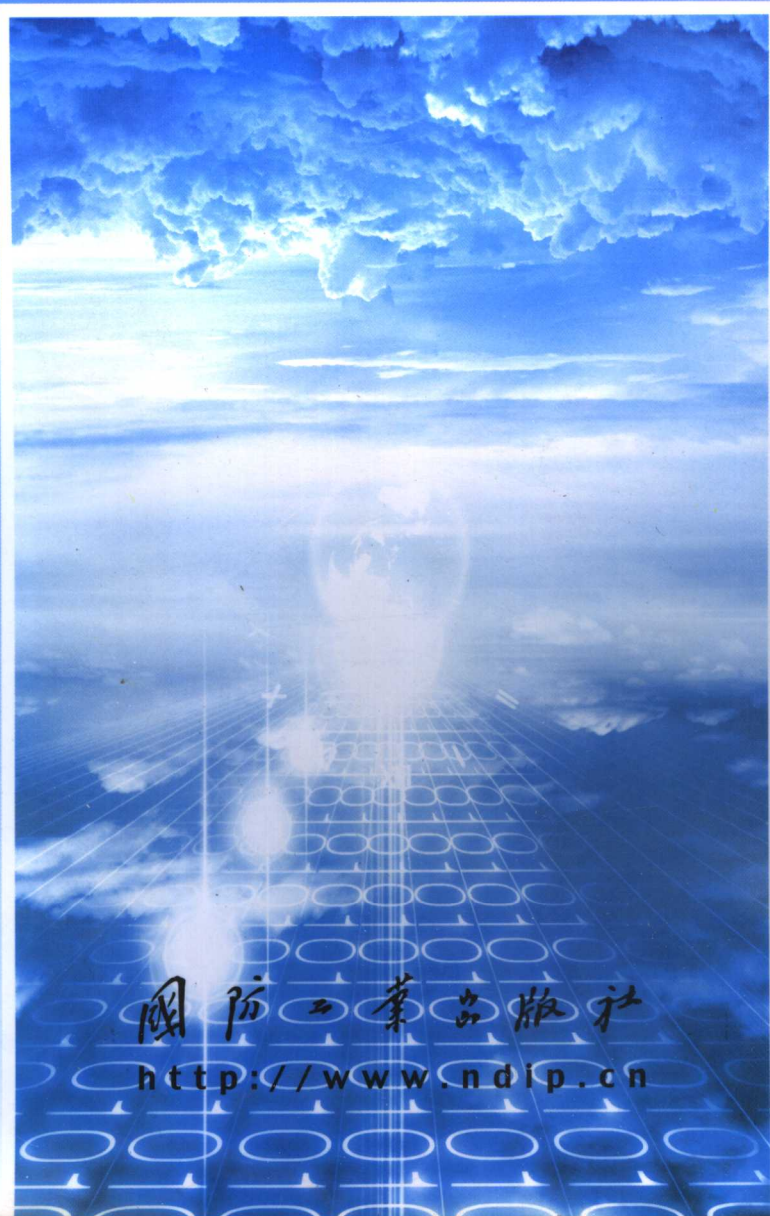


主编 王志梅 副主编 黄 河

关系数据库 基础与技术



国防工业出版社

<http://www.ndip.cn>

11.138SQ

关系数据库基础与技术

主 编 王志梅
副主编 黄 河
主 审 李永平

国防工业出版社

· 北京 ·

内 容 简 介

本书从数据库系统的应用着手,深入浅出地介绍了数据库基本原理,并且全面而系统地介绍了 SQL 语言的使用。主要包括:数据库概述、实体—联系模型、关系模型、关系数据库设计理论、SQL 语言初步、SQL 语言高级功能、关系数据库的实现技术及数据库新技术等。

本书着重于以实例引入概念,且对概念的阐述以够用为度,注重实际技能的引导与培养。

本书是一本针对高职高专的信息管理专业、计算机技术与应用专业及相关专业数据库课程的教材,也可作为数据库爱好者的参考书。

图书在版编目(CIP)数据

关系数据库基础与技术/王志梅主编. —北京:国防工业出版社,2005.1

ISBN 7-118-03683-8

I. 关... II. 王... III. 关系数据库—数据库管理系统 IV. TP311.138

中国版本图书馆 CIP 数据核字(2004)第 113664 号

国防工业出版社出版发行

(北京市海淀区紫竹院南路 23 号)

(邮政编码 100044)

新艺印刷厂印刷

新华书店经售

*

开本 787×1092 1/16 印张 13 293 千字

2005 年 1 月第 1 版 2005 年 1 月北京第 1 次印刷

印数:1—4000 册 定价:18.00 元

(本书如有印装错误,我社负责调换)

国防书店: 68428422

发行邮购: 68414474

发行传真: 68411535

发行业务: 68472764

前 言

数据库系统的出现可追溯到 20 世纪 60 年代。目前数据已经由最初的层次与网状数据库系统发展到关系数据库系统,并正在向面向对象数据库系统发展。随着数据库系统理论和数据库产品的日益成熟,数据库技术在信息产业的发展中起到了越来越重要的作用。

本书是一本针对高职高专计算机或相关专业数据库课程的教材。全书共 8 章,有*标记的为选讲内容。

第 1 章 数据库概论。从数据库系统的应用着手,介绍了数据库、数据库管理系统、数据库系统的基本概念,讨论了数据库的结构及数据模型等基本概念。

第 2 章 实体—联系模型。介绍了实体—联系(E-R)数据模型的一些主要要素:实体、属性和联系,E-R 模式的映射基数。在实体集与联系集中如何确定主码,如何将实体集、联系集以及相关的属性通过 E-R 图表示出来。

第 3 章 关系模型。系统地讲解了关系数据库的重要概念,包括关系模型的数据结构、关系代数基本运算以及关系的完整性。

第 4 章 关系数据库设计理论。首先从关系模式可能的存储异常问题,引入函数依赖的概念,然后介绍了以函数依赖为基础的关系范式,包括 1NF、2NF、3NF,最后简单介绍了数据库设计的方法与过程。

第 5 章 SQL 语言初步。从 SQL 语言的基本概念出发,首先讲述了基本的查询语句,接着引入了可以完成更加复杂的查询操作,如聚集查询、连接查询、集合查询及子查询等。

第 6 章 SQL 语言高级功能。在第 5 章的基础上,继续就 SQL 语言进行了学习。本章介绍的内容,如约束的定义、触发器的管理、定义游标的技术及安全性管理等,都是 SQL 语言的精华。

第 7 章 关系数据库的实现技术。重点介绍了两种数据库的核心实现技术:索引与散列、事务与并发控制。

第 8 章 数据库新技术。概略地介绍了当前新型的数据库系统。

本书在较全面地介绍数据库系统基本原理的基础上,重点讨论了 SQL 语言的使用。SQL 语言是关系数据库的标准语言,是学好关系数据库的前提。

本书的特点:

- (1) 以实例引入概念,对理论的阐述少而精,以够用为度。
- (2) 注重实用性且语言通俗易懂。
- (3) 整本书以学生管理系统实例为基础进行讲解,便于学生理解。

建议教学实践环境为 SQL Server 2000。

IV

本书由王志梅主编，黄河副主编，李永平主审。第1章、第7章、第8章由黄河编写，第2章由曹月芹编写，第3章由谢梅源编写，第4章、第5章、第6章由王志梅编写。

由于作者水平有限，时间仓促，书中难免有不妥之处，望广大读者指正。

作 者

2004年7月

目 录

第 1 章 数据库概述	1
1.1 数据库系统的应用	1
1.1.1 应用实例	1
1.1.2 数据库系统的应用前景	2
1.2 数据库相关概念	2
1.3 数据管理技术的发展	4
1.3.1 文件管理系统	4
1.3.2 数据库管理系统	6
1.4 数据库系统结构	7
1.5 数据模型	8
1.5.1 数据模型及分类	8
1.5.2 数据模型的 3 个要素	9
1.6 本章小结	10
习题	10
第 2 章 实体—联系模型	11
2.1 实体—联系模型的基本要素	11
2.1.1 实体与实体集	11
2.1.2 实体型和实体值	12
2.1.3 属性	12
2.1.4 联系与联系集	14
2.1.5 码	17
2.2 实体—联系图表示	19
2.2.1 E-R 符号表示	19
2.2.2 E-R 图的表示	20
2.3 E-R 模型的设计	23
2.3.1 确定实体集、属性与实体间的联系	24
2.3.2 具有复合属性、多值属性和派生属性的 E-R 图	26
2.3.3 具有弱实体集的 E-R 图	26
2.4 *E-R 图的高级技术	27
2.4.1 特殊化	28
2.4.2 概括	29
2.4.3 属性继承	30

2.5 E-R 模型的设计实例.....	30
2.5.1 E-R 图设计步骤.....	30
2.5.2 大学教学情况 E-R 图设计.....	30
2.5.3 银行企业的 E-R 图设计.....	34
2.6 本章小结.....	37
习题.....	38
第 3 章 关系模型	40
3.1 关系基本结构及术语.....	40
3.2 关系模型的数据操作.....	42
3.2.1 关系操作.....	42
3.2.2 关系代数.....	42
3.3 关系的完整性约束.....	52
3.3.1 实体完整性.....	52
3.3.2 参照完整性.....	53
3.3.3 自定义完整性.....	56
3.4 将 E-R 模型转换为关系模型.....	56
3.5 关系模型的特点.....	61
3.6 本章小结.....	61
习题.....	61
第 4 章 关系数据库设计理论	64
4.1 冗余和存储异常问题.....	64
4.2 函数依赖.....	67
4.2.1 属性间的联系.....	67
4.2.2 函数依赖.....	68
4.3 关系范式.....	70
4.3.1 第一范式.....	70
4.3.2 第二范式.....	71
4.3.3 第三范式.....	72
4.4 关系范式的规范化.....	73
4.4.1 各范式之间的关系.....	73
4.4.2 关系模式的分解准则.....	74
4.4.3 关系模式分解实例.....	74
4.5 关系数据库设计.....	75
4.5.1 数据库设计概述.....	75
4.5.2 数据库设计的基本步骤.....	75
4.5.3 数据库设计简单示例.....	77
4.6 本章小结.....	80
习题.....	80
第 5 章 SQL 语言初步	82

5.1 SQL 基本概念.....	82
5.1.1 SQL 语言标准.....	83
5.1.2 SQL 语言的特点.....	83
5.1.3 SQL 语言功能概述.....	84
5.2 SQL 的数据类型.....	84
5.2.1 数值型.....	84
5.2.2 字符数据类型.....	85
5.2.3 日期时间类型.....	86
5.2.4 货币类型.....	87
5.2.5 SQL 的标识符与关键字.....	87
5.3 基本查询语句.....	88
5.3.1 查询语句的基本结构.....	89
5.3.2 投影.....	90
5.3.3 选择.....	95
5.3.4 对查询结果进行排序.....	100
5.4 高级查询语句.....	102
5.4.1 聚集查询.....	102
5.4.2 连接查询.....	107
5.4.3 集合查询.....	113
5.4.4 子查询.....	114
5.4.5 关于引用 AS 指定的名字的规则.....	122
5.5 数据操作语句.....	122
5.5.1 插入数据.....	122
5.5.2 删除数据.....	123
5.5.3 更新数据.....	124
5.6 数据定义语句.....	126
5.6.1 定义基本表.....	126
5.6.2 定义视图.....	129
5.7 本章小结.....	132
习题.....	133
第 6 章 SQL 语言高级功能.....	136
6.1 约束.....	136
6.1.1 NOT NULL 约束.....	136
6.1.2 DEFAULT 约束.....	137
6.1.3 PRIMARY KEY 约束.....	138
6.1.4 UNIQUE 约束.....	139
6.1.5 FOREIGN KEY 约束.....	140
6.1.6 CHECK 约束.....	142
6.1.7 约束的作用对象.....	143

6.2 触发器	144
6.2.1 触发器的概念	144
6.2.2 触发器的结构	144
6.2.3 触发器示例	146
6.3 游标	149
6.3.1 游标的概念	149
6.3.2 定义游标	149
6.3.3 游标示例	150
6.4 嵌入式 SQL	151
6.4.1 接口与声明	151
6.4.2 嵌入式 SQL 的实现	152
6.5 安全控制	153
6.5.1 数据库的安全性控制	154
6.5.2 SQL 语言中的安全性控制	155
6.6 本章小结	158
习题	158
第 7 章 关系数据库的实现技术	160
7.1 索引与散列	160
7.1.1 索引的基本概念	160
7.1.2 有序索引	161
7.1.3 散列索引	165
7.1.4 SQL 中的索引定义	166
7.2 事务与并发控制	166
7.2.1 事务	167
7.2.2 SQL 中的事务控制语句	170
7.2.3 并发控制	170
7.3 本章小结	176
习题	176
第 8 章 数据库新技术	178
8.1 基于对象的数据库	178
8.1.1 关系模型的局限	178
8.1.2 面向对象数据模型	179
8.1.3 对象—关系数据库	181
8.2 XML	182
8.2.1 XML 数据结构	182
8.2.2 XML 文档模式和 XML 查询	184
8.2.3 XML 与数据库	184
8.3 数据仓库与数据挖掘	185
8.3.1 数据仓库 (Data Warehousing)	186

8.3.2 OLAP.....	188
8.3.3 数据挖掘(Data Mining).....	189
8.3.4 实现	191
8.4 其他新技术.....	192
8.5 本章小结	193
习题	193
附录	194
参考文献	198

第 1 章 数据库概述

本章学习目标

- 理解与数据库相关的概念及数据库的作用
- 了解数据库技术的演变
- 理解数据库体系的 3 个层次
- 掌握数据模型概念、分类及数据模型的 3 个要素

随着信息管理水平的不断提高，信息资源已成为社会各行各业的重要资源和财富，用于信息管理的数据库技术也得到了很大的发展，其应用领域也越来越广泛。

现在，数据库已经成为我们每天生活中不可缺少的一部分，虽然有人可能并没有意识到这一点。为了方便讨论，我们认为“数据库”是相关数据的集合，而“数据库管理系统”（Database Management System, DBMS）是管理和控制对数据库进行访问的软件，更概括的术语——“数据库系统”，是指与数据库相关的应用程序的集合。设计数据库系统的目的是为了管理大量的信息。在本章里，我们将简要介绍数据库系统的基本原理。

1.1 数据库系统的应用

1.1.1 应用实例

数据库系统的应用非常广泛。以下是一些具有代表性的典型示例，通过这几个典型的示例，能够了解到数据库技术到底可以起什么作用，也使我们对这门信息技术的支撑技术有一个感性的认识。

1. 银行业务系统

银行在数据库中存储各个客户的信息、账户、贷款和交易。你到 ATM 上取款实际上是这样一个过程：在 ATM 上插入信用卡，并且输入密码，ATM 会连接到银行数据库去查询密码以进行核对。在确认了你是合法用户后，ATM 就允许你进行存款取款或者其他操作。当你决定要取出一定数额的现金并且告诉 ATM 后，ATM 会到银行数据库中检查你所取的金额是否超出了你的账户余额以及你的信用额度，如果没有超出，ATM 就在银行数据库中修改你的账户信息和添加本次交易的信息，然后付给你现金。

2. 铁路售票系统

铁路公司利用数据库进行售票管理。当你去火车站或任意一个售票点去购买火车票时，售票应用程序首先连接到数据库中，检查是否还有你所要的车次的余票，如果有，就修改数据库中的信息：把某个车厢的某个座位标识为已经出售。然后把火车票打印出

来给你。当然还有一种比较复杂的情况，当某个车次只剩下一张票了，两个售票窗口同一时间发现有一张余票，并且都开始尝试订购这最后一张票。我们的数据库系统不得不处理这种情况，必须保证某个车厢的某个座位只能出售一次，所以系统要通知一个窗口订购成功，通知另一个窗口订购失败。

3. 超市购物系统

到超市购物也用到数据库。收银员使用条形码阅读器来扫描你所购买的每种货物，收银台程序使用条形码从货物数据库中查询该项货物的价格，并在收银机上显示价格，然后根据购买量修改库存数量。更好的系统会监测该项产品库存数量是否低于某个最低极限值，该系统可能会自动设置一张订单来订购此种产品以增加库存。

4. 图书管理系统

到图书馆借阅图书，同样离不开数据库。图书馆一般会有一个数据库，其中的保存的信息包括图书的信息、读者的信息、借阅的信息等。它允许读者基于作者、出版社、书名或者其他信息查找所需书籍。通过条形码输入器，图书馆的数据库系统处理借阅、续借、还书等操作。

5. 学生选课系统

学校同样有数据库的应用，比如构建一个学生选课的数据库系统。简单来说，这个系统中应该保存学生的基本信息（学号、姓名、班级等）、课程的信息（课程号、课程名、课时等），还有选课信息（此信息描述各个学生选了哪些课程以及所获成绩）。学生可以在此系统进行选课，实际上就是往数据库中添加一些选课信息。教师可以在系统中登记学生所获成绩，学生可以查询自己的选课情况和课程成绩。

上面仅仅是几种数据库系统应用，很显然，你会发现我们周围还有许多其他应用情况。尽管我们平时都熟知这些应用，也在使用它们，但是它们的背后，却隐藏着复杂的高级技术。下面我们开始学习这些技术。

1.1.2 数据库系统的应用前景

在当今的信息时代中，我们的生活越来越多地依赖信息的存取和使用，数据库系统正日益广泛地应用到人们的生活中。我们可以使用数据库访问银行账户信息，从而使存取钱更方便、快捷；在股票交易中，使用数据库可以很方便地将钱从银行户头转换到股票户头等。

信息需求的增长使数据库系统的应用日益重要，范围日益广泛，数据库和数据库管理系统正在探寻前所未有的应用领域。目前，数据库系统已经应用到医学监控、医学诊断、计算机辅助设计、计算机辅助制造、计算机辅助工程、能源管理、图书馆管理、航空系统、天气预报、交通预订、旅馆预订等许多领域。

数据库系统的发展满足了用户共享信息的需求，随着在线信息的增加以及越来越多的用户希望访问在线信息，今后还会开发出更多的面向应用的数据库系统。

1.2 数据库相关概念

本节将为数据库技术中的一些术语提供正式的定义。

1. 数据库

数据库 (Database) 是逻辑上相关的可共享的数据 (以及数据的描述) 集合。用于处理组织所需的信息。

下面详细分析数据库的定义, 以便全面理解这个概念。

1) 可共享的数据集合

数据库是一个巨大的数据存储地, 可以同时被许多部门和用户使用。这些用户所需要的所有数据用最小冗余集成在一起。更重要的是, 数据库通常不属于任何一个部门或用户, 而是组织内的共享资源。

2) 数据的描述

除了存储组织的数据, 数据库也存储数据的描述, 因此, 数据库也定义为完整记录的自描述集合。数据的描述, 也就是元数据 (关于数据的数据), 被称为数据字典。正是数据库的自描述特征提供了数据独立性的特点。这就意味着如果有新的数据结构要加入到数据库中, 或者要修改数据库中已经存在的数据结构, 那么, 如果应用程序不直接依赖于被修改的部分, 则对该使用该数据库的应用程序来说没有影响。例如: 在 1.1 节的最后的选课系统的例子中, 如果我们增加一种新的数据结构 (一张新的表) 来存储教师的基本信息, 由于这种新的数据结构和旧的选课应用程序没有直接关系, 所以并不影响原来的选课应用程序的使用。但是, 如果修改了一个已存在应用程序使用的数据结构 (比如删除了一列), 那么这个应用程序会受到这个变化的影响, 因此也必须进行相应的修改。

3) 逻辑相关

当我们分析了组织中的信息需求后, 我们就要标识数据库所描述的对象和它们之间的逻辑关系。例如: 在 1.1 节的最后的选课系统的例子中, 我们分析了学校需要存储和管理学生选课信息, 发现数据库至少要描述两个对象: 学生和课程; 而且还需要描述这两个对象之间的逻辑关系: 选课——一个学生可以选择多门课程, 一门课程可以被多个学生所选择。

2. 数据库管理系统

数据库管理系统 (DBMS) 是用户可以定义、创建和维护数据库以及提供对数据库有限制的访问的软件系统。

DBMS 是在用户、应用程序和数据库之间发挥作用的软件。DBMS 允许用户从数据库中插入、更新、删除和检索数据。

图 1.1 说明了数据库的访问形式。它表明学生和教师使用他们的应用程序, 通过 DBMS 对数据库进行访问。应用程序处理数据入口、数据维护和生成报表。数据的物理结构和存储由 DBMS 管理。

3. 数据库系统

数据库系统就是带有数据库的计算机系统, 包括 5 部分内容。

- (1) 数据库 (DB)。
- (2) 数据库管理系统 (DBMS)。
- (3) 应用 (Application): 使用数据库的应用程序。
- (4) 人员: 包括普通用户和数据库管理员 (Database Administrator, DBA)。数据库

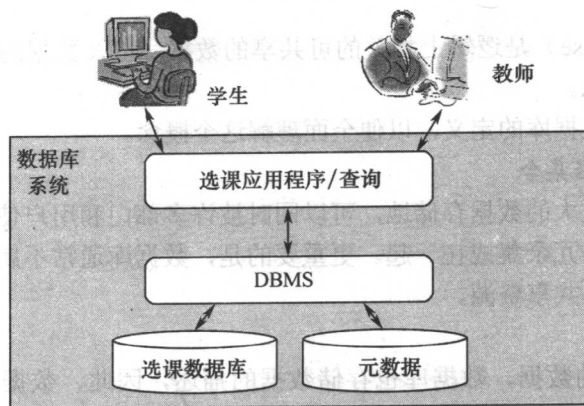


图 1.1 选课系统使用 DBMS

是整个组织的数据资源，因此组织设立了专门的数据资源管理机构来管理数据库，DBA 则是这个机构的一组人员，负责全面地管理和控制数据库管理系统。

(5) 硬件。

1.3 数据管理技术的发展

数据管理是数据库的核心任务，数据管理技术的发展，是随着计算机硬件技术、软件技术及数据处理的需求的发展而不断发展的。硬件、软件和数据，推动了数据库技术从传统的文件管理阶段向数据库管理阶段的演变。

要想很好地理解现代数据库的特征，最好先看一下在数据库技术产生之前，人们是如何保存和使用数据的。

1.3.1 文件管理系统

早期的数据是采用文件系统进行管理的，即将数据保存在文件中。用户的应用程序直接操作文件中的数据。在文件系统中，数据按其内容、结构和用途分成若干命名的文件。文件一般为某一用户或用户组所有，但也可以指定与其他用户共享。用户可以通过操作系统对文件进行打开、读、写、关闭等操作。

下面将更加详细地讨论 1.1 节中学生选课的例子。选课系统要保存学生信息、课程信息以及选课信息。如果要在计算机中保存这些信息，首先想到是，保存在操作系统的文件中。比如创建一个文件保存学生信息，定义好每个学生的信息占用 100 个字，学生学号占用这 100 个字节中的第 0 个~第 9 个字节，姓名占用第 10 个~第 19 个字节……各个属性依次排列下来。这样就可以编写应用程序访问这个文件，由于已经知道各个属性的排列规律，所以可以正确检索、修改、添加、删除学生信息。

随着需求的增加，新的应用程序将会加入系统中。比如，我们要提供教师任课情况的管理，我们就要建立一个新的永久性文件来保存教师信息，进而就需要管理教师应用程序加入系统，如图 1.2 所示。所以随着时间的推移，越来越多的文件和应用程序会加入到系统中。

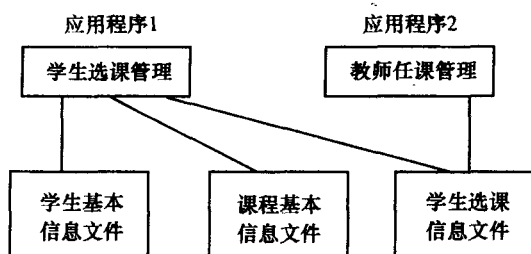


图 1.2 文件管理系统示例

以上所描述的典型的文件处理系统是传统的操作系统所支持的。永久记录被存储在多个不同的文件中，人们编写不同的应用程序来将记录从适当的文件中取出或加入到适当的文件中。在 DBMS 出现之前，各个组织通常采用这种系统来存储信息。

在文件系统中存储信息的主要弊端包括以下几方面。

1. 数据冗余和不一致

由于文件和程序是在很长的一段时间内由不同的程序员创建的，因此，不同的文件可能采用不同的格式。此外，相同的信息可能在不同文件中重复存储，比如，学生班级可能既在学生信息文件中存储也在选课信息文件中存储，这种冗余不仅使存储和访问的开销变大，还可能导致数据不一致性，即同一数据的不同副本不一致。比如一个学生转班了，我们如果仅仅在学生信息文件中进行更改，而没有在选课信息文件中进行更改，就会出现这种不一致的情况。

2. 数据访问困难

假设某个老师想找出某门课程到底有哪些学生选，但是原来的系统没有这样的功能，只有列出所有课程的选课情况的功能。这时，这个老师有两种选择：一是使用原来的程序产生所有课程的选课情况，再手工选择出这门课程的选课情况；另一种是编写新的应用程序来完成这项工作。显然这两种选择都不是很好。如果这个老师编写了相应的程序完成这项工作，几天后，这个老师又要统计没有选课的学生信息，可以预见，这个老师又不得不再一次面临前面这两种不尽人意的选择。传统的文件系统不支持以一种方便而有效的方式去获取所需数据。

3. 数据孤立

由于数据分散在不同的文件中，而这些文件又可能具有不同的格式，因此编写一个检索适当数据的新的应用程序是很困难的。

4. 完整性问题

数据库中所存储数据的值必须满足某些特点的一致性约束。例如，学生的年龄必定是大于 0 的。开发者会在应用程序加入适当的代码来进行约束检查。如果有新的约束加入时，我们就必须修改应用程序代码来实现这些约束，而这点往往是比较困难的。尤其是这种约束涉及多个文件的多个数据项时，情况会变的更加复杂。

5. 原子性问题

和 CD 唱机或汽车一样，计算机系统也会发生故障。一旦故障发生，数据就应该可以被恢复到故障发生以前的一致状态。对于很多应用来说，这样的保证是至关重要的。

银行应用中的转账操作是个经典例子。让我们进行一个操作：把 A 账户的 1000 元转入到 B 账户。这样的转账操作分两个步骤：A 账户减去 1000 元和 B 账户增加 1000 元。如果在这个过程中发生了故障，很可能只完成了第一个步骤而来不及完成第二个步骤，这样就造成了数据的不一致：1000 元钱就这样消失了。显然，为了保证数据的一致性，这里的借和贷两个步骤必须是要么都发生要么都不发生。也就是说，转账这个操作必须是原子的——要么全部发生要么根本不发生。事实上在文件系统中，保持原子性是很难做到的。

6. 并发访问异常

为了提高系统的总体性能和加快响应速度，许多系统允许多个用户同时更新数据。在这样的环境中，并发的更新操作相互影响，可能导致数据的不一致。假设 A 账户中有 1000 元，假如同时有两个客户从此账户中取款，一个取出 500 元，一个取出 300 元。假设取款操作过程是：取出原账户的余额，在其上减去取款额，再将结果写回。如果这两个取款操作并行执行，可能它们读到的余额都是 1000 元，并将分别写回 500 元和 700 元。账户中余额到底是 500 元还是 700 元就要看到底哪个操作先写回了。但是，很显然这两个结果都是错误的，正确的余额应该是 200 元。为了消除这种情况发生的可能，系统必须进行某种形式的管理。由于文件可能被多个程序访问，而这些程序又是独立的，管理就很难进行。

7. 安全性问题

并非所有的用户都可以访问所有的数据。例如，在选课系统中，学生只能查看选课信息和课程信息，他们应该没有权力查看教师信息。但是，在文件系统中，这样的安全性约束很难实现。

以上问题以及一些其他问题，加速了 DBMS 的发展。数据库系统将可以更好地解决上述在文件中存在的问题。

1.3.2 数据库管理系统

数据库技术的出现主要是为了克服文件管理系统在管理数据上的诸多缺陷，满足人们对数据管理的需求。对于上述的学生基本信息管理和学生选课管理系统来说，如果作用数据库来管理，其实现方式与文件系统有很大区别，这从图 1.1 可以很明显地看出来。这里再列出对图 1.2 所示例子使用数据库进行管理的示例，如图 1.3 所示。

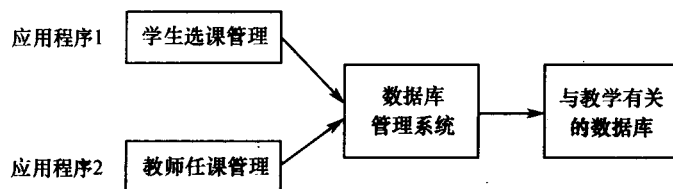


图 1.3 数据库管理系统实现示例

比较一下图 1.2 与图 1.3，可以发现两者有如下差别。

(1) 在文件管理系统中，应用程序直接访问存储数据的文件；而在数据库系统中，应用则是通过数据库管理系统来访问数据。

(2) 在数据库系统中, 数据不再仅仅为某个程序或用户服务, 存储数据的文件也不需要直接被应用程序管理, 而是由一个称为数据库管理系统的软件统一管理。

与文件系统管理数据比较后, 会发现数据库系统具有如下优点。

- (1) 将相互关联的数据集成在一起。
- (2) 较少的数据冗余。
- (3) 程序与数据相互独立。
- (4) 保证数据的安全可靠。
- (5) 最大限度地保证数据的正确性。
- (6) 数据可以共享并能保证数据的一致性。

数据库技术发展到今天已经成为一门比较成熟的技术了, 其具备如下特征。

数据库是相互关联的数据的集合, 它用综合的方法组织数据, 具有较小的数据冗余, 可供多个用户共享, 具有较高的数据独立性, 具有安全控制机制, 能够保证数据的安全性, 允许并发地使用数据库, 能有效、及时地处理数据, 并能保证数据的一致性和完整性。

需要强调的是, 所有这些特征并不是数据库中的数据所固有的, 而是由数据库管理系统提供和保证的。

1.4 数据库系统结构

数据库系统的主要目的是给用户提供了数据的抽象视图。数据库利用复杂的数据结构来表示数据, 为了简化用户的使用, 系统隐藏了关于数据存储和维护的某些细节。为了实现这种抽象, 一般来说, 数据库系统使用三级模式的结构。这3个级别代表了不同抽象层次。

1. 物理层

也称为内模式。抽象的最低层次, 描述数据实际上是怎样存储的。物理层详细描述复杂的低层数据结构。

2. 逻辑层

也称为模式。比物理层稍高的抽象层次, 描述数据库中存储什么数据及这些数据间存在什么关系。这样, 逻辑层就通过少量相对简单的结构描述了整个数据库。虽然简单的逻辑层结构的实现涉及到复杂的物理层结构, 但逻辑层的用户不必知道这样的细节。逻辑层的抽象是被数据库管理员所使用的, 管理员必须确定数据库中应该保存哪些信息。

3. 视图层

也称为外模式或子模式。抽象的最高层次, 只描述整个数据库的某个部分。尽管在逻辑层使用了比较简单的结构, 但由于数据库规模巨大, 因此仍存在一定程度的复杂性。数据库系统的很多用户并不需要关心所有的信息, 而只需要访问数据库的一部分。视图层抽象的定义正是为了使这样的用户和系统的交互变得更简单。系统可以为同一数据库提供多个视图。

在学生选课系统的例子中, 我们把学生信息、选课信息、课程信息、教师信息分别存放在4张表中, 而且分别定义每种信息包含什么内容, 比如学生信息包括学号、姓名、