

21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

数据库原理 及应用

Database Principles and Applications

麦中凡 何玉洁 编著

- 前端开发方法易学易用
- 后台管理技术全面翔实
- 理论和实践有机结合



精品系列

 人民邮电出版社
POSTS & TELECOM PRESS

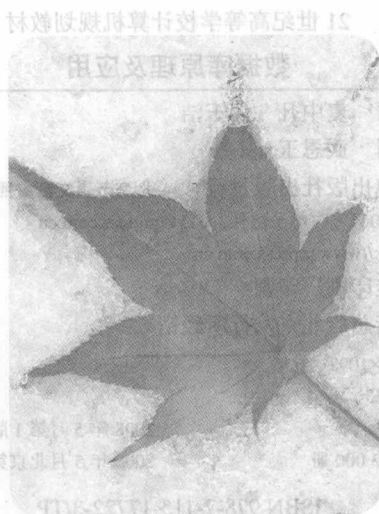
21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

数据库原理 及应用

Database Principles and Applications

麦中凡 何玉洁 编著



精品系列

人民邮电出版社

北京

图书在版编目 (CIP) 数据

数据库原理及应用 / 麦中凡, 何玉洁编著. —北京: 人民邮电出版社, 2008.5
21 世纪高等学校计算机规划教材
ISBN 978-7-115-17722-3

I. 数… II. ①麦…②何… III. 数据库系统—高等学校—教材 IV. TP311.13

中国版本图书馆 CIP 数据核字 (2008) 第 025471 号

内 容 提 要

本书主要内容包括数据库理论基础、数据库后台管理实践和数据库前端应用编程技术。数据库理论基础部分全面、系统地介绍了数据库系统的基本概念、数据库系统结构、关系数据库标准语言 (SQL)、关系数据库理论、数据库设计和数据库安全保护; 数据库后台管理实践部分以 SQL Server 2000 数据库管理系统为平台, 介绍了数据库创建、备份和恢复数据库及安全管理等实践内容; 数据库前端应用编程技术选择 Visual Basic 编程环境, 介绍了 ADO 数据控件和 ADO 对象模型的数据库编程技术。

本书注重理论和实践的统一。各章都列举大量的实例, 并附有丰富的习题。各章实验部分还包括一般论述题和上机练习题, 以帮助读者理解和掌握所学的知识。

本书内容讲解循序渐进, 深入浅出, 概念清晰, 条理性强, 符合读者学习数据库课程的认知规律。本书可作为非计算机专业本、专科学生的数据库教材。

21 世纪高等学校计算机规划教材

数据库原理及应用

- ◆ 编 著 麦中凡 何玉洁
责任编辑 武恩玉
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
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: 17.75
字数: 457 千字
印数: 1—3 000 册
- 2008 年 5 月第 1 版
2008 年 5 月北京第 1 次印刷

ISBN 978-7-115-17722-3/TP

定价: 29.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223

反盗版热线: (010)67171154



人民邮电

出版者的话

计算机应用能力已经成为社会各行业最重要的工作要求之一，而计算机教材质量的好坏会直接影响人才素质的培养。目前，计算机教材出版市场百花争艳，品种急剧增多，要从林林总总的教材中挑选一本适合课程设置要求、满足教学实际需要的教材，难度越来越大。

人民邮电出版社作为一家以计算机、通信、电子信息类图书与教材出版为主的科技教育类出版社，在计算机教材领域已经出版了多套计算机系列教材。在各套系列教材中涌现出了一批被广大一线授课教师选用、深受广大师生好评的优秀教材。老师们希望我社能有更多的优秀教材集中地呈现在老师和读者面前，为此我社组织了这套“21世纪高等学校计算机规划教材-精品系列”。

“21世纪高等学校计算机规划教材-精品系列”具有下列特点。

(1) 前期调研充分，适合实际教学需要。本套教材主要面向普通本科院校的学生编写，在内容深度、系统结构、案例选择、编写方法等方面进行了深入细致的调研，目的是在教材编写之前充分了解实际教学的需要。

(2) 编写目标明确，读者对象针对性强。每一本教材在编写之前都明确了该教材的读者对象和适用范围，即明确面向的读者是计算机专业、非计算机理工类专业还是文科类专业的学生，尽量符合目前普通高等教学计算机课程的教学计划、教学大纲以及发展趋势。

(3) 精选作者，保证质量。本套教材的作者，既有来自院校的一线授课老师，也有来自IT企业、科研机构等单位的资深技术人员。通过他们的合作使老师丰富的实际教学经验与技术人员丰富的实践工程经验相融合，为广大师生编写出适合目前教学实际需求、满足学校新时期人才培养模式的高质量教材。

(4) 一纲多本，适应面宽。在本套教材中，我们根据目前教学的实际情况，做到“一纲多本”，即根据院校已学课程和后续课程的不同开设情况，为同一科目提供不同类型的教材。

(5) 突出能力培养，适应人才市场要求。本套教材贴近市场对于计算机人才的能力要求，注重理论技术与实际应用的结合，注重实际操作和实践动手能力的培养，为学生快速适应企业实际需求做好准备。

(6) 配套服务完善，共促提高。对于每一本教材，我们在教材出版的同时，都将提供完备的PPT课件，并根据需要提供书中的源程序代码、习题答案、教学大纲等内容，部分教材还将在作者的配合下，提供疑难解答、教学交流等服务。

在本套教材的策划组织过程中，我们获得了来自清华大学、北京大学、人民大学、浙江大学、吉林大学、武汉大学、哈尔滨工业大学、东南大学、四川大学、上海交通大学、西安交通大学、电子科技大学、西安电子科技大学、北京邮电大学、中国林业大学等院校老师的大力支持和帮助，同时获得了来自信息产业部电信研究院、联想、华为、中兴、同方、爱立信、摩托罗拉等企业和科研单位的领导和技术人员的积极配合。在此，人民邮电出版社向他们表示衷心的感谢。

我们相信，“21世纪高等学校计算机规划教材-精品系列”一定能够为我国高等院校计算机课程教学做出应有的贡献。同时，对于工作欠缺和不妥之处，欢迎老师和读者提出宝贵的意见和建议。

前言

数据库技术是计算机中一个传统的基础技术,它为人们提供了科学和高效的管理数据的方法,利用数据库技术可以方便地实现数据操作、安全控制和可靠性管理等功能。数据库的这些功能是通过一系列服务提供的,人们要在特定的应用平台上使用这些功能,同时需要有相应的操作语言。关系数据库是通过 SQL 实现的,SQL 提供了方便地检索、操作和管理数据库数据的功能,它不但可以在特定的应用平台上使用,而且还可以嵌入到程序设计语言中。本书就是从数据库理论基础、数据库管理实践和数据库应用编程 3 个方面全面阐述了数据库技术的应用体系。

本书主要由 4 篇组成,第一篇(第 1 章至第 7 章)介绍了数据库系统的基本概念和基本理论,具体内容包括:数据管理的发展过程、数据库系统的组成结构、SQL 基础及数据定义功能、视图、索引、关系规范化理论、数据库保护及数据库设计几个方面。这一部分是数据库的理论基础。第二篇(第 8 章至第 11 章)介绍服务器端的数据库管理系统的功能,这里以 SQL Server 2000 数据库管理系统为例,介绍其主要功能以及在此系统对数据库理论的实现,具体包括安装和配置 SQL Server 以及在此环境中创建数据库和表、进行安全管理以及备份和恢复的技术。这部分作为第一部分的实践应用。第三篇(第 12 章至第 14 章)介绍在客户端编写数据库应用程序的方法。这部分以 Visual Basic 6.0 环境为例,介绍了数据库应用程序编写方法和数据访问接口技术,主要介绍了使用 ADO 数据控件以及 ADO 对象访问数据库的方法,同时介绍了一些常用的数据绑定控件的使用。考虑到有些学生可能没有 Visual Basic 基础,因此在本篇的开始简要介绍了 Visual Basic 的编程环境和编程基础知识,以便于这些学生顺利学习本书。本书第四篇(第 15 章)主要介绍了数据库技术的发展过程以及发展方向。

本书的最大特点是内容全面,立足学以致用,在内容选取上既包括了数据库的基础理论知识,又包括了数据库的客户端和服务端的应用技术。同时在确定前、后台软件平台时,考虑到了流行性和易获得性。SQL Server 2000 是目前应用范围广泛且非常典型的数据库管理系统,它不像 Oracle 那样复杂,同时也不像 MySQL 那样功能不够完善。Visual Basic 也是使用非常普及、方便的可视化编程环境,这两个软件很容易获得,而且对计算机的配置要求也比较低,一般学校的计算机都能满足要求,因此,非常方便学生上机实践。

本书可作为非计算机专业多用计算机(电子、通信、管理及信息处理)学科大学本科教材。随着计算机软硬件技术的不断发展,各企业和部门管理水平的不断提高和规范化,计算机的应用水平取得了长足的进步,特别是数据库技术,其应用水平及普及速度更是日新月异,数据库技术已经不再仅是计算机专业学生必须学习的课程,业已成为非计算机专业大学生需要学习和掌握的知识。作为新时代的大学生,为了能够适应社会对人才的需要,有必要全面地掌握数据库知识。

本书适合 42~52 学时的教学,各学校和各专业可根据对学生的计算机水平掌握程度的要求,在授课中对本书的一些内容进行筛选。对数据库知识要求偏低的学校和专业,可省略第 5 章“关系数据库规范化理论”和第 14 章“ADO 对象编程”部分,并将整个第四篇作为选讲内容。

为便于教师使用本书进行教学,本书提供电子教案和习题解答,有需要者请登录人民邮电出版社(www.ptpress.com.cn)免费下载。

本书主要由何玉洁执笔,同时北京航空航天大学麦中凡教授对本书提出了很多卓有成效的建议,并制定了全书的总体大纲。在本书编写过程中,得到了人民邮电出版社的大力支持和帮助,他们对本书的内容提出了许多宝贵的意见和建议,是他们一直以来的鼓励帮助我完成此书。在此,对人民邮电出版社的全体人员表示诚挚的感谢。同时感谢我的同仁和朋友:张俊超、李迎、何青、张宏旭、于小倩,他们分别参与了本书第三篇和第四篇的部分编写工作,还绘制了很多图表,同时对全书也提出了很多很好的建议和意见,是他们的积极参与和帮助,使本书得以顺利完成。

编者从事数据库教学多年,并一直致力于数据库技术及应用的研究,在数据库方面有一些自己的经验和感受,本书是作者这些年的经验和感受的一些总结。虽然作者尽了自己应尽的努力,但由于本人水平所限,书中难免有不妥之处,望广大同仁能给予批评和指正。

编 者

2008 年 2 月

目 录

第一篇 数据库理论基础

第 1 章 数据库概述..... 2

1.1 数据管理的发展..... 2

1.1.1 文件管理..... 2

1.1.2 数据库管理..... 5

1.2 数据独立性..... 7

1.3 数据库系统的组成..... 7

1.4 数据库应用结构..... 8

1.4.1 集中式结构..... 8

1.4.2 文件服务器结构..... 9

1.4.3 客户/服务器结构..... 9

1.4.4 互联网应用结构..... 10

1.5 小结..... 11

习题..... 11

第 2 章 数据模型与数据库系统

结构..... 12

2.1 数据和数据模型..... 12

2.1.1 数据..... 12

2.1.2 数据模型..... 13

2.2 概念层数据模型..... 14

2.2.1 基本概念..... 14

2.2.2 实体-联系模型..... 14

2.3 组织层数据模型..... 17

2.3.1 关系模型的数据结构..... 17

2.3.2 关系模型的数据操作..... 19

2.3.3 关系模型的数据完整性

约束..... 19

2.4 数据库系统结构..... 21

2.4.1 模式的基本概念..... 21

2.4.2 三级模式结构..... 22

2.4.3 数据库的模式映像功能与

数据独立性..... 24

2.4.4 数据库管理系统..... 24

2.5 小结..... 25

习题..... 26

第 3 章 SQL 语言基础及数据定义

功能..... 27

3.1 SQL 语言概述..... 27

3.1.1 SQL 语言的发展..... 27

3.1.2 SQL 语言的特点..... 28

3.1.3 SQL 语言功能概述..... 28

3.2 SQL 语言支持的数据类型..... 29

3.2.1 数值型..... 29

3.2.2 字符串型..... 30

3.2.3 日期时间类型..... 30

3.2.4 货币类型..... 31

3.3 数据定义功能..... 31

3.3.1 基本表..... 32

3.3.2 数据完整性约束..... 35

3.4 索引..... 39

3.4.1 基本概念..... 39

3.4.2 索引分类..... 39

3.4.3 创建和删除索引..... 40

3.5 小结..... 41

习题..... 41

第 4 章 数据操作..... 43

4.1 数据查询功能..... 43

4.1.1 查询语句的基本结构..... 43

4.1.2 简单查询..... 44

4.1.3 多表连接查询..... 56

4.1.4 使用 TOP 限制结果集..... 60

4.1.5 子查询..... 62

4.2 数据更改功能..... 68

4.2.1 插入数据..... 68

4.2.2 更新数据..... 69

4.2.3 删除数据	70	6.3.1 概念结构设计	92
4.3 视图	71	6.3.2 逻辑结构设计	95
4.3.1 基本概念	71	6.3.3 物理结构设计	98
4.3.2 定义视图	71	6.4 数据库行为设计	100
4.3.3 通过视图查询数据	73	6.4.1 功能分析	101
4.3.4 修改和删除视图	75	6.4.2 功能设计	101
4.3.5 视图的作用	76	6.4.3 事务设计	102
4.4 小结	77	6.5 数据库实施	102
习题	77	6.6 数据库的运行和维护	103
第5章 关系数据库规范化理论	79	6.7 小结	104
5.1 函数依赖	79	习题	104
5.1.1 基本概念	79	第7章 数据库保护	106
5.1.2 一些术语和符号	80	7.1 事务	106
5.1.3 为什么讨论函数依赖	81	7.1.1 事务的基本概念	106
5.2 关系规范化	82	7.1.2 事务的特征	107
5.2.1 关系模式中的键	82	7.1.3 事务处理模型	107
5.2.2 范式	83	7.2 并发控制	108
5.3 小结	86	7.2.1 并发控制概述	108
习题	86	7.2.2 并发控制措施	110
第6章 数据库设计	87	7.2.3 封锁协议	111
6.1 数据库设计概述	87	7.2.4 活锁和死锁	113
6.1.1 数据库设计的特点	88	7.2.5 并发调度的可串行性	115
6.1.2 数据库设计方法概述	88	7.2.6 两段锁协议	116
6.1.3 数据库设计的基本步骤	89	7.3 数据库备份与恢复	117
6.2 数据库需求分析	90	7.3.1 数据库故障的种类	117
6.2.1 需求分析的任务	90	7.3.2 数据库备份	118
6.2.2 需求分析的方法	91	7.3.3 数据库恢复	119
6.3 数据库结构设计	92	7.4 小结	120
		习题	120

第二篇 SQL Server 基础及应用

第8章 SQL Server 2000 基础

8.1 SQL Server 2000 概述	122
8.2 安装与测试	123
8.2.1 安装前的准备	123
8.2.2 安装及安装选项	124
8.2.3 安装 SQL Server 2000 中	

常见问题的解决	129
8.2.4 测试安装	129
8.3 SQL Server 2000 常用工具简介	130
8.3.1 企业管理器	131
8.3.2 查询分析器	131
8.4 T-SQL 简介	133
8.4.1 脚本	133

8.4.2 注释	134	10.2 SQL Server 的安全控制	163
8.4.3 变量	134	10.3 管理登录账户	165
8.4.4 流程控制语句	135	10.3.1 系统内置的登录账户	166
8.5 小结	139	10.3.2 建立用户登录账户	166
习题	139	10.3.3 修改登录账户的属性	167
上机练习	140	10.3.4 删除登录账户	168
第 9 章 数据库及数据库对象的创建与管理	141	10.4 管理数据库用户	168
9.1 数据库的创建与管理	141	10.5 管理权限	169
9.1.1 SQL Server 数据库分类	141	10.5.1 SQL Server 权限种类	169
9.1.2 SQL Server 数据库基本概念	142	10.5.2 权限的管理	170
9.1.3 创建数据库	143	10.6 角色	172
9.1.4 删除数据库	147	10.6.1 固定的服务器角色	173
9.1.5 修改数据库	147	10.6.2 固定的数据库角色	174
9.2 基本表的创建与管理	148	10.6.3 用户定义的角色	175
9.2.1 创建表及约束	149	10.7 小结	177
9.2.2 修改表结构	153	习题	177
9.2.3 删除表	154	上机练习	178
9.3 索引的创建及管理	154	第 11 章 备份和恢复数据库	180
9.3.1 创建索引	154	11.1 备份数据库	180
9.3.2 查看和删除索引	156	11.1.1 为什么进行数据备份	180
9.4 视图的创建与管理	156	11.1.2 备份内容及备份时间	180
9.4.1 创建视图	156	11.1.3 SQL Server 的备份设备	181
9.4.2 查看和修改视图	158	11.1.4 SQL Server 支持的备份类型	181
9.5 小结	159	11.1.5 备份策略	183
习题	160	11.1.6 实现备份	184
上机练习	160	11.2 恢复数据库	188
第 10 章 安全管理	162	11.2.1 恢复前的准备	189
10.1 安全控制	162	11.2.2 恢复的顺序	189
10.1.1 安全控制模型	162	11.2.3 实现恢复	190
10.1.2 数据库权限的种类及用户的分类	163	11.3 小结	193
		习题	194
		上机练习	194
第三篇 数据库应用程序开发			
第 12 章 Visual Basic 开发环境及数据访问接口	198		
12.1 Visual Basic 6.0 概述	198	12.1.1 Visual Basic 6.0 的特点	198
		12.1.2 Visual Basic 6.0 的安装与启动	199

12.1.3	Visual Basic 集成开发环境	200
12.2	Visual Basic 工程设计的基本步骤	204
12.2.1	新建工程	205
12.2.2	设计用户界面	205
12.2.3	编写代码	206
12.2.4	保存工程	206
12.2.5	运行和调试工程	207
12.2.6	生成可执行文件	207
12.3	数据绑定控件	208
12.3.1	数据绑定控件概述	208
12.3.2	DataGrid 控件	208
12.3.3	DataList 和 DataCombo 控件	210
12.4	数据访问接口	211
12.4.1	ODBC	211
12.4.2	OLE DB 和 ADO	213
12.5	小结	215
	习题	215
	上机练习	216

第 13 章 ADO 数据控件与数据

窗体向导

13.1	ADO 数据控件	217
13.1.1	ADO 数据库控件的主要属性、方法和事件	218
13.1.2	Recordset 对象的主要属性和方法	221
13.2	ADO 数据控件使用示例	223
13.2.1	示例 1	223
13.2.2	示例 2	224

13.2.3	示例 3	227
13.3	使用数据窗体向导	231
13.4	小结	235
	习题	235
	上机练习	235

第 14 章 ADO 对象编程

14.1	概述	237
14.2	Connection 对象	238
14.2.1	Connection 对象的主要属性和方法	238
14.2.2	使用 Connection 对象	239
14.3	Command 对象	239
14.3.1	Command 对象的主要属性和方法	239
14.3.2	使用 Command 对象	240
14.4	Recordset 对象	240
14.4.1	Recordset 的属性和方法	241
14.4.2	使用 Recordset 对象	244
14.5	Field 对象	244
14.6	Parameter 对象	245
14.7	Error 对象	246
14.8	使用 ADO 对象访问数据库一般步骤	247
14.9	示例	248
14.9.1	示例 1	248
14.9.2	示例 2	252
14.10	小结	254
	习题	255
	上机练习	255

第四篇 数据库技术的发展

第 15 章 数据库技术的发展

15.1	数据库技术的发展	258
15.1.1	传统数据库技术的发展历程	258
15.1.2	新一代数据库管理系统	259
15.2	面向对象技术与数据库技术的	

结合	260
15.2.1 新的数据库应用和新的数据类型	260
15.2.2 面向对象数据模型	261
15.2.3 面向对象数据库的优点	264
15.2.4 对象关系数据库与对象数据库	265

15.3 数据库技术面临的挑战	265	15.4.3 多媒体数据库	267
15.4 数据库技术的研究方向	267	15.4.4 数据库中的知识发现	268
15.4.1 分布式数据库系统	267	15.4.5 专用数据库系统	268
15.4.2 面向对象的数据库管理 系统	267	15.5 小结	268
		参考文献	270

第一篇

数据库理论基础

本篇主要介绍数据库的基础理论知识，包括数据库系统概念、关系数据库的标准操作语言——SQL、关系数据库理论、数据库的安全性和完整性、事务的并发控制以及数据库的分析和设计技术。

本篇组成

- 第1章 数据库概述
- 第2章 数据模型与数据库系统结构
- 第3章 SQL语言基础及数据定义功能
- 第4章 数据操作
- 第5章 关系数据库规范化理论
- 第6章 数据库设计
- 第7章 数据库保护

第 1 章

数据库概述

随着信息管理水平的不断提高,应用范围的日益扩大,信息已成为企业的重要财富和资源,同时,作为管理信息的数据库技术也得到了很大的发展,其应用领域越来越广泛。人们在不知不觉中扩展着对数据库的使用,比如信用卡购物,飞机/火车订票系统、图书馆对书籍及借阅的管理等,无一不使用了数据库技术。从小型事务处理到大型信息系统,从联机事务处理到联机分析处理,从一般企业管理到计算机辅助设计与制造(CAD/CAM)、地理信息系统等,数据库系统已经渗透到我们日常生活中的方方面面,数据库中信息量的大小以及使用的程度已经成为衡量企业的信息化程度的重要标志。

简单地说,数据库技术就是研究如何对数据进行科学的管理,以提供可共享、安全、可靠的数据。数据库技术一般包含数据管理和数据处理两部分。

数据库系统本质上是一个用计算机存储数据的系统,数据库本身可以看作是一个电子文件柜,也就是说,数据库是收集数据文件的仓库或容器。

1.1 数据管理的发展

自计算机产生之后,人们就希望利用它来帮助我们进行数据的存储和管理。最初对数据的管理是以文件方式进行的,也就是通过编写应用程序来实现对数据的存储和管理。后来,随着数据量越来越大,人们对数据的要求越来越多,希望达到的目的也越来越复杂,文件管理方式已经难以满足人们对数据的需求,由此产生了数据库系统,也就是用数据库系统来存储和管理数据。数据管理的发展因此也就经历了文件管理和数据库管理两个阶段。

本节介绍文件管理和数据库管理在管理数据上的主要差别。

1.1.1 文件管理

理解今日数据库特征的最好办法,是了解数据库技术产生之前,人们是如何通过文件的方式对数据进行管理的。

20 世纪 50 年代后期到 60 年代中期,计算机的硬件方面已经有了磁盘等直接存取的存储设备,软件方面,操作系统中已经有了专门的数据管理软件,一般称为文件管理系统。文件管理系统把数据组织成相互独立的数据文件,利用“按文件名访问,按记录进行存取”的管理技术,可以对文件中的数据进行修改、插入和删除操作。

在出现程序设计语言之后,开发人员不但可以创建自己的文件并将数据保存在文件中,而且

还可以编写应用程序来处理文件中的数据,即编写应用程序来定义文件的结构,实现对文件内容的插入、删除、修改和查询操作,当然,真正实现磁盘文件的物理存取操作的还是操作系统中的文件管理系统,应用程序只是告诉文件管理系统对哪个文件的哪些数据进行哪些操作。我们将开发人员编写应用程序并借助文件管理系统的功能实现对用户数据的处理的方式称为文件管理,在本章后面的讨论中将忽略掉文件管理系统,假定应用程序是直接对磁盘文件进行操作。

现在,看一下文件管理方式下对数据的操作模式。假设某学校要构建以文件管理的方式对学生进行管理的系统。在此系统中主要实现两部分功能:学生基本信息管理和学生选课情况管理。假设教务部门管理学生选课情况,各系部管理学生基本信息。学生基本信息管理中涉及学生的基本信息数据,假设这些数据保存在 F1 文件中;学生选课情况管理涉及学生的部分基本信息、课程基本信息和学生选课信息,文件 F2 和 F3 分别保存课程基本信息和学生选课信息的数据。

设 A1 为实现“学生基本信息管理”功能的应用程序,A2 为实现“学生选课管理”功能的应用程序。由于学生选课管理中要用到 F1 文件中的一些数据,为减少冗余,它将使用“学生基本信息管理”(即 F1 文件)中的数据,如图 1-1 所示。

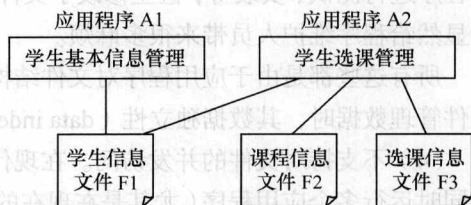


图 1-1 文件管理系统示例

文件 F1、F2 和 F3 分别包含如下信息。

F1 文件——学号、姓名、性别、出生日期、联系电话、所在系、专业、班号。

F2 文件——课程号、课程名、授课学期、学分、课程性质。

F3 文件——学号、姓名、所在系、专业、课程号、课程名、选课类型、选课时间、考试成绩。

我们将文件中所包含的每一个子项称为文件结构中的“字段”或“列”,将每一行数据称为一个“记录”。

“学生选课管理”的处理过程大致为:在学生选课管理中,若有学生选课,则先查 F1 文件,判断有无此学生;若有则再访问 F2 文件,判其所选的课程是否存在;若一切符合规则,就将学生选课信息写到 F3 文件中。

这看似很好,但仔细分析一下,就会发现用文件管理数据有如下缺点。

(1) 编写应用程序不方便。应用程序编写者必须清楚地了解所用文件的逻辑及物理结构,如文件中包含多少个字段,每个字段的数据类型,采用何种存储结构(比如链表或数组等)。操作系统只提供了打开、关闭、读、写等几个底层的文件操作命令,而对文件的查询、修改等处理都必须在应用程序中编程实现。这样就容易造成各应用程序在功能上的重复,比如图 1-1 中的“学生基本信息管理”和“学生选课管理”都要对 F1 文件进行操作,而共享这两个功能相同的操作却很难。

(2) 数据冗余不可避免。由于 A2 应用程序需要在学生选课信息文件(F3 文件)中包含学生的一些基本信息,比如学号、姓名、所在系、专业等,而这些信息同样包含在学生信息文件(F1 文件)中,因此 F3 文件和 F1 文件中存在重复数据,从而造成数据的重复,也称数据的冗余。

数据冗余所带来的问题不仅仅是存储空间的浪费(其实,随着计算机硬件技术的飞速发展,存储容量不断扩大,空间问题已经不是我们关注的主要问题),更为严重的是造成了数据的不一致(inconsistency)。例如,某个学生所学的专业发生了变化,我们一般只会想到在 F1 文件中进行修改,而往往忘记了在 F3 中应做同样的修改。由此就造成了同一名学生在 F1 文件和 F3 文件中的“专业”不一样,也就是数据不一致。人们不能判定哪个数据是正确的,尤其是当系统中

存在多处数据冗余时,更是如此。这样数据就失去了其可信性。

文件本身并不具备维护数据一致性的功能,这些功能完全要由用户(应用程序开发者)负责维护。这在简单的系统中还可以勉强应付,但在复杂的系统中,若让开发者来保证数据的一致性,几乎是不可能的。

(3) 应用程序依赖性。就文件管理而言,应用程序对数据的操作依赖于存储数据的文件的结构。文件和记录的结构通常是应用程序代码的一部分,如 C 程序的 struct。文件结构的每一次修改,比如添加字段、删除字段,甚至修改字段的长度(如电话号码从 7 位扩到 8 位),都将导致应用程序的修改,因为在打开文件进行数据读取时,必须将文件记录中不同字段的值对应到应用程序的变量中。随着应用环境和需求的变化,修改文件的结构不可避免,这些都需要在应用程序中做相应的修改,而(频繁)修改应用程序是很麻烦的。人们首先要熟悉原有程序,修改后还需要对程序进行测试、安装等;甚至修改了文件的存储位置或者文件名,也需要对应用程序进行修改,这显然给程序维护人员带来很多麻烦。

所有这些都是由于应用程序对文件结构以及文件物理特性的过分依赖造成的,换句话说,用文件管理数据时,其数据独立性(data independence)很差。

(4) 不支持对文件的并发访问。在现代计算机系统中,为了有效利用计算机资源,一般都允许同时运行多个应用程序(尤其是在现在的多任务操作系统环境中)。文件最初是作为程序的附属数据出现的,它一般不支持多个应用程序同时对同一个文件进行访问。回忆一下,某个用户打开了一个 Excel 文件,当第二个用户在第一个用户未关闭此文件前打开此文件时,会得到什么信息呢?他只能以只读方式打开此文件,而不能在第一用户打开的同时对此文件进行修改。再回忆一下,如果用某种程序设计语言编写一个对某文件中内容进行修改的程序,其过程是先以写的方式打开文件,然后写入新内容,最后关闭文件。在关闭文件之前,不管是在其他的程序中,还是在同一个程序中都不允许再次打开此文件,这就是文件管理方式不支持并发访问的含义。

对于以数据为中心的系统来说,必须要支持多个用户对数据的并发访问。

(5) 数据间联系弱。当用文件管理数据时,文件与文件之间是彼此独立、毫不相干的,文件之间的联系必须通过程序来实现。比如对上述的 F1 文件和 F3 文件, F3 文件中的学号、姓名等学生的基本信息必须是 F1 文件中已经存在的(即选课的学生必须是已经存在的学生);同样, F3 文件中课程号等与课程有关的基本信息也必须存在于 F2 文件中(即学生选的课程也必须是已经存在的课程)。这些数据之间的联系是实际应用当中所要求的很自然的联系,但文件本身不具备自动实现这些联系的功能,我们必须编写应用程序,即手工地建立这些联系。这不但增加了编写代码的工作量和复杂度,而且当联系很复杂时,也难以保证其正确性。因此,用文件管理数据时很难反映现实世界事物间客观存在的联系。

(6) 难以满足不同用户对数据的需求。不同用户会关注不同的数据。有些用户(如分配学生宿舍的人)可能只关心学生的部分基本信息(学号、姓名、性别)。

若多个不同用户希望看到的是学生不同的基本信息,那就需要为每个用户建立一个文件,这势必造成很多的数据冗余。我们希望的是,用户关心哪些信息就为他生成哪些信息,对用户不关心的数据将其屏蔽。

另有一些用户可能需要来自于多个不同文件的部分信息的组合,例如班号、学号、姓名、课程名、学分、考试成绩等。这些信息涉及三个文件:从 F1 文件中得到“班号”,从 F2 文件中得到“学分”,从 F3 文件中得到“考试成绩”;而“学号”、“姓名”可以从 F1 文件或 F3 文件中得到,“课程名”可以从 F2 文件或 F3 文件中得到。在生成结果数据时,必须对从三个文件中读取

的数据进行比较,然后组合成一行有意义的数。比如,将从 F1 文件中读取的学号与从 F3 文件中读取的学号进行比较,学号相同时,才可以将 F1 文件中的“班号”与 F3 文件中的当前记录所对应的学号和姓名组合起来,之后,还需要将组合结果与 F2 文件中的内容进行比较,找出课程号相同的课程的学分,再与已有的结果组合起来。然后再从组合后的数据中提取出用户需要的列信息。如果数据量很大,涉及的文件比较多时,我们可以想象这个过程有多复杂。因此,这种大容量复杂信息的查询,在文件管理方式中是很难处理的。

(7) 无安全控制功能。在文件管理方式中,很难控制某个人对文件的操作,比如只允许某个人查询和修改数据,但不能删除数据,或者对文件中的某个或者某些字段不能修改等。而在实际应用中,数据的安全性是非常重要的且不可忽视的。比如,在学生选课管理中,我们不允许学生修改其考试成绩。在银行系统中,更是不允许一般用户修改其存款数额。

人们对数据需求的增加,迫切需要对数据进行有效、科学、正确、方便的管理。针对文件管理的这些缺陷,人们逐步开发了以统一管理和共享数据为主要特征的数据库管理系统。

1.1.2 数据库管理

20 世纪 60 年代后期以来,计算机管理数据的规模越来越大,应用范围越来越广泛,数据量急剧增加,同时多种应用同时共享数据集合的要求也越来越强烈。

随着大容量磁盘的出现,硬件价格的不断下降,软件价格的不断上升,编制和维护系统软件 and 应用程序的成本相应的不断增加。在数据处理方式上,联机实时处理要求更多,并开始提出和考虑分布处理。在这种背景下,以文件方式管理数据已经不能满足应用的需求,于是出现了新的管理数据的技术——数据库技术,同时出现了统一管理数据的专门软件——数据库管理系统。

例如,对于上述的学生基本信息管理和学生选课管理两个子系统,如果使用数据库技术来管理,其实现方式与文件管理有很大区别。用数据库技术来管理的实现过程如图 1-2 所示。



图 1-2 数据库管理实现示例

比较图 1-1 和图 1-2,可以直观地发现两者有如下差别。

- 在文件管理中,应用程序是直接访问存储数据的文件;而在数据库管理中,应用程序则是通过数据库管理系统(DataBase Management System, DBMS)来访问数据。

- 在数据库管理中,用户不再是对每一个文件进行数据访问,而是访问一个存储了用户所需全部信息的数据库,存储数据的具体文件的信息被数据库隐藏了,而且这些文件的具体操作和存储位置等细节信息也由数据库管理系统统一管理。

与文件管理相比,数据库管理实际上是在应用程序和存储数据的数据库(在某种意义上也可以把数据库看成是一些文件的集合)之间增加的一层,即数据库管理系统,数据库管理系统是一个系统软件。不要小看这个变化,正是因为有了这个系统软件,才使得开发人员避免了编写实现复杂数据操作等功能的程序,应用程序或者用户不再需要关心数据的存储方式。数据的存储方式由数据库管理系统进行处理,其变化不再影响应用程序,应用程序也不需要任何修改。

与文件管理相比,数据库管理具有以下优点。

(1) 相互关联的数据集成。在数据库系统中,所有相关的数据都存储在一个称作数据库的环境中,应用程序可通过 DBMS 访问数据库中的所有数据。

(2) 较少的数据冗余。由于数据是统一管理的,因此可以从全局着眼,合理地组织数据。例

如，将 1.1.1 节中文件 F1、F2 和 F3 的重复数据挑选出来，进行合理的管理，这样就可以形成如下所示的几部分信息。

学生基本信息：学号、姓名、性别、出生日期、联系电话、所在系、专业、班号。

课程基本信息：课程号、课程名、授课学期、学分、课程性质。

学生选课信息：学号、课程号、修课类型、修课时间、考试成绩。

在关系数据库中，可以将每一种信息存储在一个表中（关系数据库的概念将在后边介绍），重复的信息只存储一份，当在学生选课中需要学生的姓名等其他信息时，根据学生选课中的学号，可以很容易地在学生基本信息中找到此学号对应的姓名等信息。因此，消除数据的重复存储不影响对信息的提取，同时还可以避免由于数据重复存储而造成的数据不一致问题。比如，当某个学生所学的专业发生变化时，只需在“学生基本信息”一个地方进行修改即可。

同 1.1.2 节中的问题一样，当要检索（班号，学号，姓名，课程名，学分，考试成绩）信息时，这些信息也需要从 3 个地方（关系数据库为 3 张表）得到，也需要对信息进行适当的组合，即学生选课中的学号只能与学生基本信息中学号相同的信息组合在一起，同样，学生选课中的课程号也必须与课程基本信息中的课程号相同的信息组合在一起。过去在文件管理方式中，这个工作是由开发者编程实现的，而现在有了数据库管理系统，这些繁琐的工作完全交给了数据库管理系统来完成。

因此，在数据库管理系统中，减少数据冗余不会增加开发者的负担。

（3）程序与数据相互独立。在数据库中，数据所包含的所有数据项以及数据的存储格式都与数据存储在—起，它们通过 DBMS 而不是应用程序来访问和管理，应用程序不再需要处理文件和记录的格式。

程序与数据相互独立有两方面的含义。一方面当数据的存储方式发生变化时（这里包括逻辑存储方式和物理存储方式），比如从链表结构改为散列表结构，或者是顺序和非顺序之间的转换，应用程序不必作任何修改。另一方面当数据的逻辑结构发生变化时，比如增加或减少了一些数据项，如果应用程序与这些修改的数据项无关，则不用修改应用程序。这些变化都将由 DBMS 负责维护，大多数情况下，应用程序并不知道数据存储方式或数据项已经发生了变化。

（4）保证数据的安全可靠。数据库技术能够保证数据库中的数据是安全可靠的。它的安全控制机制可以有效地防止数据库中的数据被非法使用或非法修改；其完整的备份和恢复机制可以保证当数据遭到破坏时（由软件或硬件故障引起的）能够很快地将数据库恢复到正确的状态，并使数据不丢失或只有很少的丢失，从而保证系统能够连续、可靠地运行。

（5）最大限度地保证数据的正确性。数据的正确性（也称作数据的完整性）是指存储到数据库中的数据必须符合现实世界的实际情况，比如人的性别只能是“男”和“女”，人的年龄应该在 0 到 150 之间（假设没有年龄超过 150 岁的人）。如果在性别中输入了其他值，或者将一个负数输入到年龄中，在现实世界中显然是不对的。数据的正确性是通过在数据库中建立约束来实现的。当建立好保证数据正确性的约束之后，如果有不符合约束的数据存储到数据库中，数据库能主动拒绝这些数据。

（6）数据可以共享并能保证数据的一致性。数据库中的数据可以被多个用户共享，即指允许多个用户同时操作相同的数据。当然，这个特点是针对大型的多用户数据库系统而言的，对于单用户系统，在任何时候最多只有一个用户访问数据库，因此不存在共享的问题。

多用户共享问题是数据库管理系统内部解决的问题，它对用户是不可见的。这就要求数据库能够对多个用户进行协调，保证多个用户之间对数据的操作不会产生矛盾和冲突，即在多个用户