HZ BOOKS
华章教育

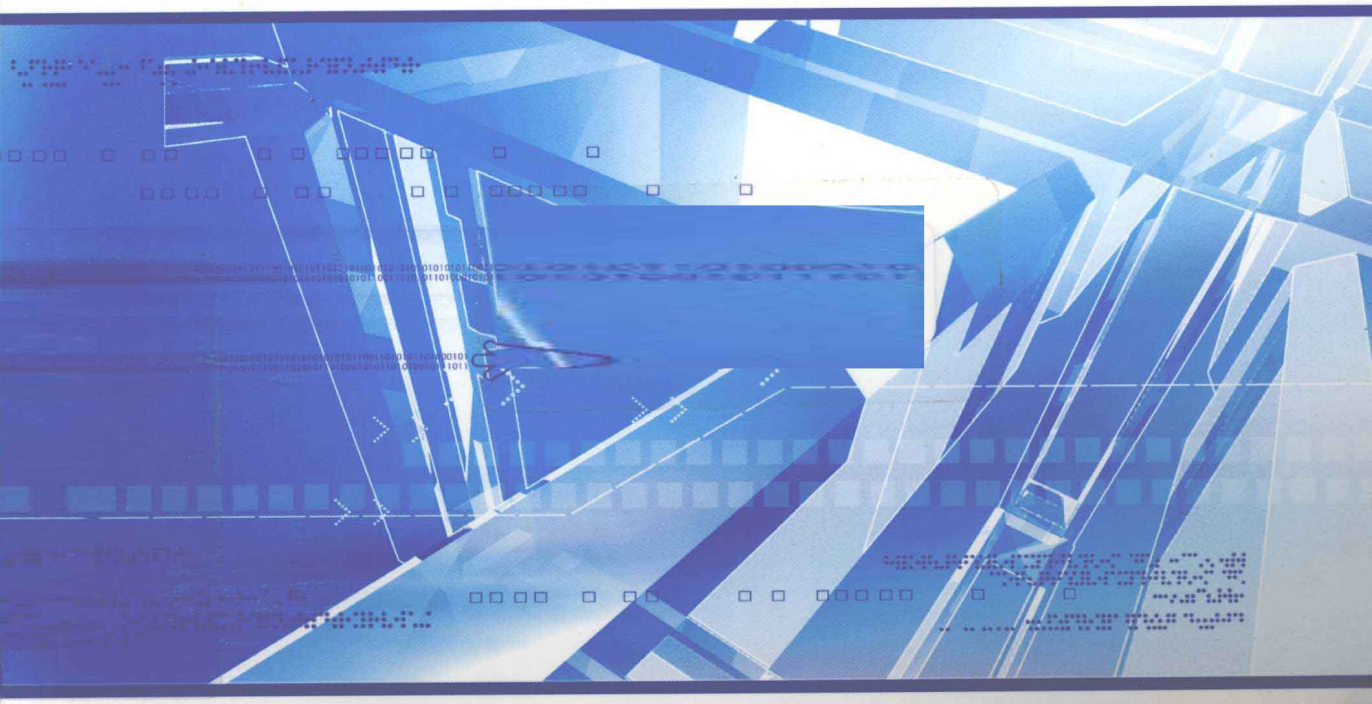
高等院校计算机教材系列

数据库原理与应用

Database Theory and Application

第2版

何玉洁 梁琦 等编著



机械工业出版社
China Machine Press

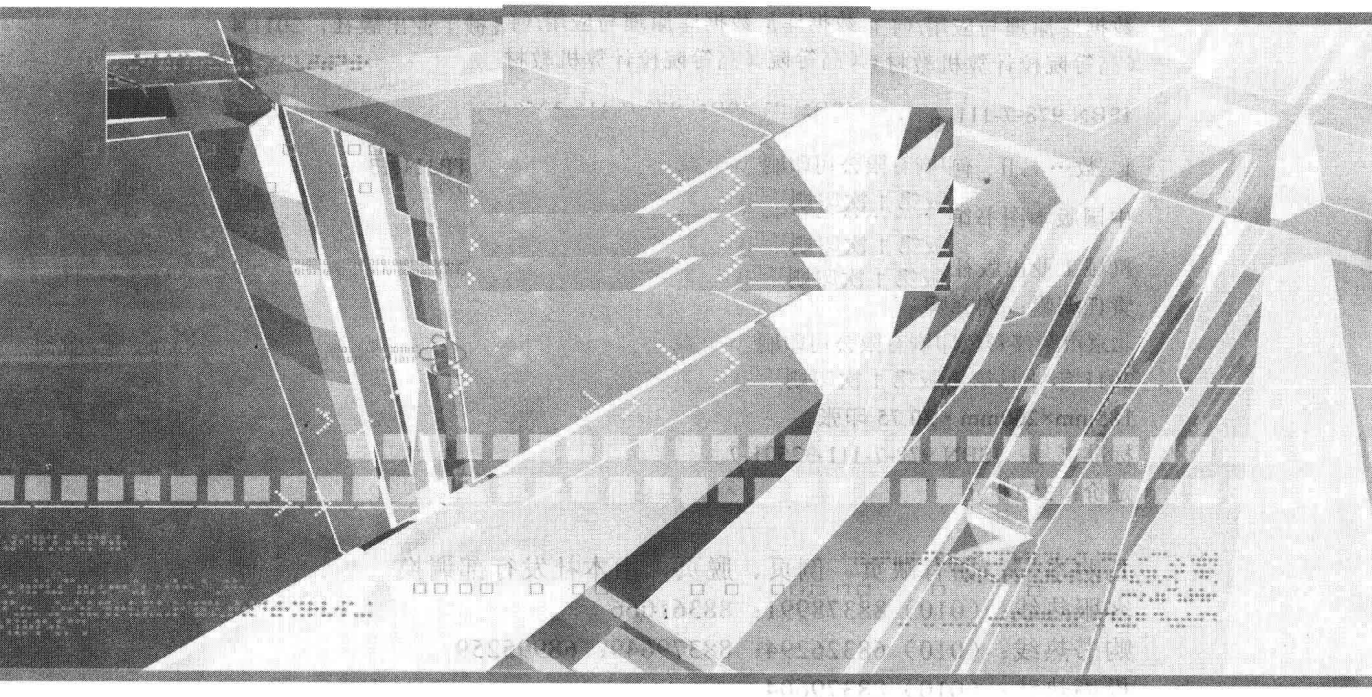
高等院校计算机教材系列

数据库原理与应用

Database Theory and Application

第2版

何玉洁 梁琦 等编著



机械工业出版社
China Machine Press

数据库技术是一门应用性很强的学科，因此在讲授数据库技术时应该从理论和应用两方面来介绍。本书正是本着这个宗旨做到了理论和应用相结合。

本书由三部分组成，第一部分偏重于数据库理论，主要介绍的是关系数据库理论；第二部分偏重于数据库服务器端的管理和编程，包括创建数据库、存储过程、触发器、安全管理、备份和恢复等，本部分选用的是 SQL Server 2005；第三部分偏重于客户端数据库应用编程，介绍了用 ASP.NET 编写 B/S 架构的数据库应用程序的基本技术。

本书内容全面、实例丰富，并为教师配备了电子教案，方便教学。本书可作为高等院校计算机专业以及信息管理等相关专业本科生数据库课程的教材，也可作为相关人员学习数据库知识的参考书。

图书在版编目（CIP）数据

数据库原理与应用/何玉洁，等编著. —2 版. —北京：机械工业出版社，2011.4
（高等院校计算机教材系列）

ISBN 978-7-111-32501-7

Ⅰ. 数… Ⅱ. 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（2010）第 221426 号

机械工业出版社（北京市西城区百万庄大街 22 号　邮政编码 100037）
责任编辑：刘立卿
北京市荣盛彩色印刷有限公司印刷
2011 年 4 月第 2 版第 1 次印刷
185mm×260mm • 20.75 印张
标准书号：ISBN 978-7-111-32501-7
定价：35.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换
客服热线：（010）88378991；88361066
购书热线：（010）68326294；88379649；68995259
投稿热线：（010）88379604
读者信箱：hzjsj@hzbook.com

第 2 版前言

《数据库原理与应用》第 1 版出版于 2006 年，距今已有将近 5 年时间。在这 5 年中，国内大学的计算机教育水平又有了很大程度的提高，另一方面，数据库技术也有了新的发展。以 SQL Server 为例，在第 1 版编写之初，SQL Server 2005 版还没有正式发布，而现在 Microsoft 公司已经发布了 SQL Server 2008。新产品的发布意味着新功能的产生。综合这两个方面，并根据近几年读者及教师对该书使用情况的建议和意见，决定对第 1 版进行修订，以使该书结构更加合理，内容更加适合计算机专业学生的需求。

第 2 版相对于对第 1 版主要修订的内容包括：

- 在后台数据库管理系统实践平台上，将 SQL Server 2000 实践平台改为 SQL Server 2005，以适应数据库技术发展需求。
- 在客户端编程实践方面，将 Visual Basic 6.0 改为 Visual Studio.NET 2005，主要以 ASP.NET 2.0 为主介绍开发 B/S 架构的数据库应用程序的基本方法。这些开发平台和技术都是目前非常流行的。
- 去掉了第 1 版的第 13 章“数据传输”、附录 A“常用的 SQL Server 内置函数”、附录 B“发布 VB 应用程序”和附录 D“习题答案”。去掉附录 D 主要是出于篇幅的考虑，有需求的老师可到华章网站下载。
- 对第 4 章“SQL 语言”的内容进行了拆分，将其中的数据操作功能独立为现在的第 5 章，将其中的索引部分提取出来与视图构成新的一章。增加了数据操作功能的介绍，同时增加了索引内容的介绍，使学生学习起来更有条理。
- 第 1 版第 5 章中的存储过程和用户自定义函数内容，在第 2 版中被放置到第二部分“服务器端技术”中，这样一方面可以与具体的数据库管理系统紧密结合，另一方面更便于学生上机实践。
- 增加了触发器和游标的介绍，可有助于提高计算机专业的学生服务器端数据库编程的能力。
- 极大地充实了原附录 C 中数据库设计实例的内容，便于学生更清晰地了解数据库的分析与设计全过程。这部分同时给出了很多实现代码，以方便教师和学生使用。

本书主要由三部分组成，第一部分偏重于数据库理论知识的介绍，由第 1~9 章组成。主要介绍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 语言的数据类型及定义表、数据完整性约束的功能、索引和视图、关系规范化理论、数据库设计以及事务与并发控制几个方面。

第二部分主要介绍服务器端的数据库技术，这里以 SQL Server 2005 数据库管理系统为例，介绍其主要功能以及在此系统中对数据库理论知识的实现，这部分由第 10~15 章组成，具体内容包括安装和配置 SQL Server 2005 以及在此环境中创建数据库、关系表、实现数据完整性约束、进行安全管理、备份和恢复等技术，同时介绍了定义存储过程、函数、游标、触发器等编程技术。这部分作为第一部分的技术实践及技能提高。

第三部分主要介绍在客户端实现对数据库数据的访问技术，这部分以 ASP.NET 2.0 为例，介绍了开发 B/S 架构的数据库应用程序的基本方法，这部分由第 16～18 章组成。主要介绍了 ASP.NET 2.0 环境的配置方法、ASP.NET2.0 内置对象的使用以及用 ASP.NET2.0 访问数据的一些基本技术。

本书还包括两个附录，附录 A 介绍了开发 Web 应用程序时一些页面布局和外观的设置方法；附录 B 给出了一个数据库分析与设计示例，该附录的目的是将前边学习的知识汇总起来，使读者通过本书的学习能够具备一般的数据库分析与设计能力。附录 B 也可作为学生学习时的上机练习题目。

本书的最大特点是内容涵盖全面，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器端的应用技术。在介绍数据库理论知识、选择服务器端和客户端工具上进行了反复思考，力争做到既紧密结合技术的发展趋势和使用实情，又便于学生学习和掌握，使读者通过本教材的学习能够全面掌握数据库知识，并能学以致用。在理论知识的讲解上，尽力做到语言流畅、条理清晰、通俗易懂，并通过大量的例子进行细致的讲解。在工具的选择上注重流行性和功能的全面性。

作者在编写本书的过程中得到了机械工业出版社华章公司总编辑温莉芳女士和姚蕾女士的极大帮助和鼓励，本教材的修订是在华章公司调研了大量教材使用教师的反馈意见基础上进行的，是这些教师和读者的大力支持与帮助，鼓励我坚持完成了此教材的修订工作。在此，对这些无私帮助我的朋友表示衷心的感谢。

该书主要由何玉洁负责组织和审定，梁琦老师编写了本书的第 16、17、18 章以及附录 A，参与本书编写工作的其他老师有路旭强、谷葆春、何青、李迎等，是他们的积极参与和帮助，使本教材的修订工作得以顺利完成。

真诚地希望读者和同行们能对这本教材提出宝贵的意见，因为我知道在教学探索的道路上没有止境。我很希望能与广大读者和同行进行交流，以帮助我不断进步。

何玉洁

2011 年 2 月

目　录

第一部分 基 础 理 论

本部分主要介绍数据库的基础理论知识，包括数据和数据模型，关系数据库的标准操作语言——SQL，数据库的完整性，如何设计性能良好的关系表，如何实现事务的并发控制以及如何对数据库应用系统进行分析和设计。

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，用于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，商场的进货与销售，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，数据库技术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。

数据库是数据管理的最新技术，其主要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，数据库本身可以看做是一个电子文件柜，也就是说，数据库是收集数据文件的仓库或容器。

本章介绍数据库系统的基本概念，包括数据管理的发展过程、数据库系统的组成以及使用数据库技术的一些考虑等。读者可从本章了解为什么要学习数据库技术，并为后续章节的学习做好准备。

1.1 一些基本概念

在系统地介绍数据库技术之前，首先介绍数据库中常用的一些术语和基本概念。

1.1.1 数据

数据（data）是数据库中存储的基本对象。早期的计算机系统主要用在科学计算领域，处理的数据基本是数值型数据，因此数据在人们头脑中的直觉反应就是数字，但数字只是数据的一种最简单的形式，是对数据的传统和狭义的理解。目前计算机的应用范围已十分广泛，因此数据种类也更加丰富，比如文本、图形、图像、音频、视频、商品销售情况等都是数据。

可以将数据定义为：数据是描述事物的符号记录。描述事物的符号可以是数字，也可以是文字、图形、图像、声音、语言等，数据有多种表现形式，它们都可以经过数字化后保存在计算机中。

数据的表现形式并不一定能完全表达其内容，有些还需要经过解释才能明确其表达的

含义，比如20，当解释其代表人的年龄时就是20岁，当解释其代表商品价格时，就是20元。因此，数据和数据的解释是不可分的。数据的解释是对数据演绎的说明，数据的含义称为数据的语义。

在日常生活中，人们一般直接用自然语言来描述事物，例如描述一门课程的信息：数据库系统基础课程，4个学分，第5学期开设。但在计算机中经常按如下形式描述：

(数据库系统基础，4，5)

即把课程名、学分、开课学期信息组织在一起，形成一个记录，这个记录就是描述课程的数据。这样的数据是有结构的。记录是计算机表示和存储数据的一种格式或方法。

1.1.2　数据库

数据库（Database，DB），顾名思义，就是存放数据的仓库，只是这个仓库是存储在计算机存储设备上的，而且是按一定的格式存储的。

人们在收集并抽取出一个应用所需要的大量数据之后，就希望将这些数据保存起来，以供进一步从中得到有价值的信息，并进行相应的加工和处理。在科学技术飞速发展的今天，人们对数据的需求越来越多，数据量也越来越大。最早人们把数据存放在文件柜里，现在人们可以借助计算机和数据库技术来科学地保存和管理大量的复杂数据，以便能方便而充分地利用宝贵的数据资源。

严格地讲，数据库是长期存储在计算机中的有组织的、可共享的大量数据的集合。数据库中的数据按一定的数据模型组织、描述和存储，具有较小的数据冗余、较高的数据独立性和易扩展性，并可为多种用户共享。

概括起来，数据库数据具有永久存储、有组织和可共享三个基本特点。

1.1.3　数据库管理系统

在了解了数据和数据库的基本概念之后，下一个需要了解的就是如何科学有效地组织和存储数据，如何从大量的数据中快速地获得所需的数据以及如何对数据进行维护，这些都是数据库管理系统（Database Management System，DBMS）要完成的任务。数据库管理系统是一个专门用于实现对数据进行管理和维护的系统软件。

数据库管理系统位于用户应用程序与操作系统软件之间，如图1-1所示。数据库管理系统与操作系统一样都是计算机的基础软件，同时也是一个非常复杂的大型系统软件，其主要功能包括如下几个方面：

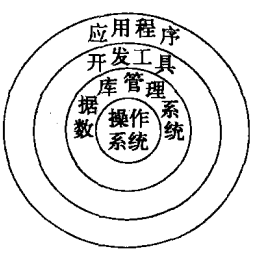


图1-1　数据库管理系统在计算机系统中的位置

1. 数据库的建立与维护功能

包括创建数据库及对数据库空间的维护，数据库的备份与恢复功能，数据库的重组功能，数据库的性能监视与调整功能等。这些功能一般是通过数据库管理系统中提供的一些实用工具实现的。

2. 数据定义功能

包括定义数据库中的对象，比如表、视图、存储过程等。这些功能的实现一般是通过数据库管理系统提供的数据定义语言（Data Definition Language，DDL）实现的。

3. 数据组织、存储和管理功能

为提高数据的存取效率，数据库管理系统需要对数据进行分类存储和管理。数据库中的数据包括数据字典、用户数据和存取路径数据等。数据库管理系统要确定这些数据的存

储结构、存取方法、存储位置，以及如何实现数据之间的关联。确定数据的组织和存储的主要目的是提高存储空间利用率和存取效率。一般的数据库管理系统都会根据数据的具体组织和存储方式提供多种数据存取方法，比如索引查找、Hash 查找、顺序查找等。

4. 数据操作功能

包括对数据库数据的查询、插入、删除和更改操作，这些操作一般是通过数据库管理系统提供的数据操作语言（Data Manipulation Language，DML）实现的。

5. 事务的管理和运行功能

数据库中的数据是可供多个用户同时使用的共享数据，为保证数据能够安全、可靠地运行，数据库管理系统提供了事务管理功能，这些功能保证数据能够并发使用并且不会产生相互干扰的情况，而且在发生故障时（包括硬件故障和操作故障等）能够对数据库进行正确的恢复。

6. 其他功能

包括与其他软件的网络通信功能、不同数据库管理系统间的数据传输以及互访问功能等。

1.1.4 数据库系统

数据库系统（Database System，DBS）是指在计算机中引入数据库后的系统，一般由数据库、数据库管理系统（及相关的实用工具）、应用程序、数据库管理员组成。为保证数据库中的数据能够正常、高效地运行，除了数据库管理系统之外，还需要专门人员来对数据库进行维护，这些专门人员称为数据库管理员（Database Administrator，DBA）。

一般在不引起混淆的情况下，常常把数据库系统简称为数据库。

1.2 数据管理技术的发展

数据库技术是应数据管理任务的需要而产生和发展的。数据管理包括对数据进行分类、组织、编码、存储、检索和维护，它是数据处理的核心，而数据处理则是对各种数据的收集、存储、加工和传播等一系列活动的总和。

自计算机产生之后，人们就希望用它来帮助我们对数据进行存储和管理。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，文件管理方式已经很难满足人们对数据的需求，由此产生了数据库技术，也就是用数据库来存储和管理数据。数据管理技术的发展因此也就经历了文件管理和数据库管理两个阶段。

本节介绍文件管理方式和数据库管理方式在管理数据上的主要差别。

1.2.1 文件管理

理解今日数据库特征的最好办法是了解在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、插入和删除等操作。

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自己定义的文件中，而且还可以通过编写应用程序的方式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。我们将由开发人员定义存储数据的文件及文件结构，并借助文件管理系统的功能编写访问这些文件的应用程序，以实现对用户数据的处理的方式称为**文件管理**。为叙述简单，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。

如果用文件管理数据，用户必须编写应用程序来管理存储在文件中的数据，其操作模式如图 1-2 所示。

假设某学校要用文件的方式保存学生及其选课的数据，并在这些数据文件基础之上构建对学生进行管理的系统。此系统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。假设用 F2 和 F3 文件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 1-3 所示（图中省略了操作系统部分）。

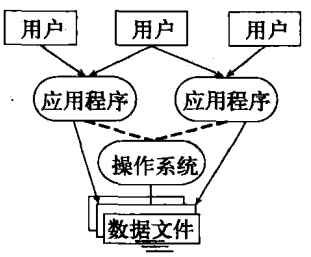


图 1-2　文件管理的操作模式

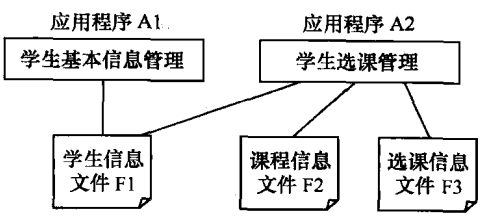


图 1-3　文件管理实现示例

假设 F1、F2 和 F3 文件分别包含如下信息：

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、修课类型、修课时间、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”功能的处理过程大致为：在学生选课管理过程中，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，就会发现直接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，采用何种逻辑结构和物理存储结构。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 1-3 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,称为数据冗余。

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,若让应用程序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;甚至在只是修改了文件的存储位置或者文件名的情况下,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,而不能在第一个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,然后修改其内容,最后再关闭文件。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,否则就不会有我们现在这么多的火车或飞机的订票点,也不会有这么多的银行营业网点。

(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编程来实现。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，我们必须通过编写应用程序，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

不同的用户（数据使用者）关注的数据往往不同。例如，对于学生基本信息，负责分配学生宿舍的部门可能只关心学生的学号、姓名、性别和班号，而教务部门可能关心的是学号、姓名、所在系和专业。

若多个不同用户希望看到的是学生的不同基本信息，则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，使用户感觉不到其他信息的存在。

可能还会有一些用户，其所需要的信息来自于多个不同的文件，例如，假设各班班主任关心的是：班号、学号、姓名、课程名、学分、考试成绩等。这些信息涉及了三个文件：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据。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，在按文件管理数据的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能够进行的操作，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。针对文件管理方式的这些缺陷，人们逐步开发出了以统一管理和共享数据为主要特征的数据库管理系统。

1.2.2　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，此外，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和应用程序的成本相应的不断增加。在数据处理方式上，对联机实时处理的要求越来越多，同时开始提出和考虑分布式处理技术。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。

从 1.2.1 节的介绍我们可以看到，在数据库管理系统出现之前，人们对数据的操作是直

接针对数据文件编写应用程序实现的，这种模式会产生很多问题。在有了数据库管理系统之后，人们对数据的操作全部是通过数据库管理系统实现的，而且应用程序的编写也不再直接针对存放数据的文件。有了数据库技术和数据库管理系统软件之后，人们对数据的操作模式发生了根本的变化，如图 1-4 所示。

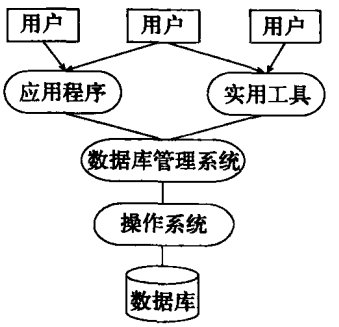


图 1-4　数据库管理的操作模式

比较图 1-2 和图 1-4，可以看到主要区别有两个：第一个是在操作系统和用户应用程序之间增加了一个系统软件——数据库管理系统，使得用户对数据的操作都是通过数据库管理系统实现的；第二个是有了数据库管理系统之后，用户不再需要有数据文件的概念，即不再需要知道数据文件的逻辑和物理结构及物理存储位置，而只需要知道存放数据的场所——数据库即可。

从本质上讲，即使在有了数据库技术之后，数据最终还是以文件的形式存储在磁盘上的，只是这时对物理数据文件的存取和管理是由数据库管理系统统一实现的，而不是每个用户通过应用程序编程实现。数据库和数据文件既有区别又有联系，它们之间的关系非常类似于单位的名称和地址之间的关系。单位地址代表了单位的实际存在位置，单位名称是单位的逻辑代表。而且一个数据库可以包含多个数据文件，就像一个单位可以有多个不同的地址一样（我们现在的很多大学，都是一个学校有多个校址），每个数据文件存储数据库的部分数据。不管一个数据库包含多少个数据文件，对用户来说他只需针对数据库进行操作，而无须对数据文件进行操作。这种模式极大地简化了用户对数据的访问。

在有了数据库技术之后，用户只需要知道数据库的名字，就可以对数据库对应的数据文件中的数据进行操作。将对数据库的操作转换为对物理数据文件的操作是由数据库管理系统自动实现的，用户不需要知道，也不需要干预。

对于 1.2.1 节中列举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，其实现方式如图 1-5 所示。

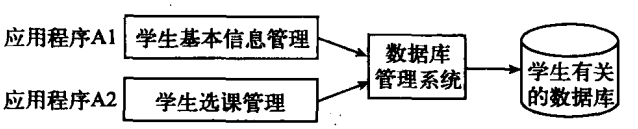


图 1-5　数据库管理实现示例

与文件管理相比，数据库管理具有以下特点：

（1）相互关联的数据集合

在数据库系统中，所有相关的数据都存储在一个称为数据库的环境中，它们作为一个整体定义。比如学生基本信息中的“学号”与学生选课管理中的“学号”，这两个学号之间是有关联关系的，即学生选课管理中的“学号”的取值范围在学生基本信息管理的“学号”取值范围内。在关系数据库中，数据之间的关联关系是通过定义外键实现的。

（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 1.2.1 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，可以将每一类信息存储在一个表中（关系数据库的概念将从第 2 章开始介绍），重复的信息只存储一份，当在学生选课管理中需要学生的姓名等其他信息时，根据学生选课管理中的学号，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 1.2.1 节中的问题一样，当所需的信息来自不同地方，比如班号、学号、姓名、课程名、学分、考试成绩等，这些信息需要从三个地方（关系数据库为三张表）得到，这种情况下，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，避免数据冗余不会增加开发者的负担。在关系数据库中，避免数据冗余是通过关系规范化理论实现的。

（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一起，它们通过 DBMS 而不是应用程序来操作和管理，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。一方面是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，应用程序不必做任何修改。另一方面是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，应用程序并不知道也不需要知道数据存储方式或数据项已经发生了变化。

在关系数据库中，数据库管理系统可以自动保证程序与数据相互独立。

（4）保证数据的安全和可靠

数据库技术能够保证数据库中的数据是安全和可靠的。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和非法修改；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。保证数据的安全是通过数据库管理系统的安全控制机制实现的，保证数据的可靠是通过数据库管理系统的备份和恢复机制实现的。

（5）最大限度地保证数据的正确性

数据的正确性（也称为数据的完整性）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，数据库管理系统能主动拒绝这些数据。