既拥有 DBA 特权又对概貌和细节最为了解的人。个别地方专指有 DBA 特权的人。

DBA 主要关心的是：

- 保证数据的完整性和一致性
- 加快执行速度和监视性能
- 减少不必要的即冗余的存贮
- 促进用户之间公共数据的共享
- 保证数据库的安全性
- 实施有规律的数据库备份

保证数据的完整性乃是一个广泛的题目，在“第 9 章一致性和并行性”中，以及“第 12 章使用 ORACLE 审查手段”中，都有一些论述。改进性能则是一般性议题，在“第 10 章调整系统以改进性能”中介绍。控制和监视空间的使用，则在“第 8 章数据是怎样被放在数据库中的？”中叙述。

SYS 和 SYSTEM: ORACLE DBA 用户

每一个 ORACLE 系统在安装时自动地有两个拥有 DBA 特权的 ORACLE 用户：SYS 和 SYSTEM。他们的目的有些不同，在下文解释之。

如果你那儿准备让许多用户当 DBA，或者一个 DBA 想要建立表或视图

供 DBA 功能来使用，那么，明智的做法是再建立几个拥有 DBA 特权的用户，这样就不会由于疏忽更改 SYS 或者 SYSTEM 所拥有的表或视图。（参见“拥有 DBA 特权的用户”一节，其中描述了只授予 DBA 用户的特权。）

用户 SYS

ORACLE 用户 SYS 使用口令 CHANGE-ON-INSTALL 来注册；其口令在系统初始化后应立即加以改变。在针对具体操作系统的“安装和用户指南”中，改变 SYS 的口令被当作安装过程的一个组成部分来叙述。其语法是：

```
SQL> GRANT CONNECT TO  
SYS IDENTIFIED BY 口令;
```

由于 SYS 这个帐户拥有全部用于数据字典的基本表，而这些表对 ORACLE 的操作是至关重要的。所以，SYS 这个帐户如果要用的话，也只能偶而用一两次。

SYS 的表中的数据，由 ORACLE 来操纵，作为数据库使用的结果，但决不由用户操纵。除了“人工地更新数据字典”一节中指出的情况之外，任何用户不得以任何方法来更改这些表。任何 ORACLE 用户不得在属于 SYS 的帐户中创建新的表。不遵守这些规定可能导致毁坏整个数据库。

用户 SYSTEM

ORACLE 用户 SYSTEM 注册口令为 MANAGER；象 SYS 一样，其口令在系统初始化后应立即加以改变。

SYS 拥有全部基本的数据字典表，而 SYSTEM 则拥有全部数据字典视图。虽然 SYSTEM 所拥有的东西对 ORACLE 操作的关键程度多少要差些，但仍然强烈地建议你不要去改动它们。

有一些 ORACLE 实用程序创建（并需要）由 SYSTEM 所拥有的表或视图；下面列出这些表或视图（因 ORACLE 版本而异）：

属于 SYSTEM 的表

SQL*Plus Help	CRT Utility	SQL*Forms	SQL*Menu
HELP	CRT	IAPAPP	DMU_APPLICATION
	CRTBOX	IAPBLK	DMU_COMMAND_TYPE
	CRT_PRODUCTS	IAPFLD	DMU_MESSAGE
	CRT_TYPE	IAPMAP	MENU_INFO
	ESC	IAPSQLTXT	MENU_OPTION
	FUNCTIONS	IAPTRG	OPTION_HELP
	GOTO_LC	IAPTRIGGER	PARAMETER_INFO
	LORC		PARAMETER_MENU
			USER_INFO
			WORK_CLASS

我们还建议，用户数据请别存放在 SYSTEM 所拥有的表里。不过 DBA 可以在这个帐户中适当地创建一些数据字典信息的附加的视图。

DBA 特权

用户 SYS 和 SYSTEM 在 ORACLE 系统中具有特殊的特权,任何其它用户,只要是显式地被授予对数据库的 DBA 存取权,也都具有这种特殊的特权。参见“有 DBA 特权的用户”,其中描述了这些特权。

第二部分:使用 DBA 工具

这部分包括几章,每章对应一个 DBA 关心的 ORACLE 程序或实用程序,这些实用程序是:

IOR:用于启动和停止 ORACLE 系统(也描述了一个较小的实用程序—SGI)

ODS:显示系统的运行状态

AIJ:记录数据库的活动,在系统故障时作恢复用

CRT:修改显示终端特征和键

第三章 利用 IOR 程序启动和停止 ORACLE

本章首先介绍 IOR 程序中的各个选择项,讨论 INIT.ORA 文件中的各参数,IOR 程序要读出并使用该文件。

另一个实用程序是 SGI。DBA 利用 SGI 来显示一个特定 INIT.ORA 文件所需要的内存。

ORACLE 系统不是一直都在运行而是每当操作系统引导后才运行。DBA 决定什么时候运行 ORACLE 和什么时候 ORACLE 可用。比如,如果 DBA 需要增加前映象文件的大小,在系统很少使用时,他能停止系统,替换前映象文件或类似的维护。

IOR 程序用来完成 ORACLE 初始化

(安装后的第一次启动)以及启动并停止 ORACLE 系统。

通常 DBA 是 IOR 程序的主要的或唯一的用户。由于 IOR 程序使用不当会导致数据库的损坏,因而将只允许极少的用户访问并运行它。在“安装及用户指南”一书中介绍了如何控制对 IOR 程序的访问。

每当用 IOR 启动或初始化 ORACLE 系统时,IOR 程序读出 INIT.ORA 文件并且根据 INIT.ORA 中设定的参数来配置全局区(SGA)。

IOR 命令和参数

下面是 IOR 命令的语法定义:

```
IOR{WARM|SHUT|CLEAR|INIT[NOCONFIRM]][DBA]
    [PFILE = <filename>][SHAREAD][LIST]
```

图 1 IOR 的语法

以下各参数都可简写为一个字母,例如 IOR [INIT]初始化数据库并启动 ORACLE 后台进程(有时也称冷启动)。

W[ARM]借助于启动后台进程重新启动 ORACLE。

S[HUT]检查活动的数据库用户。当所有的用户都注销后则使 ORACLE 停止。

C[LEAR]不管 ORACLE 用户是否注销都清除他们并立即停止。由于 CLEAR 仅仅结束后台进程,因而用户进程必须在操

作系统级上停止。

PFILE = 文件名。缺省文件名为 INIT.ORA,其中列出用于启动 ORACLE 的参数。

LIST 要求列出用于启动 ORACLE 的参数。此参数可独立使用,此时无其它动作。也可与 I,W,S 或 C 一起使用。

NOCONFIRM 与 INIT 一起使用。此时不会再给出请你进一步确认进行初始化的提示。

DBA 与 INIT 或 WARM 一起使用以限制只允许 DBA 进行注册。(除非再次用 IORW)

SHARED 与 WARM 参数一起使用,以表明是以共享方式启动数据库系统。仅用于共享分区系统。

初始化 ORACLE(IOR INIT)

在第一次启动 ORACLE 时选择初始化选择项,用于初始化数据库文件并启动后台进程。此时数据库文件中的任何用户数据都将丢失;IOR I 总是把数据库文件及 BI 文件复原到原始的空白状态。在大多数操作系统中,应当先用实用程序 CCF 建立 DB 文件和 BI 文件。

在打入 IOR I 之后,出现如下提示:

```
IOR:Connecting to ORACLE V5. 0
.20 Initializ entire database requested
ALL data will be lost -Confirm(YES /
NO)?
```

此时你必须回答大写字母 YES 才能进行初始化。若加上选择项 NOCONFIRM 则抑制此项提示。数据库被初始化后则出现:

DATABASE INITIALIZED

选择项 INITIALIZE 很少使用。通常用在两种情况下,一是 ORACLE 版本更新,另一种是重新配置操作系统文件。如果要对含有用户数据的数据库使用 INITIALIZE 选择项,则应当首先对 DB 文件及 BI 文件卸出和作备份,因为在运行 IOR I 时用户数据将会丢失。

INITIALIZE 把数据库恢复成单分区系统。如果你需要建立其它的分区(比如,对多分区系统进行版本更新),则应当在重装(import)你的数据之前作这件事。

暖启动 ORACLE(IOR WARM)

除了第一次之外,通常都是以暖启动方式启动 ORACLE。暖启动打开 DB 文件

及 BI 文件,为此要求这些文件已联机并可作为启动程序访问。在多用户系统中,它还可以启动后台进程 (ARH,BIW,BWR 和 CLN)。暖启动还把所有夭折的事务都复原(roll back)并抹去所有的临时表。由于 IOR 必须对夭折事务做处理,因而,一旦已用 IOR W 启动,尽管可能耗时间较长,也决不要强行打断它!!

在调用 IOR W 时,IOR 显示 SGA 各成份的尺寸:

- 定长数据字节数
- 变长数据字节数
- 前映象缓冲池个数
- 数据库缓冲池个数

如果出现以下信息也表明是正常的暖启动:

```
n temporary tables deleted
instance n recovered
```

在成功地完成暖启动之后,出现以下信息:

ORACLE WARM STARTED

在出现上述信息之前,若有某个用户企图向 ORACLE 注册,则会看到如下信息:

```
ORACLE initialization in progress
停止 ORACLE(IOR SHUT)
```

停止 ORACLE 运行有两种办法, IORS 或 IOR C。应当尽可能用 IOR S。它拒绝新用户再对 ORACLE 进行注册,但不影响当前还在运行的用户;它要等待所有的当前用户都注销后才停止。此时若有某个新 ORACLE 用户打算注册,则会收到如下信息:

```
ORACLE shutdown in progress
```

此时可保证在停止之前使所有的进行中的事务或圆满完成或复原,其后的暖启动则无需对夭折的事务进行复原。在停止期间可能会显示如下信息:

```
IOR:5 users still active
或是
```

IOR:No active transaction found
完成停止后则出现

IOR:ORACLE shutdown complete

此时后台进程全部停止, DB 及 BI 文件关闭。如果你希望加速停止过程, 则可采用如下的系统中断机制。用 IOR CLEAR。

清理 ORACLE(IOR CLEAR)

这种停止办法不常使用。IOR C 能立即停止而不考虑当前的 ORACLE 用户。因而当前事务的处理全部丢失, 在下次暖启动时, 这些事务全被复原。

IOR 中的 DBA 选择项

在用 IOR 命令启动或初始化数据库时, 若增加选择项 DBA, 则该系统仅允许 DBA 注册。当对数据库进行维护操作, 比如做整个数据库的卸出或重装入时, 应当只允许 DBA 使用该系统。

非 DBA 用户若企图注册, 则会出现以下信息:

ERROR ORA-1035:ORACLE only available to users with DBA privilege.

只有先把 ORACLE 停止, 然后再以不加 DBA 选择项的方式启动之后, 才允许其它用户注册。

IOR 中的 SHARED 选择项

此选择项仅能应用于共享分区系统, 比如在 VAX Clusters 上运行的 ORACLE Cluster 中。在启动时应当为 Cluster 中的每个请求 (instance) 分别指明 SHARED 参数。如果有某个请求启动 ORACLE 时不带 SHARED 参数, 则不允许其它请求再启动 ORACLE, 否则会显示:

the database is in exclusive use by another ORACLE instance.

若 ORACLE 已由一个请求以共享方式启动, 另一个请求又想以非共享方式(未指定 SHARED)启动, 则出现以下信息:

Cannot start exclusively;

in use by another ORACLE instance. 当对一个共享分区数据库进行扩充时, 必须以不带 SHARED 参数的方式运行, 然后使用 ALTFR PARTITION 命令。另外, 当要对全体数据库进行卸出时, 也应在非 SHARED 方式下进行。

INIT.ORA 参数文件

每当 IOR 暖启动(或重新初始化)ORACLE 它都读出一个参数文件, 由其指明 ORACLE 的操作特性。在 IOR 命令行中可指出该参数文件的名称, 其缺省名为 INIT.ORA。在本手册中使用此缺省名。事实上, 你可以另行取名。

INIT.ORA 文件中标明为 ORACLE 所使用的文件, 并且管理直接影响 ORACLE 系统性能的资源。

下面为一个缩短了 INIT.ORA 文件实例:

processes	30
buffers	50
database	oracle.dbs
before-image	oracle.bi
columns	250
tables	50
enqueue	200
users	30

图 2 INIT.ORA 样例文件

形如图 2 的一个 INIT.ORA 文件已写在安装带上。对于大多数的初始 ORACLE 安装其参数值是合适的。在你的系统已经运行, 而且你已对使用 ORACLE 积累一些经验之后, 再考虑改变某些参数。

INIT.ORA 中各参数的发行 (distributed) 值, 可以不同于缺省值。发行值全部在安装与用户指南一书中列出。若在 INIT.ORA 文件中不出现某个参数, 则该参数就取缺省值, 缺省值已被嵌入到程序码中。如果你不采用缺省值, 则需要

INIT.ORA 中明确给出该参数取值。

调整 INIT.ORA 参数

有些 INIT.ORA 参数比如数据库文件或前映象文件其含义很明显。另外一些参数指明界限,比如 PROCESSES 个数是允许并行访问的进程最大个数。另外一些参数影响系统性能。比如,设定 TABLES 则决定表缓存的尺寸,因而确定了在同一时刻允许多少表定义存入内存(在缓存中)则性能将会提高。

图 3 中称之为可变参数其值要影响到 SGA 的尺寸。协调一个系统时通常主要调整这几个参数。某些可变参数影响到系统性能,但不是强制性的边界值;比如,BUFFERS 取高值则改善性能,而低值可使性能下降;但它并不抵制它。某些可变参数还设定能力限度;比如,如果 PROCESSES 是 10,则禁止第 11 个进程注册。

可变参数	能力参数
BUFFERS	TABLE-ACCESSES
COLUMNS	TABLE-HANDLES
TABLES	ENQUEUES
TABLONAMES	OPEN-CURSORS
	PROCESSES
	TRANSACTIONS
	USERS

图 3 可变和能力参数

增大可变参数的取值可以改善系统的性能,但也要增大 SGA 的尺寸。在一个虚存操作系统中,如果 SGA 尺寸过大,会因页入页出而导致性能下降。注意,应当把操作系统中控制虚存工作区的参数设定得与 SGA 的尺寸相符。

事实上取什么样的参数值能使你的系统获益最多取决于几个数据库特性,诸如你的数据是如何安排的,在同一时刻会有多少个用户或程序是活动的。进而取决于你对系统了解的程度以及对这些可变参数是怎样管理,你可试着改变这些可变参数是否使

系统性能获得改善。

在以下的说明中“数目”是指在 SGA 中(进而是在内存常驻)同一时刻某种资源项目的最大个数。当达到此最大值时,则最近最少使用的项目被新的登记项所代替(老的登记项要由磁盘中调用)。此外应注意,INIT.ORA 中的“数目”是全局性的而不是为每个用户,除非另加说明。

例如,在 INIT.ORA 中若设定参数 TABLES 为 50,则 SGA 中可储存 50 个表定义,当访问第 51 个表时,则有一个老的将被第 51 个表所替代。如果有三个用户都在使用同一个表 SCOTT.EMP,为此 TABLES 参数仅需要有一个登记项。如果某个参数取值太低,你就会看到出错信息。

例如:

```
insufficient enqueues
out of processes
insufficient table cache entries
insufficient tablename cache entries
```

如果重复出现这类信息,你可稍等一会儿,待系统较空闲时再次重试。另一作法是使 ORACLE 系统停止,增大 INIT.ORA 中的有关参数,然后再启动 ORACLE。

依赖于操作系统的参数

有许多参数的取值或范围要依赖于操作系统。例如参数 BUFFERS 指明在主存储器中数据缓冲区的数目;但是这种缓冲区的尺寸要依赖于系统。在相应的安装和用户指南中都作了说明。

注意:有许多 INIT.ORA 参数是不可替换的。

INIT.ORA 参数细述

在附录 D 中给出 INIT.ORA 参数小结。

AI_BUFFERS

它指明后映象缓冲区的数目。缺省为 3，通常不作改变。该参数在 ORACLE V5.0 中无效。

AFTER_IMAGE

它指明后映象日志的目录或文件名。其缺省值及装配值都为空值。如果指定为非空，则激活后映象日志。若有多个日志文件则此参数可多次出现。日志文件是按其在 INIT.ORA 文件中出现的次序来使用的。

在 AIJ (后映象日志) 一节中将作更详细介绍。

AI_FILE-SIZE

它指明在 ORACLE 块中日志文件的尺寸。如果设定为零或未作说明，则日志被做成固定尺寸的文件，其名称在参数 AFTER_IMAGE 中给出。若取值为零，则建立名为 SQNnnnnnnn.AIJ 的日志文件，其尺寸是在 AFTER_IMAGE 给出的目录中所给定的 K 字节数。缺省值为 0。

AI_WARN_PCNT

当 AIJ 文件已达到一定百分比时，则在操作员控制台上出现警告信息。缺省值为 0。如果所有的后映象文件都满了或是目录已满，则 ORACLE 将会停止，在重新启动 ORACLE 之前要作备份，并使 AIJ 文件刷新。

AUDIT_TRAIL

这个参数控制审计追踪是否激活。其值为 0 或不出现此参数，则审计不活动。缺省值为 0，但 SQL 中 AUDIT 语句不顾及这个设定值。

BEFORE_IMAGE

指明前映象文件名称。该文件必须已被建立且已为运行 IOR 指定名字 (在大多数操作系统中是 ORACLE CCF 实用程序来建立)。其缺省值与系统有关。

BI_BUFFERS

— 12 —

指定前映象缓冲池个数。其缺省值依赖于操作系统且不应改变。

BI_HIGH

该参数用于共享分区系统，比如使用 VAX Clusters 的 VMS 系统。它指明一个 ORACLE 请求 (instance) 的前映象文件区的结束块。其缺省值为前映象文件的结尾。在安装共享分区时设定该参数，但可由 DBA 改动它。在 VMS 中当运行 CREINIT.COM 时，自动设定该参数 (BI_LOW 表示起始块)。

BI_LOW

该参数用于共享分区系统，比如使用 VAX Clusters 的 VMS 系统。它指明一个 ORACLE 请求的前映象文件区的起始块。其缺省值为前映象文件的起点。在安装共享分区时设定该参数，但 DBA 可以修改它 (BI_HIGH 表示结束块)。

BLOCK_SIZE

它指定数据库块的尺寸。缺省值依赖于操作系统且不应当改动。块尺寸主要有两种，2KB (DEC VAX/VMS, SG AOS/VS 及众多 VNIIX 环境) 或 4KB (IBM VM/SP 或 MVS 环境)。

BUFFERS

该参数规定在主存中高速数据缓冲池的个数。(缓冲池的尺寸与操作系统有关; 在 VMS 中为 2K) 缺省值为 50。一个经过修改的缓冲池在如下情况下强行写入磁盘: 该缓冲池已满或执行了提交 (Commit), 复原 (rollback), 注销 (logoff) 以及字典操作 (DDL)。尽管对其最大值未作限制, 但它的取值直接影响到 SGA 的尺寸。增多缓冲池的个数, 则直接在主存中找到所需数据的机会就增多, 因而使 ORACLE 所要做的 I/O 减少。

BUFF_HASH_BKTS

此参数指定用于找出 ORACLE 缓冲池的杂凑表 (Hash table) 中的项目数、