



普通高等教育“十一五”国家级规划教材

大学计算机规划教材

大型数据库管理系统

技术、应用与实例分析

——基于SQL Server (第3版)

◆ 孟宪虎 马雪英 邓绪斌 编著



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

普通高等教育“十一五”国家级规划教材
大学计算机规划教材

大型数据库管理系统

技术、应用与实例分析

——基于SQL Server（第3版）

孟宪虎 马雪英 邓绪斌 编著

电子工业出版社
Publishing House of Electronics Industry
北京·BEIJING

内 容 简 介

本书是普通高等教育“十一五”国家级规划教材。全书系统地介绍 SQL Server 的相关原理、管理和应用程序设计,原理、管理、程序设计并重,以一个数据库实例贯穿始终,将 SQL Server 技术分解成 16 个专题,每个专题根据自身技术特点对操作进行实例分析。本书主要内容包括:数据库原理与数据库应用设计,数据库存储原理和数据库创建,数据库表结构分析和表的创建与管理;数据库程序机构及设计,自定义函数设计,游标技术及操作,存储过程和触发器,事务和并发控制,视图的规划、设计与操作,索引的机理和设计,数据安全及访问控制、数据备份与恢复等。每章均配有针对性实验,提供配套电子课件、实例数据库、习题参考答案和教学网站。

本书可作为高等学校计算机和信息管理等相关专业的本科生和研究生教材,也可作为社会相关从业人员的自学和培训教材,对于开发信息管理系统的技术人员具有较高的参考价值。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

大型数据库管理系统技术、应用与实例分析:基于 SQL Server/孟宪虎,马雪英,邓绪斌编著. —3版. —北京:电子工业出版社,2016.7

普通高等教育“十一五”国家级规划教材

ISBN 978-7-121-28676-6

I. ①大… II. ①孟… ②马… ③邓… 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(2016)第 092057 号

策划编辑:王羽佳

责任编辑:王羽佳 特约编辑:曹剑锋

印 刷:三河市华成印务有限公司

装 订:三河市华成印务有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编:100036

开 本:787×1092 1/16 印张:18.75 字数:545 千字

版 次:2008 年 7 月第 1 版

2016 年 7 月第 3 版

印 次:2016 年 7 月第 1 次印刷

定 价:45.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888, 88258888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式:(010)88254535, wyj@phei.com.cn。

第3版前言

当下我们处在信息社会已经是毋庸置疑的了。从淘宝一天网上几百亿的交易金额，到春晚几百亿次“摇一摇”微信红包，或是公众一天上亿次对“穹顶之下”的刷屏，再到大数据下的精准广告投放，以及国家大力倡导的“互联网+”，这里面都有一个技术在支撑，那就是数据库技术。

数据库管理系统是数据库技术的核心，是一个数据管理软件，具有数据定义、数据操作、数据库运行管理和数据库维护等功能。近年，为了适应信息社会快速发展的需要，各种数据库管理系统不断升级换代，新的技术不断出现。

SQL Server 2000到SQL Server 2005经历了5年，再到SQL Server 2008用了3年，然后是SQL Server 2012，现在SQL Server 2014已经开始面世。不可否认，SQL Server每推出一个新的版本，都会有新的技术在里面体现，数据库管理系统的高级应用人员应该关注新的功能。但是，这本教材面向的是本科生或研究生，讲述的是如何进行数据库设计、创建和使用，其内容是数据库管理系统中最基本、最核心的知识和技术。纵观SQL Server发展变化，数据库操作和T-SQL变化并不大，因此，这次修订淡化了SQL Server具体版本的概念，对某些只是在某个SQL Server版本上才能操作的内容进行了删除，仅保留了个别必须用具体版本才能说明的例子。除个别程序特殊必须在某些版本上使用外，书中几乎所有程序均可在SQL Server 2000到SQL Server 2012系统上运行。

随着数据库承载数据量的不断加大和数据库管理系统对应功能的扩展和增强，数据库优化越来越重要，书中对数据查询优化做了简单介绍。如果读者需要了解数据库优化更深入的知识和技术，欢迎和作者探讨。

本书第1版于2008年7月出版，2011年进行了第2版修订，架构做了比较大的改动，在经过多次印刷使用的基础上，现再次进行修订淡化了具体软件版本的概念，并改正了原来存在的错误。尽管如此，仍然会有不合理或错误的地方，恳请读者批评指正！

感谢一贯支持的读者和同事！

孟宪虎

2016年7月于慧鸣湖

前 言

本书是普通高等教育“十一五”国家级规划教材。

大型数据库管理系统 SQL Server 是近年来国内外广泛使用的数据库管理系统，它既支持 C/S 模式系统，也支持 B/S 模式系统，是开发管理信息系统常用的主流数据库管理系统之一。国内许多高等学校已陆续开设了这门课程。但是，对于 SQL Server，市场上大量的相关书籍有的是围绕实用技术讲解的，有些类似于帮助说明书、专题介绍，而有些偏重于程序设计，有些主要讲解原理。对于专门学习大型数据库管理系统技术的高年级本科生和研究生在有限的时间内学到各种需要的知识点，明显不方便，因此在教学中不得不准备多本相关参考书。

基于上述原因，将大型数据库管理系统的基本原理和在这些原理指导下的数据库管理、技术和程序设计结合起来是本书作者想要得到的结果，也是学生参加工作前急需掌握的一门重要技术。因此，作者基于多年该门课程教学的体会和经验，参考了众多数据库和 SQL Server 相关书籍及其他资料，编写了这本教材。

本书可作为高等学校计算机应用和信息管理与信息系统等相关专业本科生和研究生的教材，也可作为社会相关从业人员的自学培训教材，对于开发信息管理系统的技术人员具有较高的参考价值。

本书具有以下特色：

- (1) 原理、管理、程序设计并重，使得内容全面，例题丰富，概念清晰，循序渐进，易于学习。
- (2) 分解难点，设计成 16 个专题，既独立成章，又前后关联，便于教师教学。
- (3) 采用通俗易懂、容易理解的方法叙述复杂的概念，结合实例分析，帮助读者逐步掌握必要的技术知识。
- (4) 采用一个实例贯穿全书，使学生可以通过本书学习，彻底明白如何用 SQL Server 技术完成数据库应用系统的设计、实现和管理。

本书主要内容包括：数据库基础知识、数据库应用设计、SQL Server 2005 安装、配置与管理、SQL Server 数据库存储原理和创建、表的结构分析及完整性创建管理、SQL Server 查询处理和表数据编辑、T-SQL 程序结构、SQL Server 事务和并发控制、视图的规划与操作、索引的机理、规划和设计、游标操作和应用、存储过程和用户存储过程设计、触发器原理及使用、用户自定义函数设计、数据库安全及访问控制、数据备份与恢复、数据复制等，每章均配有针对性实验。

本书向使用本书作为教材的教师提供配套电子课件、实例数据库、习题参考答案，请登录华信教育资源网 (<http://www.hxedu.com.cn>) 注册下载。

本书第 1 版于 2008 年 7 月出版，经过 3 年实践教学的检验，结合广大使用本书作为教材的老师和同学提出的建议和意见，我对全书内容进行了整体修订。本次修订引用或参考了第 1 版的重要内容成果，在此向马雪英老师和邓绪斌老师深表谢意！

本书在编写过程中同时得到了浙江财经学院领导和信息学院领导及全体同事的大力支持，在此向他们和所有帮助和关心本书编写的朋友致以衷心的感谢！

在编写本书的过程中，我们参考了众多相关参考书、资料和 SQL Server 联机帮助，为了表示尊敬和感谢，在本书的最后我们尽量罗列说明，如有遗漏敬请谅解。

书中难免有许多不足和错误之处，恳请读者批评指正！

孟宪虎
2011 年 5 月

目 录

第 1 章 数据库基础和数据库设计	1
1.1 数据库系统	1
1.1.1 数据、信息、数据库	1
1.1.2 数据库管理系统	2
1.2 数据库系统结构	2
1.2.1 数据库系统模式的概念	2
1.2.2 数据库系统的三级模式结构	3
1.2.3 数据库的二级映像功能与数据独立性	4
1.2.4 数据库系统用户结构	4
1.3 关系数据库及其设计	6
1.3.1 关系数据库	6
1.3.2 关系数据库设计	6
1.3.3 关系数据库的完整性	12
1.4 教学管理数据库操作任务	13
实验与思考	15
第 2 章 服务器的安装配置和使用	17
2.1 SQL Server 概述	17
2.1.1 SQL Server 版本和环境需求	17
2.1.2 SQL Server 的特点和组成	17
2.1.3 SQL Server 安装注意事项	19
2.1.4 SQL Server Management Studio 介绍	21
2.1.5 分离和附加数据库文件	22
2.2 网络协议配置	23
2.2.1 SQL Server 通信结构	23
2.2.2 配置服务器端网络协议	24
2.2.3 配置客户端网络协议	25
2.3 添加新的注册服务器	25
2.3.1 新建注册服务器	25
2.3.2 连接到数据库服务器	26
2.4 链接服务器建立及其使用	26
2.4.1 链接服务器简介	26
2.4.2 创建链接服务器	27
2.4.3 创建链接服务器登录标志	28
2.4.4 访问链接服务器	29
2.4.5 访问链接服务器的实例	30
实验与思考	31
第 3 章 SQL Server 数据库结构和管理	33
3.1 数据库物理存储结构	33
3.1.1 数据库文件和文件组	33
3.1.2 数据文件的使用分配	35
3.1.3 事务日志文件结构	37
3.2 数据库的逻辑组织	39
3.2.1 数据库构架	39
3.2.2 系统数据库	40
3.2.3 用户数据库	41
3.3 数据库创建与管理	41
3.3.1 创建数据库	41
3.3.2 管理数据库	44
实验与思考	49
第 4 章 表的存储原理及完整性创建管理	50
4.1 SQL Server 表的类型	50
4.1.1 SQL Server 的临时表	50
4.1.2 SQL Server 的系统表和系统视图	51
4.2 表的存储原理	52
4.2.1 内部存储概述	52
4.2.2 SQL Server 数据记录结构	53
4.3 SQL Server 数据类型	56
4.3.1 数值型数据	56
4.3.2 货币型数据	56
4.3.3 字符型数据	57
4.3.4 日期/时间数据类型	57
4.4 数据表的创建和管理	57
4.4.1 数据表结构的创建	57
4.4.2 数据表结构的管理	64
实验与思考	66
第 5 章 查询处理和表数据编辑	68
5.1 查询数据	68
5.1.1 简单查询	68

5.1.2	统计	75	7.2	事务的分类和控制	111
5.1.3	连接查询	78	7.2.1	事务的分类	111
5.1.4	子查询	80	7.2.2	事务控制	112
5.1.5	联合查询	83	7.3	编写有效事务的建议	114
5.2	表数据编辑	83	7.3.1	编写有效事务的指导原则	114
5.2.1	插入数据	83	7.3.2	避免并发问题	114
5.2.2	修改数据	85	7.4	事务处理实例分析	115
5.2.3	删除数据	86	7.5	分布式事务	118
实验与思考		87	7.5.1	分布式事务的两阶段提交	118
第6章 索引的机理、规划和管理		89	7.5.2	分布式事务的处理过程	119
6.1	索引的作用与结构	89	7.5.3	分布式事务实例分析	119
6.1.1	索引概述	89	7.6	并发控制	120
6.1.2	SQL Server 索引下的数据组织 结构	90	7.6.1	SQL Server 锁的粒度及模式	120
6.2	索引类型	92	7.6.2	封锁协议	123
6.2.1	聚集索引和非聚集索引	93	7.6.3	事务隔离	124
6.2.2	主键索引和非主键索引	93	7.6.4	死锁处理	126
6.2.3	唯一索引和非唯一索引	93	实验与思考		127
6.2.4	单列索引和复合索引	93	第8章 Transact-SQL 程序结构		129
6.3	规划设计索引的一般原则	94	8.1	注释和变量	129
6.3.1	什么类型查询适合建立索引	94	8.1.1	T-SQL 程序的基本结构	129
6.3.2	索引设计的其他准则	94	8.1.2	注释	131
6.3.3	索引的特征	95	8.1.3	变量	132
6.3.4	在文件组上合理放置索引	95	8.1.4	变量赋值	132
6.3.5	索引优化建议	96	8.2	运算符和表达式	133
6.4	索引的创建和删除	96	8.2.1	算术运算符	133
6.4.1	创建索引	96	8.2.2	位运算符	133
6.4.2	删除索引	99	8.2.3	连接运算符	133
6.5	查询中的执行计划	100	8.2.4	比较运算符	133
6.5.1	查看查询执行计划	101	8.2.5	逻辑运算符	133
6.5.2	索引和未索引执行计划的比较	102	8.2.6	表达式	134
6.6	索引使用中的维护	105	8.3	函数	134
6.6.1	维护索引的统计信息	105	8.3.1	数学函数	134
6.6.2	维护索引碎片	106	8.3.2	字符串函数	135
实验与思考		108	8.3.3	时间日期函数	137
第7章 SQL Server 事务和并发控制		109	8.3.4	转换函数	138
7.1	事务	109	8.3.5	配置函数	139
7.1.1	事务与并发控制的关系	109	8.4	流程控制	140
7.1.2	事务对保障数据一致和完整性 的作用	110	8.4.1	块语句	140
			8.4.2	条件语句	141
			8.4.3	CASE 语句	141

8.4.4	循环语句	143	11.4	修改和删除用户自定义函数	188
8.4.5	等待语句	143	11.4.1	修改用户自定义函数	188
8.4.6	GOTO 语句	144	11.4.2	删除用户自定义函数	190
8.4.7	返回语句	144	11.5	用户自定义函数实例分析	190
8.5	程序应用实例分析	145	实验与思考		192
	实验与思考	147	第 12 章	存储过程和用户存储过程设计	194
第 9 章	视图的规划与操作	148	12.1	存储过程概述	194
9.1	视图的作用与规划	148	12.1.1	存储过程的概念和分类	194
9.1.1	视图的作用	148	12.1.2	存储过程的优点	194
9.1.2	视图的规划	149	12.2	系统存储过程	195
9.2	视图操作	150	12.2.1	系统存储过程分类	195
9.2.1	创建视图	150	12.2.2	一些常用的系统存储过程	196
9.2.2	视图的修改、重命名和删除	154	12.3	创建和执行用户存储过程	197
9.2.3	查询视图	156	12.3.1	创建用户存储过程	197
9.2.4	更新视图	157	12.3.2	执行用户存储过程	199
9.2.5	特殊类型视图简介	160	12.4	带状态参数的存储过程及实例分析	203
9.3	视图应用综合实例分析	164	12.4.1	存储过程执行状态值的返回	203
实验与思考		167	12.4.2	实例分析	203
第 10 章	游标操作和应用	168	12.5	修改和删除存储过程	205
10.1	游标声明	169	12.5.1	修改存储过程	205
10.1.1	游标声明命令	169	12.5.2	删除存储过程	206
10.1.2	游标变量	171	12.6	存储过程设计实例分析	207
10.2	游标数据操作	172	实验与思考		211
10.2.1	打开游标	172	第 13 章	触发器原理及使用	212
10.2.2	读取游标数据	173	13.1	触发器基本概念	212
10.2.3	关闭游标	175	13.1.1	触发器的概念及作用	212
10.2.4	释放游标	175	13.1.2	触发器的种类	213
10.2.5	游标定位修改和删除操作	177	13.2	触发器原理	213
10.3	游标应用实例分析	178	13.2.1	插入表的功能	213
实验与思考		180	13.2.2	删除表的功能	214
第 11 章	用户自定义函数设计	181	13.2.3	插入视图和删除视图	214
11.1	用户自定义函数概述	181	13.3	触发器的创建和管理	214
11.1.1	用户自定义函数的特点	181	13.3.1	创建触发器	214
11.1.2	用户自定义函数的类型	181	13.3.2	管理触发器	218
11.2	创建用户自定义函数	182	13.3.3	修改、删除触发器	219
11.2.1	使用对象资源管理器	182	13.4	使用触发器实现强制业务规则	220
11.2.2	使用 CREATE FUNCTION 命令 创建用户自定义函数	183	13.4.1	INSERT 触发器	220
11.3	用户自定义函数的调用	187	13.4.2	UPDATE 触发器	221
			13.4.3	DELETE 触发器	222

13.4.4	INSTEAD OF 触发器	223	15.4	数据库恢复概述	266
13.4.5	递归触发器	225	15.4.1	系统自启动的恢复进程	266
13.4.6	嵌套触发器	225	15.4.2	用户手工恢复数据库的准备	266
13.5	使用触发器的 T-SQL 限制	225	15.5	数据库恢复	267
13.6	触发器应用实例分析	226	15.5.1	RESTORE 语句的语法格式	267
实验与思考		229	15.5.2	数据库恢复	268
第 14 章	数据库安全及访问控制	231	15.6	备份与恢复数据库实例分析	271
14.1	SQL Server 安全认证模式	231	15.6.1	用户数据库备份恢复	271
14.2	SQL Server 登录账户的管理	232	15.6.2	系统数据库恢复方法	272
14.2.1	Windows 登录账户的建立 与删除	232	实验与思考		273
14.2.2	SQL Server 登录账户建立 与删除	234	第 16 章	数据复制与转换	274
14.3	数据库访问权限的建立与删除	236	16.1	复制概述	274
14.3.1	建立用户访问数据库的权限	236	16.1.1	复制结构	274
14.3.2	删除用户访问数据库的权限	237	16.1.2	复制类型	276
14.4	角色管理	238	16.1.3	复制代理	277
14.4.1	固定服务器角色	238	16.1.4	可更新订阅	277
14.4.2	数据库角色	240	16.2	配置复制	278
14.5	数据库权限管理	246	16.2.1	创建服务器角色和分发数据库	278
14.5.1	权限种类	246	16.2.2	配置复制选项	279
14.5.2	授予权限	247	16.2.3	删除复制配置信息	279
14.5.3	禁止权限	249	16.3	创建发布出版物	280
14.5.4	取消权限	250	16.4	订阅出版物	280
14.6	安全控制设置的实例分析	251	16.5	管理复制选项	280
实验与思考		253	16.5.1	可更新的订阅选项	280
第 15 章	数据备份与恢复	255	16.5.2	筛选复制数据	282
15.1	数据备份概述	255	16.5.3	可选同步伙伴	282
15.1.1	备份策略规划	255	16.6	复制监视器	283
15.1.2	数据一致性检查	257	16.7	数据导入导出	283
15.2	备份前的准备	258	16.7.1	SQL Server 数据表数据导出	283
15.2.1	设置恢复模式	258	16.7.2	SQL Server 数据表数据导入	284
15.2.2	掌握备份设备管理	259	16.8	复制实例	284
15.3	数据库备份	261	实验与思考		286
15.3.1	BACKUP 语句的语法格式	261	附录 A	样例数据库创建及数据输入	287
15.3.2	执行数据库备份	262	参考文献		292

第1章

数据库基础和数据库设计

数据库是存储在一起被集中管理的相关数据的集合。

数据库的系统结构是对数据的三个抽象级别，它们分别是内模式、概念模式和外模式。这三级结构的差别一般很大，为了实现这三个抽象级别在内部的联系和转换，数据库管理系统在三级结构之间提供了两个层次的映像：外模式/概念模式映像、概念模式/内模式映像。这两层映像保证了数据库系统中的数据能够具有较高的逻辑独立性和物理独立性。

从最终用户角度来看，数据库系统分为单用户结构、主从式结构、客户-服务器结构和分布式结构。

本章还将介绍数据库的设计，在关系数据库方面主要介绍规范关系数据库的理论，并给出一个相应的实例。如果读者对数据库原理比较熟悉，本章内容可以略过。如果读者没有系统地接触过数据库原理，学习本章对后续章节的学习有一定的必要性。

1.1 数据库系统

数据库系统是一个比较宽泛的概念，包括数据库、数据库管理系统，以及使用数据库的用户和支撑数据库管理系统运行的软硬件。我们在此仅对与数据库系统相关的部分概念进行简单介绍，更深入的知识请读者参考相关教材或书籍。

1.1.1 数据、信息、数据库

1. 数据

数据(Data)是描述事物的符号记录，是数据库中存储的基本对象。数据可以是数值数据，如某个具体数字，也可以是非数值数据，如声音、图像等。虽然数据有多种表现形式，但经过数字化处理后，都可以输入并存储到计算机中，并能成为其处理的符号序列。

2. 信息

信息(Information)是具有一定含义的、经过加工的、对决策有价值的的数据。所以说，信息是有用的数据，数据是信息的表现形式。数据如果不具有知识性和有用性，则不能称为信息。从信息处理角度看，任何事物的属性都是通过数据来表示的，数据经过加工处理后，具有了知识性并对人类活动产生决策作用，从而形成信息。信息有如下特点：无限性、共享性、创造性。

3. 信息与数据的关系

在计算机中,为了存储和处理某些事物,需要抽象出对这些事物感兴趣的特征,组成一个记录来描述。例如,在档案中,如果人们感兴趣的是姓名、性别、出生年月、籍贯、所在系别、入学日期,就可以这样描述:(李明,男,1985,浙江,计算机系,2004),因此这里的记录就是数据。它的含义可以解释为:李明是个大学生,1985年出生,男,浙江人,2004年考入计算机系。当然,李明也可以解释为教师,2004年入职。

数据的形式不能完全表达其内容,需要经过解释。数据的解释是指对数据含义的说明,数据的含义又称为数据的语义,也就是数据包含的信息。信息是数据的内涵,数据是信息的符号表示,是载体。数据是符号化的信息,信息是语义化的数据。

4. 数据库

数据库(DataBase, DB)是长期存储在计算机内的、有组织的、可共享的数据集合。数据库中的数据按一定的数据模型组织、描述和存储,用于满足各种不同的信息需求,并且集中的数据彼此之间有相互的联系,具有较小的冗余度、较高的数据独立性和易扩展性。

1.1.2 数据库管理系统

数据库管理系统是位于用户和操作系统之间的一层数据管理软件,它的主要功能包括以下几个方面。

1. 数据定义功能

数据库管理系统提供数据定义语言 DDL,用户通过它可以方便地对数据库中的数据对象进行定义。

2. 数据操纵功能

数据库管理系统提供数据操纵语言 DML,用户可以使用操纵语言实现对数据库的基本操作,如查询、插入、删除和修改等。

3. 数据库的运行管理

数据库的建立、运行和维护由数据库管理系统统一管理和控制,以保证数据的安全性、完整性、多用户对数据的并发使用,以及发生故障后的系统恢复。

4. 数据库的建立和维护功能

数据库管理系统具有数据库初始数据的输入、转换功能,数据库转储、恢复功能,数据库的重组功能,以及性能监视、分析功能等。这些功能通常由一些实用程序完成。

1.2 数据库系统结构

1.2.1 数据库系统模式的概念

模式(Schema)是数据库中全体数据的逻辑结构和特征的描述,它仅涉及型的描述,不涉及具体的值。模式的一个具体值称为模式的一个实例(Instance)。同一个模式可以有多个实例。模式是相对稳定的,而实例是相对变动的,因为数据库中的数据是在不断更新的。模式反映的是数据的结构及其联系,而实例反映的是数据库某一时刻的状态。

1.2.2 数据库系统的三级模式结构

数据库系统结构分为三层，即内模式、概念模式（模式）和外模式，如图 1-1 所示。这个三级结构有时被称为“三级模式结构”，最早是在 1971 年的 DBTG 报告中提出的，后来被收入到 1975 年的美国 ANSI/SPARC 报告中。虽然现在 DBMS 的产品多种多样，并在不同操作系统支持下工作，但是大多数系统在总的体系结构上都具有三级模式的机构特征。

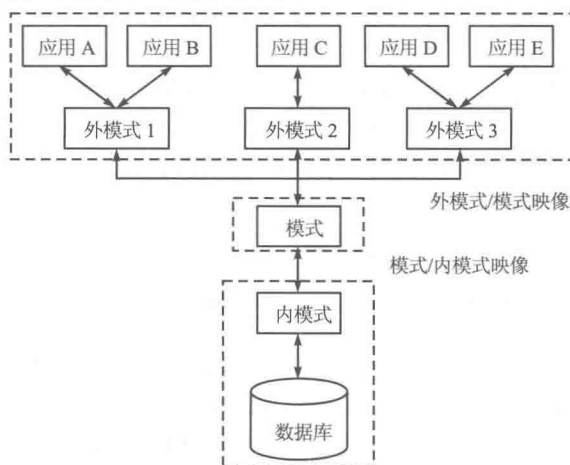


图 1-1 数据库系统的三级模式结构

从某个角度看到的数据特性称为“数据视图”（Data View）。

- 外部级最接近用户，是单个用户所能看到的数据特性，单个用户使用的数据视图的描述称为“外模式”。
- 模式涉及所有用户的数据定义，是全局的数据视图。全局数据视图的描述也称为“概念模式”。
- 内模式最接近于物理存储设备，涉及实际数据存储的结构。物理存储数据视图的描述称为“内模式”。

1. 概念模式（所有用户的公共视图）

一个数据库只有一个概念模式，它以某一种数据模型为基础，综合地考虑了所有用户的需求，并将这些需求有机地结合成一个逻辑整体。

概念模式由许多记录类型的值组成。例如，它可能包括部门记录值的集合、职工记录值的集合、供应商记录值的集合、零件记录值的集合等。模式根本不涉及物理表示和访问的技术，它只定义信息的内容，在模式中不能涉及存储字段表示、存储记录队列、索引、哈希算法、指针或其他存储和访问的细节。这样，模式即可真正实现物理数据的独立性。

定义模式时，不仅定义数据的逻辑结构，还要定义数据之间的联系，定义与数据有关的安全性、完整性要求。

在数据库管理系统（DBMS）中，描述概念模式的数据定义语言称为“模式 DDL”（Schema Data Definition Language）。

2. 外模式（用户可见的视图）

外模式也称为子模式（Subschema）或用户模式，它是数据库用户能够看见和使用的局部数据的逻辑结构和特征的描述，是数据库用户的数据视图，是与某个应用有关的数据的逻辑表示，是用户与数据库系统的接口，是用户用到的那部分数据的描述。一个系统一般有多个外模式。

外模式是保证数据库安全性的一个有力措施。每个用户只能看见和访问所对应的外模式中的数据,数据库中的其余数据是不可见的。用户使用数据操纵语言(Data Manipulation Language, DML)语句对数据库进行操作,实际上是对外模式的外部记录进行操作。

描述外模式的数据定义语言称为外模式 DDL。有了外模式后,程序员或数据库管理员不必关心概念模式,只与外模式发生联系,按照外模式的结构存储和操纵数据。

外模式又称为“用户模式”或“子模式”,通常是概念模式的逻辑子集。

3. 内模式(存储模式)

一个数据库只有一个内模式,它是数据物理结构和存储方式的描述,是数据在数据库内部的表示方法。它定义所有的内部记录类型、索引和文件的组织方式,以及数据控制方面的细节。

注意,内模式与物理层仍然不同。内部记录并不涉及物理记录,也不涉及设备的约束。比内模式更接近物理存储和访问的那些软件机制是操作系统的一部分,即文件系统。

描述内模式的数据定义语言称为内模式 DDL。

1.2.3 数据库的二级映像功能与数据独立性

数据库的三级模式结构是数据的三个抽象级别。它把数据的具体组织留给数据库管理系统(DBMS)去做,用户只要抽象地处理数据,而不必关心数据在计算机中的表示和存储。三级结构之间的差别一般很大,为了实现这三个抽象级别的内部联系和转换,DBMS在三级结构之间提供了两个层次的映像:外模式/概念模式映像、概念模式/内模式映像,见图 1-1。这两层映像保证了数据库系统中的数据能够具有较高的逻辑独立性和物理独立性。

1. 外模式/概念模式映像

外模式/模式映像用于定义外模式和概念模式之间的对应性,即外部记录和内部记录间的关系。

当模式发生改变时,由数据库管理员对各个外模式/模式的映像做相应改变,可以使外模式保持不变,应用程序是依据数据的外模式编写的,从而应用程序不必修改,保证了数据与程序的逻辑独立性,简称数据的逻辑独立性。

2. 模式/内模式映像

模式/内模式映像用于定义概念模式和内模式间的对应性,实现两级的数据结构、数据组成等的映像对应关系。

模式/内模式映像定义了数据库全局逻辑结构与存储结构之间的对应关系,当数据库的存储结构改变时,由数据库管理员对模式/内模式映像做相应的改变,可以使模式保持不变,从而应用程序也不必改变,保证了数据与程序的物理独立性,简称数据的物理独立性。

1.2.4 数据库系统用户结构

从最终用户角度来看,数据库系统分为单用户结构、主从式结构、客户-服务器结构和分布式结构。

1. 单用户数据库系统

单用户数据库系统是一种早期最简单的数据库系统,如图 1-2 所示。在这种系统中,整个数据库系统(包括应用程序、DBMS、数据)都装在一台计算机上,由一个用户独占,不同计算机之间不能共享数据。

2. 主从式结构数据库系统

主从式结构是指一个主机带多个终端的多用户结构,如图 1-3 所示。在这种结构中,数据库系统

(包括应用程序、DBMS、数据)都集中存放在主机上,所有处理任务都由主机来完成,每个用户通过主机的终端并发地存取数据库,共享数据资源。

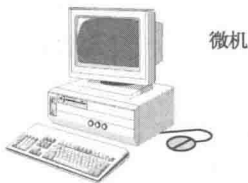


图 1-2 单用户数据库系统



图 1-3 主从式结构数据库系统

3. 客户-服务器结构数据库系统

主从式数据库系统中的主机是一个通用计算机,既执行 DBMS 功能,又执行应用程序。随着工作站功能的增强和广泛使用,人们开始把 DBMS 的功能和应用分开,网络中某个(些)结点上的计算机专门用于执行 DBMS 功能,称为数据库服务器,简称服务器。其他结点上的计算机安装 DBMS 的外围应用开发工具,支持用户的应用,称为客户机,这就是客户-服务器结构的数据库系统,如图 1-4 所示。

在客户-服务器结构中,客户端的用户请求被传送到数据库服务器,数据库服务器进行处理后,只将结果返回给用户(而不是整个数据),从而显著减少了网络上的数据传输量,提高了系统的性能、吞吐量和负载能力。另一方面,客户-服务器结构的数据库往往更加开放。客户端和服务端一般都能在多种不同的硬件和软件平台上运行,可以使用不同厂商的数据库应用开发工具,应用程序具有更强的可移植性,同时可以减少软件维护开销。

4. 分布式结构数据库系统

分布式结构是指数据库中的数据在逻辑上是一个整体,但分布在计算机网络的不同结点上。网络中的每个结点都可以独立处理本地数据库中的数据,执行局部应用,也可以同时存取和处理多个异地数据库中的数据,执行全局应用,如图 1-5 所示。它的优点是适应了地理上分散的公司、团体和组织对于数据库应用的需求;不足是数据的分布存放给数据的处理、管理与维护带来困难,当用户需要经常访问远程数据时,系统效率会明显地受到网络带宽的制约。

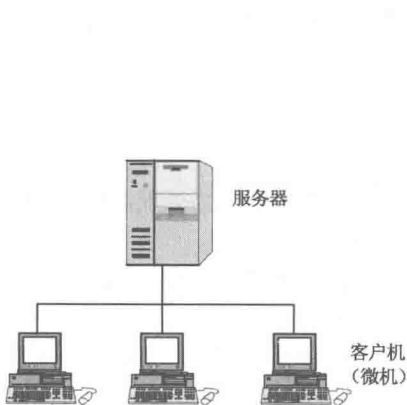


图 1-4 客户-服务器结构数据库系统

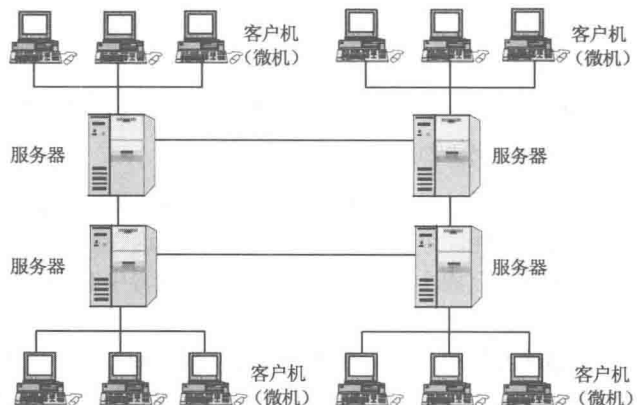


图 1-5 分布式结构数据库系统

1.3 关系数据库及其设计

1.3.1 关系数据库

在关系数据库中，一个关系就是一张二维表，由行和列组成，如图 1-6 所示。表中的一行就是一个元组（也称为记录），表中的列为一个属性，给每个属性起一个名即为其属性名（也称字段名）。

属性名	导师姓名	专业名称	学生姓名
	张清政	计算机专业	李勇
	张清政	计算机专业	刘晨
元组 (t)	刘逸	信息专业	王敏
	非码	非码	候选码 (主码)

图 1-6 关系（表）的结构

关系数据库的特点如下。

① 关系中的每个字段（属性）不可再分，是数据库中的最基本单位。

② 每一竖列字段是同属性的，每个列的顺序是任意的。

③ 每一行记录由一个事物的诸字段项构成，记录的顺序可以是任意的。

④ 不允许有相同的字段名，也不允许有相同的记录行。每个关系都有主码关键字（Key）的属性集合，用以唯一地标识关系中的各个记录行。

⑤ 解决实际问题往往需要多个关系，关系和关系是有联系的，这种联系也用关系表示。在一个给定的应用领域中，所有关系及关系之间联系的关系的集合构成一个关系数据库。

1.3.2 关系数据库设计

数据库的设计质量直接影响数据库管理系统对数据的控制质量。数据库设计是指对于一个给定的应用环境，根据用户的信息要求、处理需求和数据库的支撑环境，利用数据模型和应用程序模拟现实世界中该单位的数据结构和处理活动的过程，是数据设计和数据处理设计的结合。规范化的数据库设计要求数据库内的数据文件的数据组织应获得最大程度的共享、最小的冗余度，消除数据及数据依赖关系中的冗余部分，使依赖于同一个数据模型的数据达到有效的分离，保证在输入修改数据时数据的一致性和正确性，保证数据与使用数据的应用程序之间的高度独立性。同时，在设计时还要将数据和操作数据的行为紧密结合起来，保证数据的完整性约束。

1. 需求分析

需求分析阶段的任务是收集数据库所需要的信息内容和数据处理规则，确定建立数据库的目的。在需求分析调研中，必须与用户充分讨论，确定数据库所要进行的数据处理范围、数据处理的流程及数据取值范围的界定。

描述需求分析常用的方法有数据流图、数据字典等。

2. 概念结构设计

概念结构设计是对现实世界的一种抽象，即对实际的人、物、事和概念进行人为处理，抽取人们关心的共同特性，忽略非本质的细节，并把这些特性用各种概念精确地加以描述。

为了能够完成上述目标，我们把现实世界中客观存在并可相互区别的事物称为实体（Entity）。比如，一个职工、一个学生、一个部门、学生的一次选课、老师与系的工作关系。

把描述实体的某一特性称为属性（Attribute），一个实体可以由若干属性值来描述。比如，一个学生实体可以由“学号，姓名，性别，年龄，所在系”等属性的属性值（20021001，张三，男，21，计算机系）来描述。

同类实体中的实体彼此之间是可以区别的，能够唯一标示实体的属性集合称做实体的码或关键字。

实体集之间存在各种联系 (Relationship)，主要有三类：一对一联系 (1:1)、一对多联系 (1:n)、多对多联系 (m:n)。

(1) 一对一联系 (1:1)

对于实体集 A 中的每一个实体，实体集 B 中有 0 个或 1 个实体与之联系，反之亦然，则称实体集 A 与实体集 B 具有一对一的联系。

例如，假设一个班级只能由一个班主任（教师）管理，一个班主任也只能管理一个班，则教师与班级之间具有一对一的联系。

(2) 一对多联系 (1:n)

对于实体集 A 中的每一个实体，实体集 B 中有 0 个或多个实体与之联系，反之，对于实体集 B 中的每一个实体，实体集 A 中有 0 个或 1 个实体与之联系，则称实体集 A 与实体集 B 具有一对多的联系。

例如，一个班级有若干学生，每个学生只在一个班级中学习，则班级与学生之间具有一对多的联系。

(3) 多对多联系 (m:n)

对于实体集 A 中的每一个实体，实体集 B 中有 0 个或多个实体与之联系，反之，对于实体集 B 中的每一个实体，实体集 A 中有 0 个或多个实体与之联系，则称实体集 A 与实体集 B 具有多对多的联系。

例如，一门课程同时有若干学生选修，而一个学生同时选修多门课程，则课程与学生之间具有多对多的联系。

描述概念模型的有效工具是 E-R 模型。

3. 逻辑结构设计

关系模型的逻辑结构是一组关系模式的集合。E-R 图则是由实体、实体的属性和实体之间的联系 3 个要素组成的。所以将 E-R 图转换为关系模型，实际上就是要将实体、实体的属性和实体之间的联系转化为关系模式。这种转换一般遵循如下原则。

(1) 实体与实体属性的转换

一个实体型转换为一个关系模式。实体的属性就是关系的属性。实体的码就是关系的码。

例如，学生实体可以转换为如下关系模式，其中学号为学生关系的码：

学生（学号，姓名，性别，年龄，所在系）

(2) 实体间联系的转换

① 一个 1:1 联系可以转换为一个独立的关系模式，也可以将任意一端关系中的码合并到另一端的关系模式中。

如果转换为一个独立的关系模式，则与该联系相连的各实体的码及联系本身的属性均转换为关系的属性，每个实体的码均是该关系的候选码。

如果使用关系模式合并方式，则需要在一个关系模式的属性中加入另一个关系模式的码和联系本身的属性，而原来的码不变。

例如，假设一个班级只能由一个班主任（教师）管理，一个班主任也只能管理一个班，则教师与班级之间具有一对一的联系。将其转换为关系模式有 3 种方法。

● 转换成一个独立的关系模式：

管理（职工号，班级号）

● 将“教师”关系中的码“职工号”与“班级”关系模式合并，在“班级”关系增加“职工号”属性：

班级 (班级号, 学生人数, 职工号)

- 将“班级”关系中的码“班级号”与“教师”关系模式合并, 在“教师”关系中增加“班级号”属性:

教师 (职工号, 姓名, 性别, 职称, 班级号)

推荐使用合并的方法。

② 一个 1:n 联系可以转换为一个独立的关系模式, 也可以将一端关系中的码与 n 端对应的关系模式合并。

如果转换为一个独立的关系模式, 则与该联系相连的各实体的码及联系本身的属性均转换为关系的属性, 而关系的码为 n 端实体的码。

如果使用关系模式合并方式, 则需要在 n 端关系模式的属性中加入一端关系模式的码和联系本身的属性, 而原来的码不变。

例如, 假如有一个学生“组成”的联系, 即一个学生只能属于一个班级, 一个班级可能有多个学生, 该联系为 1:n 联系, 将其转换为关系模式有两种方法。

- 转换成一个独立的关系模式:

组成 (学号, 班级号)

- 将其与“学生”关系模式合并, 增加“班级号”属性:

学生 (学号, 姓名, 年龄, 所在系, 班级号)

推荐使用合并的方法。

③ 一个 m:n 联系转换为一个关系模式。

必须转换为一个独立关系, 与该联系相连的各实体的码及联系本身的属性均转换为新关系的属性。而关系的码为各实体码的组合。

例如, 假如有一个学生“选修”的联系, 即一个学生可以选修多门课程, 一门课程可以被多个学生选修, 该联系是一个 m:n 联系, 将其转换为如下关系模式:

选修 (学号, 课程号, 成绩)

4. 数据库表的优化与规范化

在数据需求分析的基础上, 进行概念结构和逻辑结构设计, 并将数据信息分割成数个大小适当的数据表。例如, 可以得到学生的相关数据信息 (如表 1-1 所示的学生选课表), 学生选课数据表包含学号、姓名、所在院系、电话、城市、课程编号、课程名称、成绩等属性。

表 1-1 学生选课表

学号	姓名	院系	电话	城市	课号	课名	成绩
S060101	王东民	计算机	135****	杭州	C102	C 语言	90
S060102	张小芬	计算机	131****	宁波	C102	C 语言	95
S060103	李鹏飞	计算机	139****	温州	C103	数据结构	88
S060101	王东民	计算机	135****	杭州	C103	数据结构	80
S060103	李鹏飞	计算机	139****	温州	C108	软件工程	85
S060101	王东民	计算机	135****	杭州	C106	数据库	85
S060101	王东民	计算机	135****	杭州	C108	软件工程	78
S060102	张小芬	计算机	131****	宁波	C106	数据库	80
S060109	陈晓莉	计算机	136****	西安	C102	C 语言	90
S060110	赵青山	计算机	130****	太原	C103	数据结构	92