

深入浅出

SQL Server

数据库开发

龙 帅 / 编著

- ◆ 由国内一线 SQL Server 教师精心编著，融会作者多年的实际编程经验与开发技巧
- ◆ 每个知识点都辅以实例进行讲解，综合实例更能让您快速掌握 SQL Server 的精髓
- ◆ 大量的程序代码注释，有的放矢地讲解利用 SQL Server 开发动态数据库的全过程
- ◆ 本书内容全面、系统，知识讲解深入浅出，堪称 SQL Server 数据库设计的权威教材



随书光盘内含书中所讲范例的完整源代码，课后上机操作题答案及源代码和教学用 PowerPoint 演示课件



中国青年出版社
中国青年电子出版社

<http://www.21books.com> <http://www.cgchina.com>

TP311.138
304D

深入浅出

SQL Server

数据库开发

龙 帅 / 编著



中国青年出版社
中国青年电子出版社

<http://www.21books.com> <http://www.cgchina.com>

本书由中国青年出版社独家出版。未经出版者书面许可，任何单位和个人不得以任何形式复制或传播本书的部分或全部内容。

图书在版编目(CIP)数据

深入浅出 SQL Server 数据库开发 / 龙帅编著. —北京：中国青年出版社，2006

ISBN 7-5006-7035-4

I.深... II.龙... III.关系数据库—数据库管系统，SQL Server 2000 IV. TP311.138

中国版本图书馆 CIP 数据核字（2006）第 089253 号

书 名：深入浅出 SQL Server 数据库开发

编 著：龙 帅

出版发行：中国青年出版社

地址：北京市东四十二条21号 邮政编码：100708

电话：(010) 84015588 传真：(010) 64053266

印 刷：山东高唐印刷有限责任公司

开 本：787 × 1092 1/16 印 张：23.75

版 次：2006年9月北京第1版

印 次：2006年9月第1次印刷

书 号：ISBN 7-5006-7035-4

定 价：36.00元（附赠1CD）

前 言

目前, Internet 凭借着它通过上百个国家数千万个网站向全世界的人们提供信息资源的实力, 成为世界上最大的数据库资料库。它的迅猛发展, 也在一定程度上使得数据库技术在网络上的应用越来越受到人们的重视。

Microsoft 公司推出的 SQL Server 数据库管理系统是大型关系数据库管理系统中的佼佼者, 它建立在成熟而强大的关系模型基础上, 可以很好地支持客户/服务器模型, 能够满足各种类型的企业客户对软件供应商构建商业应用程序的需要, 并且在易用性、可伸缩性、可靠性以及数据仓库等方面进行了显著的改进和提高。

SQL Server 2000 同之前版本相比, 其新特性主要表现在以下几点: 支持扩展标记语言 (XML); 强大的、灵活的基于 Web 的分析; 增强的可伸缩性和可靠性; 快速的市场引入时间; 支持 OLEDB 和多种查询; 分布式数据库分布视图。

本书以实例为主线, 运用大量的图片和丰富的实例, 由浅入深、真实地再现了作者指引读者学习 SQL Server 的全过程。全书共分为 17 章, 其内容包括数据库系统的基础知识; SQL Server 的安装及设置; 数据库和表的基本操作; 基本的 T-SQL 语言语法; 索引的创建、管理及使用; 视图的创建、修改及删除; 数据表的高级查询; 身份验证模式的相关内容; 角色和权限的创建、设置及更改; 数据的备份和还原; SQL 的各项代理服务; 数据完整性; 事务的还原和锁定; 各种不同的复制类型; 组件服务的相关内容; 如何在其他环境中使用 SQL Server。本书的最后一章详细介绍了如何以 SQL Server 为基础, 搭建一个完整的售后管理系统, 在理论讲解的同时增加了对读者实际动手能力的培养。

为方便广大教师进行教学以及学生进行自学, 本书在每章后均配有精心设计的填空、选择及上机操作等各种形式的课后练习题。这些练习题对读者巩固各章的知识, 增加实际操作经验将起着非常重要的作用, 而这些练习题的答案均以附录的形式在书后提供。同时, 考虑到实际教学以及参考学习的需要, 在本书配套光盘中还收录了全书实例以及上机操作题中所涉及到的全部程序代码, 方便读者在学习过程中进行对照, 从而可以更好地加深理解各个知识点的相关内容。

本书适合作为初、中级读者学习 SQL Server 的入门图书, 而丰富的实例与各种技巧的融合也适合对 SQL Server 有一定了解的读者将本书作为参考图书深入学习。同时, 本书也可作为大中专院校、高职高专、社会培训班的计算机教材使用。

由于时间仓促和水平有限, 书中不足之处在所难免, 欢迎广大读者批评指正。

作 者

2006 年 9 月

目 录

第 1 章 数据库系统介绍

1.1 数据库概述	1
1.1.1 数据库系统的历史	1
1.1.2 数据库系统	2
1.1.3 数据库	3
1.1.4 数据的组织结构	4
1.1.5 分布式数据库	5
1.2 数据模型	6
1.2.1 数据模型的构成	6
1.2.2 数据模型的类型	6
1.3 基本范式	9
1.3.1 第一范式 (1NF)	10
1.3.2 第二范式 (2NF)	10
1.3.3 第三范式 (3NF)	11
1.4 E-R 模型	13
1.4.1 实体	13
1.4.2 属性	13
1.4.3 关系	14
1.4.4 关系的度	15
1.4.5 E-R 图	15
1.5 小结	16
1.6 习题	16

第 2 章 SQL Server 2000 概述

2.1 SQL Server 2000 简介	17
2.1.1 SQL Server 2000 的特点	18
2.1.2 SQL Server 2000 的系统需求	18
2.2 SQL Server 2000 的安装	20
2.3 SQL Server 2000 的管理程序	24
2.3.1 服务管理器	24
2.3.2 企业管理器	26
2.3.3 查询分析器	27

2.4 注册 SQL Server 服务器	28
2.4.1 用注册向导注册服务器	28
2.4.2 用企业管理器注册服务器	32
2.5 网络设置	33
2.6 小结	34
2.7 习题	34

第 3 章 数据库和表的基本操作

3.1 数据库的基本操作	36
3.1.1 数据库文件	36
3.1.2 数据库文件组	37
3.1.3 创建数据库	37
3.1.4 查看和删除数据库	43
3.2 表的基本操作	44
3.2.1 表的创建	44
3.2.2 表的删除	46
3.2.3 表的基本查询	48
3.2.4 记录的插入	51
3.2.5 记录的修改	53
3.2.6 记录的删除	55
3.3 实例精讲	56
3.3.1 入门: 建立公司数据库	56
3.3.2 进阶: 创建人力资源表	58
3.3.3 熟练: 对表进行各项操作	59
3.4 小结	63
3.5 习题	63

第 4 章 T-SQL 简介

4.1 SQL 语言简介	65
4.2 T-SQL 语言简介	66
4.2.1 T-SQL 语言的发展	66
4.2.2 T-SQL 的数据类型	67
4.3 T-SQL 的结构化程序设计	71

4.3.1 条件语句	71	5.5.3 全文索引的使用	108
4.3.2 转移语句	72	5.6 实例精讲	108
4.3.3 循环语句	73	5.6.1 入门: 建立一个简单索引	110
4.4 存储过程	75	5.6.2 提高: 查看索引	110
4.4.1 存储过程的概念	75	5.6.3 熟练: 建立复杂索引	111
4.4.2 建立和删除存储过程	75	5.7 小结	112
4.4.3 在存储过程中使用参数	77	5.8 习题	113
4.4.4 使用 RETURN 语句	78	第 6 章 视图	
4.4.5 使用 sp_helptext 语句存储过程	80	6.1 视图的相关概念	114
4.5 实例精讲	81	6.1.1 视图	114
4.5.1 入门: 使用变量	81	6.1.2 视图的类型图	115
4.5.2 提高: 使用结构化编程	82	6.1.3 视图的限制	115
4.5.3 熟练: 使用存储过程	83	6.1.4 视图的优点	116
4.6 小结	86	6.2 视图的创建	116
4.7 习题	87	6.2.1 使用 T-SQL 语言创建视图	116
第 5 章 索引		6.2.2 使用企业管理器创建视图	119
5.1 索引概述	88	6.2.3 使用创建视图向导创建视图	123
5.1.1 数据的查询过程	88	6.3 视图的修改和删除	125
5.1.2 索引的引入	89	6.3.1 视图的修改	125
5.1.3 索引的结构	90	6.3.2 视图的删除	125
5.2 索引的分类	91	6.3.3 视图的修改和删除的 T-SQL 语句	126
5.2.1 简单索引和复杂索引	91	6.4 SQL Server 中的视图	126
5.2.2 唯一索引和非唯一索引	92	6.4.1 分区视图	126
5.2.3 聚集索引和非聚集索引	93	6.4.2 分区视图的实现	127
5.2.4 全文索引	95	6.4.3 索引视图	128
5.3 索引的创建	95	6.4.4 SQL Server 中的查询优化	130
5.3.1 用向导创建索引	95	6.5 实例	131
5.3.2 用企业管理器创建索引	97	6.5.1 建立数据库	131
5.3.3 用 T-SQL 语句创建索引	99	6.5.2 建立视图	132
5.3.4 全文索引的创建	101	6.6 小结	133
5.4 索引的管理	104	6.7 习题	133
5.4.1 用企业管理器管理索引	105	第 7 章 表的高级查询	
5.4.2 用 T-SQL 语句管理索引属性	106	7.1 聚合函数	135
5.5 索引的使用	106	7.1.1 AVG 函数	136
5.5.1 索引使用分析	106	7.1.2 BINARY_CHECKSUM 函数	137
5.5.2 一般索引的使用	107	7.1.3 CHECKSUM 函数	137

7.1.4 COUNT 函数	138	8.5 小结	170
7.1.5 COUNT_BIG 函数	139	8.6 习题	171
7.1.6 GROUPING 函数	140	第 9 章 角色和权限	
7.1.7 MAX 函数	141	9.1 权限	172
7.1.8 MIN 函数	141	9.1.1 权限概述	172
7.1.9 SUM 函数	142	9.1.2 对象权限的设置	173
7.1.10 STDEV 函数	143	9.1.3 语句权限的设置	176
7.1.11 STDEVP 函数	143	9.2 角色	178
7.1.12 VAR 函数	144	9.2.1 角色概述	178
7.1.13 VARP 函数	145	9.2.2 角色的创建	179
7.2 使用 GROUP BY 子句	145	9.2.3 角色权限的更改	182
7.3 使用 UNION 组合结果	147	9.2.4 角色成员的更改	183
7.4 子查询	149	9.2.5 固定服务器角色	184
7.5 使用 CASE 处理条件数据	151	9.3 实例精讲	186
7.6 并行查询处理	154	9.3.1 入门: 创建一个角色	186
7.7 分布式查询	155	9.3.2 提高: 为角色分配权限	187
7.8 小结	156	9.3.3 熟练: 为特定的用户设定特定的 权限	187
7.9 习题	156	9.4 小结	188
第 8 章 身份认证模式		9.5 习题	188
8.1 SQL Server 的身份认证模式概述	157	第 10 章 数据的备份和还原	
8.1.1 登录 SQL Server	157	10.1 备份和还原概述	190
8.1.2 Windows NT 认证模式	158	10.1.1 备份和还原的概念	190
8.1.3 混合认证模式	158	10.1.2 数据库系统故障	191
8.1.4 设定验证模式	159	10.1.3 SQL Server 的事务日志	192
8.1.5 登录账户和用户	159	10.2 备份	195
8.2 创建用户账户	160	10.2.1 备份的种类	195
8.2.1 用登录向导创建用户账户	160	10.2.2 备份操作	196
8.2.2 用企业管理器创建用户账户	162	10.3 还原	205
8.2.3 用 T-SQL 语句创建登录账户	164	10.3.1 还原概述	205
8.3 建立和管理 SQL Server 数据库用户	164	10.3.2 还原操作	206
8.3.1 用企业管理器创建用户	165	10.4 实例精讲	212
8.3.2 用 T-SQL 语句创建数据库用户	165	10.4.1 入门: 创建备份	212
8.4 实例精讲	166	10.4.2 提高: 对数据库进行恢复	213
8.4.1 入门: 创建一个登录账户	166	10.4.3 熟练: 恢复到某个故障点	215
8.4.2 提高: 创建一个数据库用户	168	10.5 小结	217
8.4.3 熟练: 管理 SQL Server 登录	169		

10.6 习题	218	12.2.12 关闭和启动触发器	260
第 11 章 SQL 代理服务		12.2.13 管理触发器	260
11.1 SQL 代理服务简介	219	12.3 使用企业管理器管理触发器	261
11.1.1 SQL 代理服务概述	219	12.3.1 创建触发器	262
11.1.2 SQL 代理服务错误日志	220	12.3.2 修改触发器	262
11.2 作业	221	12.3.3 删除触发器	263
11.2.1 作业概述	221	12.4 约束	263
11.2.2 创建作业	221	12.4.1 NOT NULL 约束	263
11.3 警报	231	12.4.2 CHECK 约束	264
11.3.1 警报概述	231	12.4.3 UNIQUE 约束	265
11.3.2 创建警报	231	12.4.4 PRIMARY KEY 约束	266
11.4 操作员	237	12.4.5 FOREIGN KEY 约束	267
11.4.1 操作员概述	237	12.5 默认值	270
11.4.2 创建操作员	238	12.5.1 默认值的相关 T-SQL 语句	270
11.5 实例精讲	240	12.5.2 使用企业管理器管理默认值	271
11.5.1 入门: 建立备份 BSQFlow 数据库的 作业	240	12.6 实例精讲	272
11.5.2 提高: 创建警报和操作员	244	12.7 小结	274
11.5.3 熟练: 使用 T-SQL 语言创建 作业	245	12.8 习题	275
11.6 小结	247	第 13 章 事务和锁定	
11.7 习题	247	13.1 事务概述	276
第 12 章 数据完整性		13.1.1 ACID 属性	276
12.1 数据完整性概述	248	13.1.2 事务的几种模式	278
12.2 触发器	249	13.1.3 事务的相关信息	281
12.2.1 deleted 和 inserted 表	249	13.2 事务还原	283
12.2.2 触发器概述	249	13.2.1 系统自动还原	283
12.2.3 触发器的基本操作	250	13.2.2 程序还原	283
12.2.4 一个简单的触发器	250	13.3 事务锁定	283
12.2.5 创建 INSERT 触发器	252	13.3.1 共享锁 (Shared)	284
12.2.6 创建 DELETE 触发器	253	13.3.2 排他锁 (Exclusive)	285
12.2.7 创建 UPDATE 触发器	255	13.3.3 更新锁 (Update)	285
12.2.8 创建 INSTEAD OF 触发器	257	13.3.4 意向锁 (Intent)	285
12.2.9 创建 AFTER 触发器	258	13.3.5 构架锁 (Schema)	285
12.2.10 修改触发器	260	13.3.6 大容量更新锁 (Bulk Update)	285
12.2.11 删除触发器	260	13.4 实例精讲	285
		13.4.1 入门: 开始显式事务	285
		13.4.2 进阶: 设置保存点	287

13.4.3 熟练: 锁的使用	288	交易	327
13.5 小结	289	15.7.3 精通: 使用 MS DTC 实现电子 商务	328
13.6 习题	289	15.8 小结	329
第 14 章 复制		15.9 习题	329
14.1 复制简介	291	第 16 章 在各种环境中使用 SQL Server	
14.1.1 复制的重要概念	291	16.1 访问接口简介	330
14.1.2 复制的过程	292	16.1.1 ODBC 简介	330
14.1.3 设定发布和订阅	293	16.1.2 ADO 简介	333
14.2 快照复制	297	16.2 在各种环境中使用 SQL Server	334
14.2.1 快照复制概述	297	16.2.1 在 Visual C++ 中使用 SQL Server	334
14.2.2 发布快照复制	298	16.2.2 在 Visual Basic 中使用 SQL Server	336
14.3 事务复制	309	16.2.3 在 ASP 中使用 SQL Server	337
14.4 合并复制	309	16.2.4 在 JSP 或 Java Servlet 中使用 SQL Server	339
14.5 实例精讲	310	16.3 实例解析	340
14.5.1 入门: 配置发布服务器	310	16.3.1 入门: 用 Visual Basic 连接到 数据库	340
14.5.2 进阶: 发布事务复制	311	16.3.2 提高: 用 Visual C++ 操作数据库	341
14.5.3 熟练: 订阅事务复制	313	16.3.3 熟练: 一个用 JSP 完成注册的 例子	343
14.6 小结	316	16.4 小结	347
14.7 习题	316	16.5 习题	347
第 15 章 组件服务		第 17 章 售后管理系统	
15.1 管理控制台	317	17.1 数据库设计	349
15.2 COM+ 应用程序	318	17.2 系统设计	351
15.3 事件查看器	319	17.3 具体实现步骤	351
15.4 Windows 服务的管理	320	17.3.1 ODBC 的设置	351
15.5 分布式事务处理协调器	323	17.3.2 Delphi 表单的设计	354
15.5.1 概念	323	17.3.3 连接数据库的具体实现	356
15.5.2 MS DTC 的管理	324	17.4 小结	361
15.6 MS DTC 应用实例	325	附录 习题答案	
15.6.1 银行交易	325		
15.6.2 电子商务	325		
15.7 实例精讲	326		
15.7.1 入门: 一个简单的 MS DTC 运用	326		
15.7.2 提高: 使用 MS DTC 实现银行			

第 1 章 数据库系统介绍

本章概述

数据库技术作为计算机科学的一门分支学科，已经有几十年的历史。目前，数据库技术已经广泛应用于企业和机构，成为其中数据运转不可缺少的中枢。数据库系统到现在经历了多个阶段，发展出了许多新特性，但其中蕴含着的原理和方法并没有太多的改变，这些原理和方法的掌握对于理解数据库应用来说有着非常重要的作用。

学习要点

- 数据库的历史
- 数据库的概念
- 数据模型
- 数据库范式
- E-R 模型

1.1 数据库概述

数据库已经有几十年的发展历史，如今它的应用非常广泛，从学生的课程表、公司的销售记录，到人才市场的档案库、证券市场的交易记录等都可以应用数据库来进行方便的存储、管理以及查看操作。

1.1.1 数据库系统的历史

在 20 世纪 60 年代以前，计算机主要应用在国防和科学计算等数值计算方面，人们处理的数据较少，采用文件方式进行处理就能够满足处理数据的需要，到了上世纪 70 年代，随着计算机处理能力的加强和存储容量的提高，使得大量的数据依靠计算机处理变成了可能。这时，处理数据的速度及数据存储的效率变得极其重要，人们开始对如何进一步提高数据使用效率进行研究，这一时间开始出现了 DBMS、层次和网状数据库、关系数据库、E-R 模型、查询语言 SQL、查询优化、索引技术，事务管理，以及 OLTP 等许多至今还在沿用的技术。

自上世纪 80 年代以来，计算机技术以惊人的速度发展着，并且以前所未有的速度扩散至地球的每一个角落。存储介质的迅速发展，使得信息存储单位也由字节提升到 GB 甚至 TB，数据的组织和管理也日益严峻地展现在人们面前。如何安全、高效、容易地组织和管理大量的数据，成为了数据库系统必须解决的问题。在这个时期开始出现高级数据模型，其中包括扩充关系和对象关系，以及面向应用的研究、数据仓库和数据挖掘等技术。

1.1.2 数据库系统

数据库系统是指以数据库为基础，为用户提供数据信息服务的系统。通常它由以下部分组成。

- (1) 计算机及计算机网络。
- (2) 数据库和数据库管理系统 (DBMS)。
- (3) 以前两者为基础的数据库应用程序。

计算机和计算机网络构成了数据库系统的硬件部分，即将学习的 SQL Server 2000 就属于一个数据库和数据库管理系统，而在 SQL Server 2000 上开发的一些系统，如股票交易系统或酒店管理系统等就是数据库的应用程序。

图 1-1 所示为一个典型的数据库系统的应用图。画面中最左边的用户通过计算机终端上的数据库应用程序来使用数据库，通过数据库应用程序访问数据库可以让用户不必关心访问数据库的细节。计算机及计算机网络构成了数据库的物理基础，所有数据库应用程序、数据库及数据库管理系统都必须通过计算机及计算机网络才能实现。图 1-1 的最右端是数据库服务器，上面存放了数据库及数据库管理系统，用于让管理员对数据库进行操作和维护。

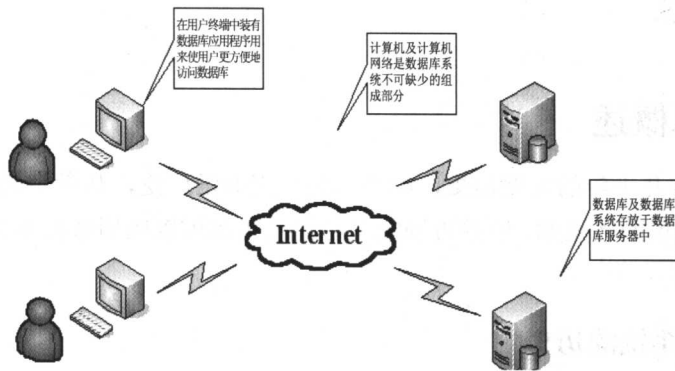


图 1-1 数据库系统示意图

一个数据库系统应当具有如下特征。

- (1) 数据冗余小：任一数据在数据库系统中应有尽可能少的副本。
- (2) 数据完整：需求的数据应完整地包含在数据库系统中。
- (3) 数据集成：数据库系统应将数据对象的各个相关数据整理成一个单独的逻辑数据结构。
- (4) 提供数据共享：数据库系统中多个用户可共享同一数据信息。
- (5) 提供安全控制：对不同的数据库用户提供不同的数据集合，不支持用户跨集合共享数据，如员工 A 仅可查看员工 A 的工资表，而不可查看员工 B 的工资表。
- (6) 具备数据独立性：在数据库中的数据，通过开放数据库接口与应用程序交换信息，这样使应用程序与数据相互分离开来。

1.1.3 数据库

数据库系统的核心是数据库，它是一个为了满足人们信息需求而使用的相互关联的有序数据集合。在这里，信息需求有序、相互关联和数据集合是数据库的特征。要注意的是数据库是一个满足人们信息需要的数据集合。这意味着，人们不需要的信息不应当成为这个数据库的一部分，并且数据库中的数据组织不管如何复杂，其目的都是为了呈现在用户面前一个简单而满足需求的信息。

数据库在结构上可以划分为3个层次：物理层、概念层和用户层。

(1) 物理层是数据库最核心的层次。它是存储和传输数据的物理设备集合，即物理数据库，物理层由物理模式（又称为内部模式）描述。

(2) 概念层是数据库的逻辑表示，包括数据的逻辑定义以及数据间的逻辑联系。概念层由概念模式定义，这一层也被称为概念模型。

(3) 用户层是用户所看到和操作的数据库，它是一个或几个特定用户所使用的数据子集，还是概念模型的逻辑子集。它的定义由外部模式决定。

图 1-2 所示为一个数据库示意图。其中描述了一个数据库的常见元素。表是最基本的元素。约束是表中属性的限制。索引用于提高某个属性的查找效率。触发器用于在某一事件发生时自动执行需要的操作。视图在形式上也是表格，但它是对数据库表的一组“观察值”，这组“观察值”是各个用户所需要组合的数据。存储过程类似于函数，用于完成某一特定目的的一系列操作。

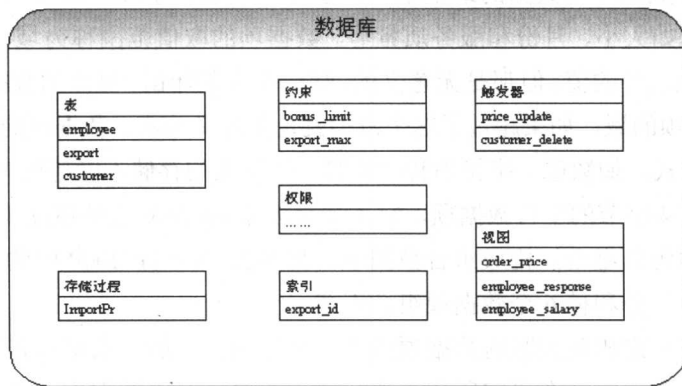


图 1-2 数据库示意图

有序和相互关联则表明了数据库并非是一团杂乱的数据，它们是一组有着相互关系的并按照一定顺序排列的数据。图 1-3 所示的数据库仅列出了表元素的关系。由图可见，首先从数据库中最左边看起，雇员表 (Employee) 中的雇员号属性 (empid) 对应着联系表 (Contact) 中的雇员号属性 empid。这样，联系表 (Contact) 就表明了某个雇员负责的所有顾客。然后再由顾客号关联到顾客表 (Customer) 中的顾客号 (custid)，在顾客表 (Customer) 中，每个顾客号对应了一个顾客的属性，如名字 (name)、移动电话 (mobilephone) 和地址 (address)。

等信息,而最后的货物号(stdid)又对应了这个顾客所需要的货物。通过对应货物表(Storehouse)中的货物号属性,就可以找到这个货物的名称、进仓时间、价值和体积等细节信息。

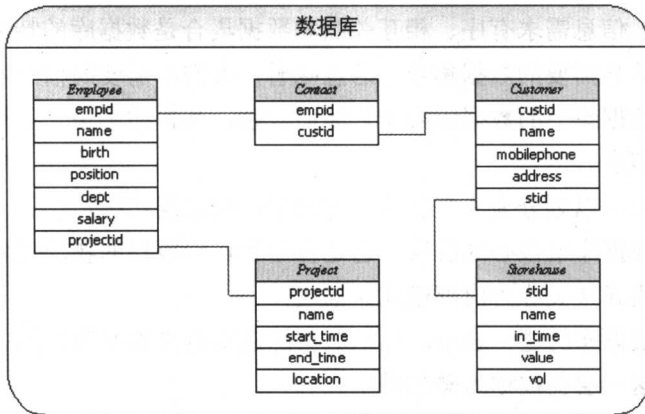


图 1-3 数据库的数据组织示意图

上述的一系列表与表的联系直观地表示了一个雇员负责的顾客及顾客所需货物的联系。

1.1.4 数据的组织结构

在数据库中,数据是现实世界中信息的载体,也是现实世界中的信息在数据库中的物理映射。数据的组织按概念可以分为 4 级:数据项、记录、文件和数据库(从小到大)。

(1) 数据项:数据项是数据的最小单位,也称为元素、字段。数据项与现实世界的某一个属性相对应,比如大小、月份和服务质量等。数据项的取值范围称为域,如果不设定约束,数据项可以接受域以外的值,但那是无意义的。如一个人的年龄,可能的数据项不能超过 200,200 就是这个数据项的域,如果超过了这个范围就是无意义的值。数据项的值可以是任何系统定义的有意义的形式,如数值、字符串和时间等。数据项的存储不能超过其规定的存储空间,比如一个仅能存放 4 字节的字符数据项,它不能存放如 abcdefg 这种超过 4 字节长度的值。

不同的数据项可以组合,构成组合数据项。如 SQL Server 2000 中的“日期时间”可以由日、月、年、时、分和秒 6 个数据项组合而成。

(2) 记录:有一定相互关联的数据项组成一个记录。一般一条记录是表示某个现实世界中的实体,如一个公司、一个研究所和一个科研小组等。对于数据库系统来说,记录是处理和存储信息的基本单位。记录常常是用户关心的实体的数据集合,构成该记录的数据项表示实体的若干属性。

(3) 文件:数据库文件中存放的各个记录的值是给定类型的(逻辑)记录的全部具体值的集合。文件根据记录的组织方式和存取方法可以分为:顺序文件、索引文件、直接文件和倒排文件等。

(4) 数据库:它是比文件更大的数据组织。非空数据库包含一个或一个以上的文件。数据库可以看成是具有一定联系的数据的集合,也可以看成是具有一定联系的多个记录的集合。

1.1.5 分布式数据库

数据库依据位置的不同可分为集中式数据库和分布式数据库。集中式数据库指数据库所有的部分均集中在同一位置。比如，小型企业服务器均在一个机房内，可以很好地实现数据库服务器的管理，但访问服务器的各客户进行不同的远程地点访问时，因每个客户机都要通过远程链路访问，其访问效率将大打折扣。

图 1-4 所示为集中式数据库模式，近距离用户通过局域网访问数据库，速度和效率很高，但远程的用户通过公司 VPN 网络连入，中间必须经过广域网，他们的数据库访问效率较低。

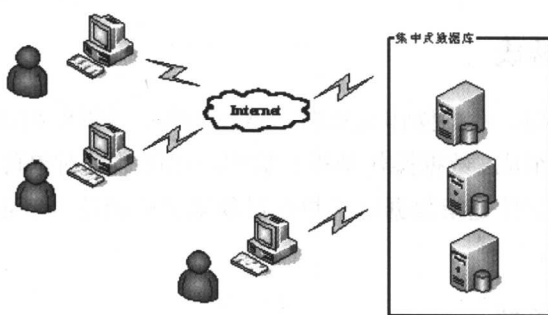


图 1-4 集中式数据库模式

分布式数据库指数据库的各个部分分布在不同的位置。这样，若有许多远程客户机访问，它们在大部分情况下可以仅访问处于它们本地的服务器以获得所需的数据。

图 1-5 所示为分布式数据库系统，A，B，C 三个位置的数据库服务器均为一个分布式数据库服务器，位置 A 的数据库服务器上大量存放了当地用户常用的数据，当 A 位置的用户需要用到数据时，首先在 A 位置的服务器上寻找，如果有需要的数据，则在本地对数据库进行操作，如果没有需要的数据，才通过 Internet 连接到数据存放的另一个分布式数据库服务器中进行操作。这样，如果数据库服务器设计适当，大部分的数据操作都可以在本地进行，仅需对极小部分数据进行远程操作，大大提高了数据库操作的效率。

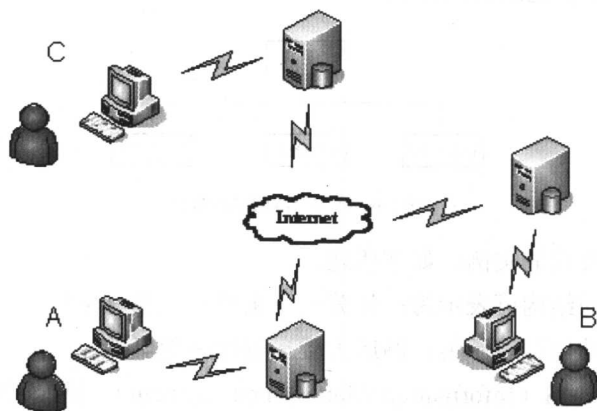


图 1-5 分布式数据库模式

1.2 数据模型

数据模型是一种工具，它描述出了人们的信息需求，并将这种需求通过易于数据库系统实现的形式表现出来。简而言之，数据模型就是将信息需求抽象化、规范化后形成的一套模型。

数据模型用于描述人们信息需求，它显然可以描述现实世界，并以易于数据库系统实现的形式表现出来，使得它将成为数据库结构的基础。由此可见，数据模型是沟通现实世界和抽象计算机世界的桥梁。

1.2.1 数据模型的构成

数据模型由数据结构、数据操作及数据完整性构成。数据结构由数据对象及该对象中所有数据成员之间的关系组成。数据操作是指对数据库中的数据对象可能进行的操作，如查询、插入和更新等。数据完整性是指数据对于数据对象来说应当符合一定的规则或制约，使得数据对象合理正确地相容。

1.2.2 数据模型的类型

数据模型应当尽可能地接近现实的信息需求，经多年的发展已形成如下几种常见的数据模型。

1. 层次结构模型

层次数据模型来源于数据结构中的树，是一种类似于树的结构。层次模型的特征如下。

- (1) 它有且仅有一个根结点。
- (2) 一个结点与另一个结点若有联系则尽可能为“父子关系”。
- (3) 每个结点均处于某一级别之上。
- (4) 每个结点均可通过“父子关系”的指针找到。

图 1-6 所示为一个常见的层次模型。

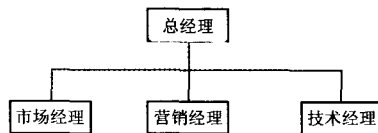


图 1-6 人力资源的层次结构

有许多基于层次结构的实例，如下所述。

- (1) 一台计算机的结构可表示为：计算机→主机→主板→电容
- (2) 一个酒店的结构可表示为：酒店大楼→餐厅→餐桌

IBM 公司推出的 IMS (Information Management System) 是层次型数据库系统的典型代表。层次型数据库对于非层次型数据对象类型来说，其使用效率较低，因此现在已很少有人

使用层次型数据模型了。

2. 网状数据模型

网状数据模型是一种类似于图的结构，其特征如下。

- (1) 不存在级别。
- (2) 一个结点可拥有多个父结点或多个子结点。
- (3) 记录有若干数据项，且这些数据项可有多个值。

网状数据模型在现实生活中很普遍，如一个客户与销售人员和产品的联系，主要体现在：一个客户可接受多个销售人员的销售，而一个销售人员可销售多个产品给客户，一个产品可被销售给多个客户，它也可被多个销售人员销售。

图 1-7 所示为一个网状数据模型。这是一个原料厂和商店的网状模型，其中原料厂 A 可为商店 A、商店 B 和商店 C 供货。同样的，商店 A 也可以从原料厂 A、原料厂 B 和原料厂 C 进货。在这里并不存在父子的级别关系，并且点与点之间是多对多的关系。

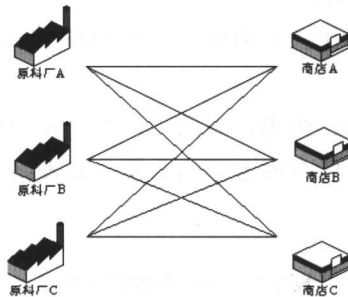


图 1-7 网状数据模型

网状数据模型的代表有 Cullinet 软件公司的 IDMS、Honeywell 公司的 IDSII、Univac 公司的 DMS1100，以及 HP 公司的 IMAGE 等。网状数据库对于层次型和非层次型的结构都可以很好地模拟，故在数据库发展史上占有重要的地位。

3. 关系数据模型

网状数据库和层次数据库对数据的集中以及共享问题进行了一定程度的解决，但是在数据独立性和抽象程度上仍显不足。

1970 年 6 月是关系数据库的一个里程碑。在这个月里，IBM 圣约瑟研究实验室的高级研究员埃德加·考特在 Communications of ACM 上发表了《大型共享数据库数据的关系模型》一文。ACM 后来在 1983 年把这篇论文列为从 1958 年以来的 25 年中最具里程碑意义的 25 篇论文之一，因为它首次明确而清晰地数据库系统提出了一种崭新的模型，即关系模型。

“关系”是数学中的一个基本概念，集合中的元素所组成的若干有序偶对就是关系，它反映了客观事物间的一定联系。如产业链中的利益关系，生物种群中的分工关系，人与人之间的关系等。计算机的结构设计、编译器设计、算法分析与程序结构、信息检索等，都应用了关系的概念。考特是第一个用关系的概念来建立数据模型，用以描述、设计与操纵数据库的人。

关系数据模型是以集合论中的关系概念为基础发展起来的。关系模型中无论是实体还是实体间的联系均由单一的结构类型——关系来表示。在实际关系数据库中的关系也称为表。一个关系数据库主要是由若干个表组成。

关系式数据模型把一些复杂的数据结构归结为简单的二元关系，即行列对应的表格形式。关系数据模型通过外键来建立数据结构间的关系，外键即表与表之间的某一属性联系，这是关系数据模型与网状数据模型和层次数据模型之间的一个显著区别，在变换存储方式或物理位置时，层次模型和网状模型必须改变其物理指针，而关系型数据模型在进行上述改变时，只需改变外键即可完成变换过程。

关系数据模型和网状模型有许多类似之处，同时存在如下特征。

- (1) 具有层次级别。
- (2) 由关系表组成。
- (3) 关系可一对多。
- (4) 各关系表间由外键建立联系。

现代的数据库绝大多数都属于关系数据库，如 DBI, FoxPro, Access, Oracle, SQL Server, Sybase 以及 System 2000 等。

关系数据库的关系表是一张二维表，表的“行”表示一个实例，即一个具体的对象，而表的“列”表示属性，即这个实例有哪方面的数据，而表格中的数据就表明了这个实例某一属性的具体值，如表 1-1 所示。

表 1-1 一张典型的关系表

姓 名	职 位	工 资	办 公 室
李小勇	总经理	5613	A-131
张 斌	人事经理	4211	A-132
陈 瑞	销售经理	4015	B-432
王进飞	采购员	2702	B-421
卢 进	采购员	2702	B-423

在表 1-1 中，姓名、职位、工资和办公室等列就是关系表的属性，它表示的是需记录对象的各方面属性，而每 1 行表示一个具体的人，即前面所述的一个具体实例。

在这里应注意行和列不能颠倒，如表 1-2 所示的表不是关系表。

表 1-2 错误的关系表

姓 名	李小勇	张 斌	陈 瑞	王进飞	卢 进
职 位	总经理	人事经理	销售经理	采购员	采购员
工 资	5613	4211	4015	2702	2702
办 公 室	A-131	A-132	B-432	B-421	B-423

主键是指可以惟一地标识某个实例的属性。这个属性之所以可以惟一地标识一个实例，是因为在一个关系表中，任意两个实例的这个属性都是不相同的。这种属性叫关键属性或主