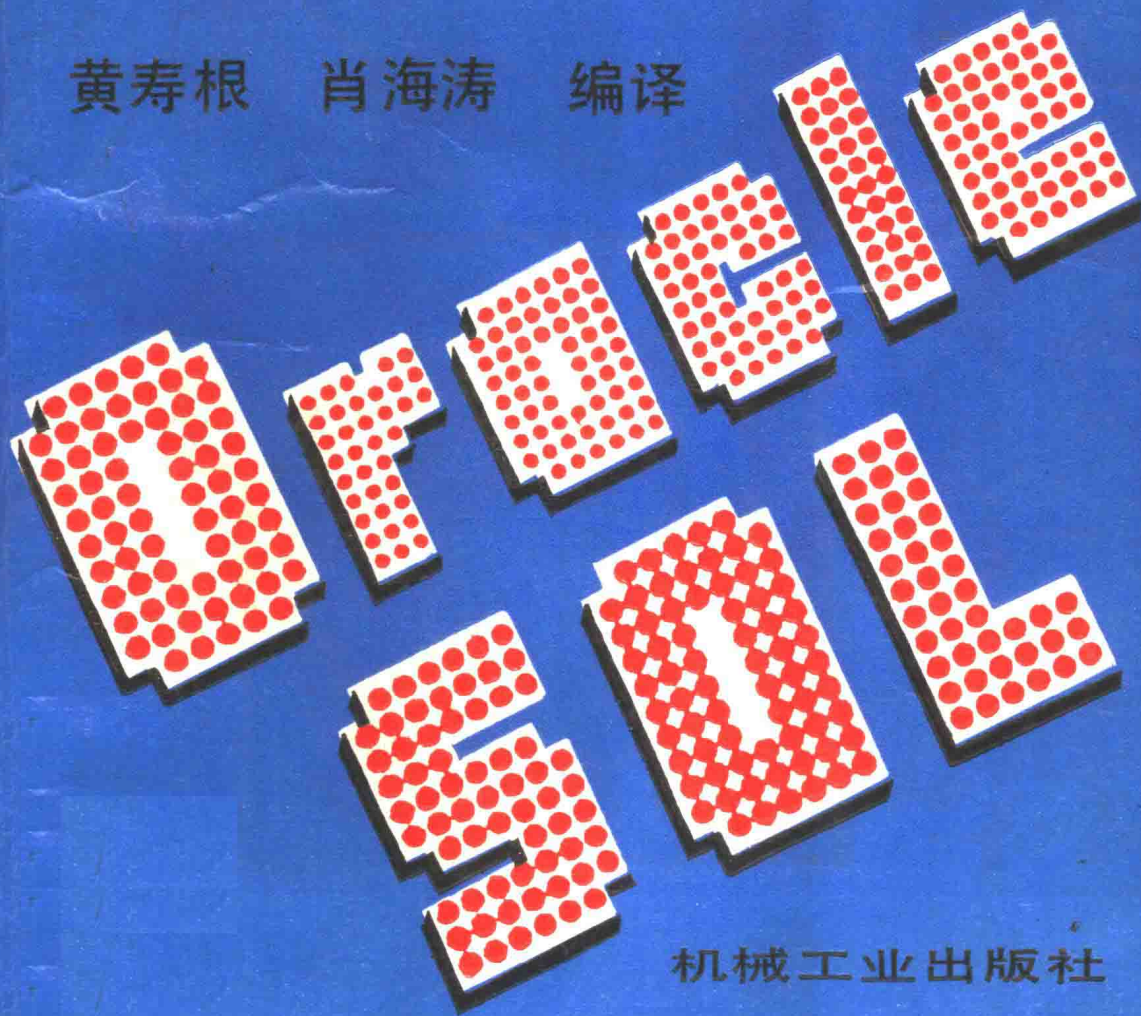


Oracle 数据库管理系统和 SQL 标准数据库语言

黄寿根 肖海涛 编译



机械工业出版社

ORACLE数据库管理系统 和SQL标准数据库语言

黄寿根 肖海涛 编译



机械工业出版社

(京) 新登字054号

内 容 简 介

SQL语言是IBM公司于70年代推出的结构化查询语言, 现已被国际标准化组织以及许多国家采纳为访问关系型数据库管理系统的标准语言。ORACLE关系型数据库在几十种通用计算机系统上均有其相应的运行版本, 在我国的应用也日益广泛。

本书结合作者多年使用SQL语言和ORACLE关系型数据库的经验, 以介绍PC机上运行版本的使用为基础, 系统地介绍了SQL语言以及ORACLE关系型数据库的各种主要软件工具。在其它机型和操作系统上运行的ORACLE软件, 其概念与使用方法均与PC机上相类似, 只不过有些功能键有些变化。

本书适用对象为计算机软件人员、数据库应用方面的人员, 以及大专院校有关计算机专业的教师、学生。

ORACLE数据库管理系统和SQL标准数据库语言

黄寿根 肖海涛 编译

责任编辑: 陈周兰 责任校对: 李秋荣

封面设计: 肖 晴 版式设计: 李慧萍

机械工业出版社出版(北京阜成门外百万庄南里一号)

(北京市书刊出版业营业许可证出字第117号)

海洋出版社印刷厂印刷

机械工业出版社发行·各地新华书店经售

开本 787×1092·1/16 印张 25.625·字数 640 千字

1993年3月北京第一版·1993年3月北京第一次印刷

印数 00,001-03,100·定价: 19.00元

*

ISBN 7-111-03601-2/TP·173

前 言

美国Oracle公司是世界上著名的数据库软件公司之一。十多年前，该公司就推出了Oracle关系型数据库管理系统。至今，已在PC-DOS、VAX/VMS、Unix和IBM VM/CMS等多种世界上广泛运用的操作系统上都有其相应的运行版本，Oracle可以在微型机，小型机，以及中、大型计算机系统上运行。

Oracle系统是以SQL为基础的关系型数据库管理系统。SQL(Structured Query Language)是IBM公司于70年代研制的结构化查询语言。后来，美国国家标准化学会(ANSI)和国际标准化组织(ISO)相继采纳SQL(或称SQL语言)作为访问关系型数据库管理系统的标准语言。在此之前，世界上还没有标准的数据库访问语言。Oracle系统采用了SQL语言。

Oracle数据库管理系统提供了全面的数据库开发、运行和管理等功能。由于它采用了SQL语言，因此与采用SQL标准的关系型数据库管理系统之间有很好的兼容性。用Oracle系统开发的用户数据库有很好的可移植性，它能十分方便地从一个计算机系统移植到另一个计算机系统，因为Oracle系统在微型机到大型机上都有其版本。用Oracle系统开发的应用数据库，随着该单位的发展和业务的增长，其计算机系统也随之扩展，其应用数据库也能十分容易地扩充(包括应用数据库结构的修改和扩充)，因此Oracle系统有很好的可扩展性。由于各种异种计算机上都可配备Oracle系统，这样易于联网，网上各用户数据库之间的信息易于交换和共享。例如，可在微型机上配备用户专用数据库，而在大、中型机上管理大型的用户共享数据库，微机用户可直接访问大、中型机上的共享数据库，也可把微机上的专用数据库中的数据送入共享数据库。这为建立高效的分布式数据库提供了可能。因此，Oracle系统具有良好的可联结性。

Oracle关系型数据库管理系统由多个功能模块构成，为用户提供了多种多样的访问数据库的接口，用户可按自己的爱好选用。其中，如SQL*Forms等，具有第四代语言的特点，为开发一个有友好的用户界面的用户数据库提供了高效的工具。Oracle系统配有多种高级程序语言接口，也为应用开发工作提供了极大的方便。Oracle系统采用了聚簇(Cluster)、优化查询以及数据库安全机制等技术，提高了关系型数据库的运行速度和安全性，给用户带来了极大的利益。

十多年来，Oracle系统已成为世界上广泛运用的数据库管理系统，并可望在90年代得到进一步的发展。Oracle公司也将继续提高、改进和扩展其数据库软件产品的功能。Oracle数据库管理系统也已在我国的不少计算机系统上运行。毫无疑问，今后10年，我国也将跟随世界的潮流，广泛地运用Oracle系统来开发用户的管理信息系统(MIS)。因此，运用Oracle系统来开发用户数据库的前景是美好的。

本书由浅入深地介绍了Oracle关系型数据库管理系统的功能，同时介绍了标准数据库语言SQL。本书列举了大量的例子，通过它们告诉读者如何运用Oracle系统提供的功

能去设计、开发、运行和管理用户数据库。本书对不同计算机系统上的Oracle用户和潜在的Oracle用户都有参考价值，并为了解和学习标准的数据库语言SQL提供了机会。

本书第一章概要地介绍了Oracle的功能特点和它的组成部分，读者从中可了解Oracle关系型数据库管理系统的全貌，尤其是了解Oracle为用户提供的多个访问数据库的接口。第二章介绍了结构化查询语言SQL的基本语法和组成；介绍了SQL*Plus这个Oracle系统的最基本的用户接口；还介绍了用户如何启动和进入Oracle系统以及在Oracle系统中进行一些基本的操作。第三章通过例子告诉读者如何去设计、创建用户数据库。第四章通过大量例子，全面地介绍了运用SQL命令和SQL*Plus命令查询数据库的各种灵活而有效的方法，可满足用户多种多样的需要。第五章介绍了视图功能。第六章较全面地介绍了Oracle系统中所用到的操作符和函数，为读者选用它们提供了方便。第七章介绍了第四代语言SQL*FORMS，告诉读者如何用它设计和运用一个与用户十分友好、方便的数据库接口。第八、九章介绍了另外两个数据库接口SQL*Report Writer和SQL*Menu，它们具有各自的特点和用途。第十章较全面地介绍了Oracle系统的安全机制，用户可运用它们所提供的功能为用户数据库服务。第十一章介绍了索引和聚簇技术，这为用户优化用户数据库的运行性能提供了有效的工具。第十二章介绍了Oracle系统与高级程序语言的接口Pro*C，读者通过对Pro*C的学习，不但可把对Oracle用户数据库的操作嵌入面向过程的C语言程序中，也便于触类旁通地掌握Pro*系列的其它高级程序语言接口的使用规律。第十三章介绍了Oracle的输入输出实用程序，以用于两个Oracle用户数据库之间，以及数据库和外部文件之间的数据交换。第十四章介绍了SQL*CALC接口，使已习惯于运用电子表格的用户使用SQL*CALC来访问Oracle用户数据库。第十五章介绍了Professional Oracle在微型计算机上的安装方法。

本书由黄寿根组织编译。其中第一章至第六章、第十五章由黄寿根编译，第七章至第十四章由肖海涛编译。虽然编译者在使用Oracle数据库管理系统多年的基础上编译了本书，但由于水平有限，经验还少，难免有许多错误和不当之处，敬请读者指正。

编译者

1991年于上海

目 录

第1章 ORACLE概述	1
1 关系型数据库管理系统的功能.....	1
1.1 数据库管理员.....	2
1.2 最终用户.....	2
1.3 系统开发人员.....	2
2 ORACLE怎样组织数据.....	3
3 ORACLE怎样访问数据.....	4
4 ORACLE环境.....	5
4.1 数据库管理工具.....	5
4.2 数据访问和处理工具.....	6
4.3 编程工具.....	8
4.4 网络连接工具.....	8
5 为什么选择ORACLE.....	8
5.1 ORACLE给予用户安全性.....	9
5.2 ORACLE具有竞争性.....	9
5.3 ORACLE支持应用开发.....	9
5.4 ORACLE使用SQL命令.....	10
6 提要.....	10
第2章 SQL命令和SQL * PLUS命令	11
1 ORACLE执行程序.....	12
2 通过SQL * PLUS进入ORACLE.....	12
2.1 修改“超级”用户的口令.....	13
2.2 赋与用户特权.....	13
2.3 删除用户和修改特权.....	15
3 SQL简介.....	16
3.1 SQL语句的类型.....	16
3.2 SQL语句剖析.....	18
3.3 SQL * PLUS联机求助命令 HELP.....	19
3.4 纠正当前命令行的错误.....	20
3.5 SQL * PLUS的数据类型.....	20
4 基本的SQL命令以及操作.....	22
4.1 创建数据库表.....	22
4.2 在表中插入数据.....	24

4.3	用SELECT命令查询数据库	24
4.4	从表中删除行	29
4.5	更新行中的数据	30
4.6	在表中增加列和修改列	31
4.7	删除表	33
5	数据库的安全性——创建视图	33
6	SQL*PLUS命令	35
6.1	暂停组合键	35
6.2	存贮和检索SQL命令	35
6.3	编辑SQL命令和SQL*PLUS命令	36
6.4	捕捉文件供打印输出	37
6.5	执行宿主计算机操作系统的命令	37
6.6	在不同的用户名下进入SQL*PLUS	37
7	退出ORACLE	38
7.1	退出SQL*PLUS	38
7.2	阻止新用户进入ORACLE, 停止ORACLE运行	38
7.3	从主存贮器撤走ORACLE	39
8	提要	39
第3章	创建数据库	40
1	设计关系型数据库	40
1.1	定义数据库	40
1.2	数据库举例	42
2	表的组成部分	44
2.1	表的命名	44
2.2	数据类型	45
2.3	显示格式	47
3	在SQL*PLUS中创建表	48
3.1	SPACE选择项	49
3.2	CLUSTER选择项	49
4	选用另一些表中的列来创建表	49
5	创立同义词	50
6	建立表结构信息的文档	50
6.1	表结构信息的获得	50
6.2	添加注释	51
6.3	数据字典表的运用	52
7	变更表的结构	53
7.1	重新命名表	53
7.2	变更表结构	53
7.3	删除一个表	54

8	处理表中的数据	54
8.1	在表中插入数据	54
9	输入数据的简便工具SQL*FORMS	55
9.1	表格	58
9.2	块	59
10	提交、退回和自动提交	64
11	使用UPDATE命令更新数据	65
12	提要	66
第4章	查询数据库	63
1	查询单个表	63
1.1	投影操作	69
1.2	选择操作	72
2	排序操作	74
3	分组的总操作	77
3.1	GROUP BY 选择项	77
3.2	HAVING选择项	78
3.3	BREAK 命令	81
3.4	COMPUTE 命令	82
4	同时访问多个表的信息: 联结操作	84
4.1	外联结	86
5	子查询	87
5.1	相关子查询	88
6	查询表达操作符	89
6.1	树型结构信息的查询	92
7	编辑命令和命令缓冲区	94
7.1	SQL*Plus命令的存贮	96
8	提要	96
第5章	视图的定义和运用	97
1	视图的作用	97
2	工程管理应用实例	99
3	定义和查询单表视图	100
3.1	定义视图	101
3.2	视图访问特权	104
3.3	查询视图	105
3.4	创立同义词简化视图操作	106
4	定义和查询复杂视图	106
4.1	不用联结操作定义源于两个基表的视图	107
4.2	用联结操作定义源于两个基表的视图	108
4.3	三基表视图	110

4.4	格式命令COLUMN	112
5	基表结构变化后视图一致性的维护	113
6	通过视图插入、更新和删除数据库的数据	115
6.1	单表视图	116
6.2	单表视图中的例外	120
6.3	多表视图	121
7	删除视图	121
8	数据字典中的系统视图	122
9	提要	122
第6章	操作符和函数	123
1	操作符	123
1.1	逻辑操作符	123
1.2	值操作符	130
1.3	语法操作符	131
1.4	查询表达式操作符	133
2	函数	133
2.1	字符函数	133
2.2	算术函数	139
2.3	日期和时间函数	141
2.4	组函数	144
2.5	其它函数	145
2.6	变换函数	148
3	格式	149
3.1	字符格式	149
3.2	数字格式	149
3.3	日期格式	150
4	提要	152
第7章	SQL*FORMS	154
1	SQL*FORMS, 在应用开发方面的一个新概念	154
1.1	SQL*FORMS的功能	155
2	进入SQL*FORMS	155
2.1	学习使用功能键	156
2.2	表格层	157
2.3	块层	158
3	进行全屏幕编辑	160
3.1	设计一个复杂表格	161
3.2	屏幕编辑的使用	161
3.3	域的修改	164
3.4	块的修改	169

3.5 表格运行可选项	169
4 运用触发器编程	171
4.1 触发器类型	171
4.2 选择触发器窗口	176
4.3 触发器步骤有效语句	179
5 生成表格	182
6 运行表格	184
6.1 RUN命令的语法	184
6.2 运行表格时的可选项	184
6.3 键盘功能	187
6.4 查询	187
7 表格的文档	188
8 提要	189
第8章 SQL*REPORT WRITER	190
1 报表类型	190
2 进入SQL*REPORT WRITER	191
3 主菜单	191
4 在菜单系统中操作	192
5 帮助功能	192
6 操作	193
7 输入查询	196
7.1 输入SELECT语句	197
7.2 父—子关系	199
8 编排报表的格式	200
8.1 修改域的设置	200
8.2 通过计算得出的域	203
9 检查组的设置	205
9.1 加入总计值	208
9.2 修改文本信息的设置	211
10 报表菜单	215
11 参数的设置	216
12 复查报表	217
13 运行报表	217
13.1 在命令行中设置参数	218
14 存放报表定义的数据库表	218
15 提要	219
第9章 SQL*MENU	220
1 如何理解SQL*MENU	220
1.1 调用SQL*MENU	221

1.2	SQL * MENU的开发用主…页鉴	223
2	管理菜单的保密性	224
3	建立用户菜单	227
3.1	更新应用信息	228
3.2	建立顶层菜单	228
3.3	帮助文本菜单	234
3.4	替代参数菜单	234
3.5	生成菜单	234
4	库和应用的管理	236
5	用户菜单文档的建立	237
6	提要	238
第10章	数据库的安全机制	239
1	雇员数据库	239
2	系统提供的保护	239
3	鉴别子系统	239
3.1	系统级和库体级特权	241
4	建立用户系统级特权	241
4.1	改变“超级用户”口令	241
4.2	建立一个数据库管理员用户	242
4.3	建立非DBA的用户	242
4.4	建立带有RESOURCE特权用户	243
4.5	列出用户的特权	243
4.6	收回系统级特权	245
5	建立用户库体级特权	245
5.1	赋与库体特权	246
5.2	收回库体特权	252
6	对表的子集(视图)运用特权	253
6.1	对列访问的限制	253
6.2	对行访问的限制	255
6.3	建立多基表视图	256
6.4	列出视图定义	257
7	审计	258
7.1	启动审计	258
7.2	审计系统的访问	260
7.3	审计表和视图	263
8	提要	264
第11章	优化系统性能	266
1	索引	266
1.1	有选择地使用索引	266

1.2	影响性能最显著的索引	267
1.3	建立索引	267
1.4	列出索引	271
1.5	运用索引的SELECT语句	271
1.6	检验索引	272
1.7	删除索引	272
1.8	测量特性: TIMING	274
2	聚簇	274
2.1	建立聚簇	275
2.2	建立聚簇表	278
2.3	删除聚簇	280
3	提要	282
第12章 ORACLE与高级语言C的接口		283
1	新颖而高效的预编译接口	283
2	选择语言	283
3	使用Pro*C程序的场合	284
4	Pro*C组成部分	284
4.1	C语言基础知识	284
4.2	PRO*C程序举例	286
4.3	说明变量	288
4.4	把Oracle的功能纳入程序	290
4.5	连接数据库	290
4.6	执行语句	290
4.7	出错处理和状态变化	299
4.8	提交、退回和退出	301
5	编译过程	302
5.1	预编译C程序	302
5.2	编译生成目标文件	305
5.3	连接	305
5.4	用户出口程序	305
6	提要	306
第13章 ORACLE实用程序: EXP、IMP、ODL和SQL*LOADER		307
1	输出和输入Oracle数据库信息	307
1.1	输出Oracle数据库信息	307
1.2	输出操作举例	310
1.3	输入Oracle数据库信息	314
1.4	输入实用程序运行举例	317
2	从外部文件中装入数据: ODL	319
2.1	ODL的功能	321

2.2	ODL概述	321
2.3	调用ODL命令行	322
2.4	ODL命令行参数可选项	322
2.5	ODL控制文件	323
2.6	由ODL产生的输出文件: LOG和BAD	326
2.7	ODL出错信息	329
3	从外部文件中装入数据: SQL*LOADER	330
3.1	装载PROPERTY表	331
3.2	装入统计数据文件	335
4	提要	339
第14章 SQL*CALC 的使用		340
1	起始运行	341
2	标准的电子表格的特性	343
2.1	输入文字和数字	343
2.2	输入重复的文字	344
2.3	执行拷贝命令	344
2.4	得到帮助信息	345
2.5	调整显示值: LAYOUT	345
2.6	保存电子表格	346
2.7	打印电子表格	348
3	查询数据库	349
3.1	进入ORACLE	350
3.2	SQL*CALC的SELECT语句	352
3.3	从PROPERTY数据库中查询销售信息	353
3.4	自动上溢和下溢选择	357
3.5	利用WHERE短语查询记录	358
3.6	拷贝SQL SELECT语句给其它单元	360
3.7	保持最新值: 重新执行查询	363
4	建立新的数据库表和插入数据	364
5	更新数据库记录	366
6	其它数据库操作	367
7	退出SQL*Calc	368
8	提要	368
第15章 Professional Oracle 的安装		369
1	Oracle软件包包含些什么	369
2	运行Oracle所需资源	369
3	安装前的准备工作	369
3.1	备份Oracle软件介质	369
3.2	修改CONFIG, SYS和AUTOEXEC, BAT文件	370

4	安装 ORACLE	371
4.1	安装的初始化工作	372
4.2	安装数据库管理系统	375
4.3	安装 SQL*Plus	376
4.4	安装实用程序	378
5	提要	384
附录A	Oracle的保留字	385
附录B	数据字典中的视图	386
附录C	SQL命令和SQL*Plus命令	386
附录D	SQL*FORMS功能键	393

第1章 ORACLE概述

Oracle 关系型数据库管理系统提供了一个完善的软件环境，它为用户运用它开发用户应用数据库系统提供了有效的手段和良好的工具。安装在不同计算机系统上的Oracle的基本功能是一样的，用户运用它开发应用数据库的方法也基本相同。一般说来，在介绍Oracle关系型数据库管理系统时，总是以某计算机系统上的Oracle为例，这并不妨碍学习和运用其它计算机系统上的Oracle。本书就以在PC-DOS、MS-DOS或OS/2操作系统上运行的关系型数据库管理系统，即称之为Professional Oracle的为例来介绍Oracle关系型数据库管理系统的。Professional Oracle与其它计算机系统上的Oracle十分相似，用户用它开发的应用数据库有良好的移植性。为简便起见，在本书的以下章节，简称Professional Oracle关系型数据库管理系统为Oracle，或Oracle系统。仅在个别地方，认为有必要说明时，才明确指明Professional Oracle。

一般说来，不是非得精通了Oracle才能运用Oracle。但是，对Oracle了解越深，则运用Oracle所提供的功能越自如。这一章向读者简要地介绍Oracle的功能，看看它能为用户做些什么事情。

1 关系型数据库管理系统的功能

一个现代的关系型数据库管理系统，简称RDBMS，有着广泛的用途，能完成多种工作。总体上说来，它是作为数据的逻辑描述与物理存贮之间的一个透明接口。实际上，它提供一系列灵活而完善的功能和手段，来处理信息。一般说来，你可以用它做下列事情：

- 定义数据库
- 查询数据库
- 编辑、增加和删除数据库中的数据
- 修改数据库的结构
- 当多用户访问数据库时，为数据提供保护
- 在网络中实现通讯
- 卸出和装入数据，实现数据库和另一个系统之间的数据交换

由于RDBMS给予用户强大的处理数据的能力，因此它可以作为管理信息系统(MIS)方面应用的基础。

上述功能中哪一个对用户更为重要些，取决于用户要解决什么样的实际问题。你可能负责创建和维护一个数据库；你也可能是个一般用户，要在一个已经存在的数据库上完成某项具体工作；或者，你是一个应用系统开发人员，在某个阶段内，每个人各自运用上述RDBMS提供的某几项功能。如果你使用版本为5.1或更高的Professional Oracle，则你一个人同时扮演了上述三个角色。下面稍为详细地介绍这三方面的工作。

1.1 数据库管理员 (Database Administrator)

为避免一个复杂的系统发生混乱，需要有专人为其负责。对数据库负责任的人称为数据库管理员（简称DBA）。数据库管理员拥有一组特权，以便使他完成对数据库的控制和管理。任何具有一个或一个以上用户的数据库系统都应该有一位数据库管理员。对于一个单用户系统，例如Professional Oracle来说，仅有的一个用户，一人扮演多个角色，有时他作为系统的数据库管理员；有时他是位拥有有限数据库特权的本来用户意义上的一般用户。

数据库管理员的工作职责包括以下几方面：

- 创建基本的数据库存储结构
- 修改数据库的结构
- 备份和重新恢复数据库
- 监视和维护数据库中数据的一致性
- 在数据库和外部文件之间传送数据
- 监视和控制用户对数据库的访问
- 安排数据库的物理存储

对于Oracle系统以及对于DBA而言，上述任务的复杂性决定了需要一个中央文档系统（Central Documentation System），组合到任何一个关系型数据库管理系统中去。在下面可以看到，这一章将要讨论到的Oracle数据字典（Data Dictionary），它就是这样一个中央文档系统。

1.2 最终用户

最终用户指用户数据库的使用者，他们的工作职责要简单些，但也十分重要。他们必须能够：

- 查询数据库
- 产生和打印输出报表
- 与其它用户分享数据或维护数据的安全
- 运用常规的方法更新数据库
- 从一般用户的观点定义数据库的应用

1.3 系统开发员

系统开发员开发从单用户到复杂的多用户的数据库应用系统，或称用户数据库。开发工作的难度取决于眼前所要解决的实际问题的复杂程度。系统开发员必须做到：

- 提供直观的屏幕系统，让一般用户能用它方便地访问数据库
- 让用户在屏幕上编辑和控制他们的输入信息
- 按用户的不同需要，为他们提供数据，或对他们保密某些数据
- 在行和列一级上给数据加锁
- 为用户提供运用条件逻辑来访问和处理数据的功能
- 为用户提供运用传统的过程语言来访问数据库的功能
- 为用户提供访问计算机外围设备的功能

- 提供其它软件包与数据库之间传递数据的接口
- 当用户数据库系统扩展时,能把原数据库方便地移植到其它大中型计算机上的功能

一个系统开发人员本身不一定是个程序员。由于现代关系型数据库管理系统提供的功能大大提高了系统开发的效率,正如电子表格软件把会计的劳动密集型的财务交叉平衡工作转化为高水平的财务分析一样,现代关系型数据库管理系统能省去大量重复性的、费时的程序编制工作。过去,大量的时间花费在如何输入、更新和显示输出数据的程序编制上,而现在则可使系统开发人员把大量时间化费在如何设计更完善和更有效的管理信息系统上。

象第四代语言(4GLs)这样的新型软件工具一样,允许软件开发者集中注意力在数据的逻辑意义上,而不是它们的具体存贮位置上。非过程性的表格软件和数据处理语言,使以更少的时间开发出更高效的应用系统成为可能。这也是区分商品软件市场上各种关系型数据库管理系统性能优劣的依据。软件产品提供的功能越先进,用户数据库开发者的工作变得越简单。

为用户的应用系统选择合适的开发工具是件困难的事,但选得好,将使用户的应用系统的开发进展顺利。当前有一件事是肯定的,一个成功的应用系统必须远远超过传统的系统。传统的系统缺乏适应性,它往往在一个固定的结构上简单地输入、更新、显示和打印输出一个数据文件。现代的目标是一个适应性强的系统,它使信息管理以表格兼容的方式满足人们的需求。

开发一个应用系统,使它随着新的需求而能方便迅速地修改它也是件重要的事情。要改造一个过时的系统的代价是昂贵的。因此,把数据存贮在一个容易修改的结构中,使用适应性强的软件工具来处理数据,一旦需要改造这个系统时,所费的代价也就小得多了。

另外,不管当前的应用系统所需的应用范围是多么有限,但一定要预先考虑到它的可扩充性。例如,对于现在来讲,一个单用户的商业方面的应用系统已成为过去的事情了。而今,已发展为多个用户都要去访问的一个数据库。因此,当前的应用系统应事先考虑多个用户访问它时的安全性,还应有清晰的硬件扩充方案。

一个现代关系型数据库管理系统必须能满足上述这些需求。Oracle关系数据库管理系统给用户提供了—套完善的数据管理、应用开发的工具,以及给最终用户提供理想的工作环境。版本5.1以上的Oracle系统,集合了许多软件功能模块,给用户提供了多种方法和手段,用户可按个人的意愿和需要使用该系统。

下面在向用户介绍Oracle的功能以及它的组成部分以前,先介绍一下关于Oracle是怎样组织、安排和访问数据的基础知识。

2 ORACLE怎样组织数据

在Oracle中,所有的数据都存贮在表(Table)中。一个表由一些列(Columns),又称域(Fields)和行(Rows)组成。从一个表中,用户可以选择一些列和行组成另一个表在屏幕上显示出来,或打印输出。图1-1表示了一个表的基本结构。

视图(View)来源于表,创建视图仅为实现某些目的。视图看起来象一个“真正”的表,但在数据库系统里,视图仅仅作为一个定义存贮在数据库里的。由于这个原因,视图被称之为虚表(Virtual Table)。导出该视图的源表称之为基表(Base Table)。