

普.通.高.等.学.校  
计算机教育“十二五”规划教材

# 数据库系统教程

(第2版)

*DATABASE SYSTEM TUTORIAL*  
*(2<sup>nd</sup> edition)*

何玉洁 ◆ 主编



 中国工信出版集团

 人民邮电出版社  
POSTS & TELECOM PRESS

普.通.高.等.学.校  
计算机教育“十二五”规划教材

# 数据库系统教程

## (第2版)

*DATABASE SYSTEM TUTORIAL*  
(2<sup>nd</sup> edition)

何玉洁 ◆ 主编

人民邮电出版社  
北京

## 图书在版编目(CIP)数据

数据库系统教程 / 何玉洁主编. -- 2版. -- 北京 :  
人民邮电出版社, 2015. 12  
普通高等学校计算机教育“十二五”规划教材  
ISBN 978-7-115-40245-5

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

中国版本图书馆CIP数据核字(2015)第217928号

## 内 容 提 要

本书共4篇15章,其中第I篇是基础篇,由第1~7章组成,主要内容包括数据库概述、数据模型与数据库结构、关系数据库、SQL基础及数据定义功能、数据操作语句、索引和视图、触发器和存储过程;第II篇是设计篇,由第8~10章组成,主要内容包括关系规范化理论、实体-联系模型、数据库设计;第III篇是系统篇,由第11~14章组成,具体内容包括安全管理、实务与并发控制、数据库恢复技术、查询处理与优化;第IV篇是发展篇,由第15章组成,主要介绍了大规模数据库架构。最后还有三个附录,其中附录A介绍了SQL Server 2008的安装、配置,以及如何在该平台下创建和维护数据库;附录B是SQL Server 2008提供的一些常用的系统函数;附录C是上机实验。

本书可作为高等院校计算机专业及信息管理专业本科生的数据库教材,也可作为相关人员学习数据库的参考书。

- 
- ◆ 主 编 何玉洁  
责任编辑 武恩玉  
责任印制 沈 蓉 彭志环
  - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号  
邮编 100164 电子邮件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
三河市潮河印业有限公司印刷
  - ◆ 开本: 787×1092 1/16  
印张: 21 2015年12月第2版  
字数: 554千字 2015年12月河北第1次印刷
- 

定价: 49.80元

读者服务热线: (010)81055256 印装质量热线: (010)81055316  
反盗版热线: (010)81055315

## 第2版前言

数据库技术起源于20世纪60年代末,经过40多年的迅速发展,已经建立起一套较完整的理论体系,产生了一大批商用软件产品。随着数据库技术的推广使用,计算机应用已深入国民经济和社会生活的各个领域,这些应用一般都以数据库技术及其应用为基础和核心。因此,数据库技术与操作系统一起构成信息处理的平台已成为业界的一种共识。在计算机应用中,数据存储和数据处理是计算机的最基本功能,数据库技术为人们提供了科学、高效的管理数据的方法。从某种意义上讲,数据库技术的教学成为计算机专业教学的重中之重,数据库课程也成为很多高校计算机专业的重点核心课程。目前,市场上数据库类的教科书非常多,每本书都各有特色,本书主要博采众家之长,在完整包括数据库基础理论知识的同时,加入了将数据库知识与具体数据库管理系统结合的内容,以方便学生在实践中更好地掌握所学知识。

师者何为?“所以传道授业解惑也。”道中有惑,业中也有惑。传授知识、教授学业就是为了解答学生的疑惑。长期的数据库技术专业教学和科研实践,使编者感受到一位师者的责任重大。而学生最大的疑惑又是什么呢?编者认为是如何学以致用。专业基础的教学绝不能脱离实践,专业的教材也必须以实用为先导。经过长期总结、反复思考和精心筛选,并听取业界专家的意见,编者确定了本书以基本理论为基础,以商品化的流行数据库产品为平台,以切合实际应用为目标,有效地增强学生实践训练和动手能力的培养,真正做到学以致用。依据这些宗旨,编者编写了这本独具特色的适合大学计算机专业以及信息管理专业本科生使用的数据库教材。

本书具有以下特色。

- 内容安排求全、求新。本书从数据库基础理论、数据库设计、数据库发展、数据库实践几个方面全面阐述了数据库技术的应用体系。该书分为基础篇、设计篇、系统篇、发展篇4个部分及三个附录,这种安排最大限度地满足了教学和实践需要,相信无论是学习数据库知识的专业及非专业的学生,还是从事数据库实践的人员,都能够从中找到自己所需的内容。本书在选择实践平台时,充分考虑了软件的流行性和易获得性。后台数据库管理系统选用的是SQL Server 2008,它是目前应用范围广泛且功能完善、操作界面友好的数据库管理系统。
- 理论阐述求精、求易。数据库基础理论较为抽象,但又是实践的基础,没有扎实的基本功是无法灵活运用、付诸实践的。因而,基础理论的教学历来是重点和难点。本书在理论阐释方面力求深入浅出、突出概念和技术的直观意义,并用大量图表和示例帮助读者理解知识点,启发思维,使读者不仅能深刻理解相关理论的来源、思路、适用范围和条件,并能灵活运用,举一反三。



- 理论实践丝丝相扣。知之明也，因知进行。理论和技术的学习是为了更好地指导实践，每部分内容都是根据相关理论和应用需求进行精当的选取，不求全面泛泛取胜，只求精而实用。本书不但以图例的形式细致地描述了实践步骤，还给出了执行结果，使学生能够以行验知，以行证知，最后达到知行并进，相资为用，为进一步的学习和实践打下良好基础。同时，每章后面都有大量的习题供读者了解自己对知识的掌握程度。在实践部分除概念题外，还附有丰富的上机练习题，以方便读者上机实践。

相对于第1版，第2版删去了第1版中不常用及难度比较大的知识，增加了一些新的比较实用的知识，同时对实践平台进行了更新。主要修订内容如下。

- (1) 将实践平台从 SQL Server 2005 升级到 SQL Server 2008。
- (2) 删除了第1版第3章中的关系演算、半连接内容。
- (3) 将第1版“第8章 数据完整性约束”中的声明完整性内容合并到“第4章 SQL 基础及数据定义功能”，使表的定义内容更完整。
- (4) 将第1版“第6章 高级查询”中的“CASE 表达式”和“将查询结果保存到新表”内容合并到“第5章 数据操作语句”，删除了第1版“第6章 高级查询”中的 ANY、SOME、ALL 查询及查询结果的并、交、差运算部分。同时，在第2版的第5章增加了临时表、批量插入语句的介绍。
- (5) 将第1版“第8章 数据完整性约束”中的触发器内容及“第19章”的存储过程部分合并在一起，构成了第2版的“第7章 触发器和存储过程”。将数据库后台编程部分放置到第I篇，使第I篇内容更加完整。
- (6) 删除了第1版“第9章 关系规范化理论”中的多值依赖、第四范式、第五范式内容。
- (7) 删除了第1版第19章中的游标内容，删除了第1版第15章、21章全部内容，删除了第18章中在 SQL Server 平台中图形化方法创建表、视图和索引的内容。
- (8) 第2版增加了大规模数据库架构的介绍（第2版第15章），SQL Server 2008 提供的常用函数介绍（附录B）。
- (9) 将第1版中对 SQL Server 的安装、配置及创建数据库的介绍放置在了第2版的附录A中。

在修订本书过程中，清华大学的刘乃嘉老师在百忙之中抽出时间完成了本书第15章的编写工作，在此对刘老师表示衷心的感谢。本书是编者们对多年数据库教学和实践工作的总结，另外非常感谢数据库课程组的同仁：殷旭、张鸿斌、李宝安、谷葆春、岳清、刘京志、梁琦、韩麦燕老师，他们对本书的修订提出了积极的意见和建议，正是由于他们的积极参与和帮助，才使本书得以顺利完成。最后也是最重要的，我由衷地感谢学生们，是他们对知识的渴求，对教师的尊重让编者感到了自己的责任和价值；也是他们的勤奋努力为编者的工作精进提供了取之不竭的源泉。师者之尊，缘自“用心”。

由于时间有限，书中不足之处在所难免。在此，编者真诚地希望读者和同行们对这本书提出宝贵的意见。

编者  
何玉洁

# 目 录

## 第 I 篇 基础篇

<b>第 1 章 数据库概述</b> .....2	
1.1 概述.....2	
1.2 基本概念.....2	
1.3 数据管理技术的发展.....4	
1.3.1 文件管理.....4	
1.3.2 数据库管理.....7	
1.4 数据独立性.....10	
1.5 数据库系统的组成.....11	
习题.....12	
<b>第 2 章 数据模型与数据库结构</b> .....13	
2.1 数据和数据模型.....13	
2.1.1 数据与信息.....13	
2.1.2 数据模型.....14	
2.2 概念层数据模型.....15	
2.2.1 基本概念.....15	
2.2.2 实体-联系模型.....16	
2.3 组织层数据模型.....18	
2.3.1 层次数据模型.....18	
2.3.2 网状数据模型.....20	
2.3.3 关系数据模型.....21	
2.4 面向对象数据模型.....22	
2.5 数据库结构.....23	
2.5.1 模式的基本概念.....23	
2.5.2 三级模式结构.....23	
2.5.3 模式映像与数据独立性.....25	
习题.....26	
<b>第 3 章 关系数据库</b> .....28	
3.1 关系数据模型.....28	
3.1.1 数据结构.....28	
3.1.2 数据操作.....28	
3.1.3 数据完整性约束.....30	
3.2 关系模型的基本术语与形式化定义.....31	
3.2.1 基本术语.....31	
3.2.2 形式化定义.....33	
3.3 完整性约束.....35	
3.3.1 实体完整性.....35	
3.3.2 参照完整性.....36	
3.3.3 用户定义的完整性.....38	
3.4 关系代数.....38	
3.4.1 传统的集合运算.....39	
3.4.2 专门的关系运算.....41	
习题.....50	
<b>第 4 章 SQL 基础及数据定义功能</b> .....52	
4.1 SQL 概述.....52	
4.1.1 SQL 的发展.....52	
4.1.2 SQL 特点.....53	
4.1.3 SQL 功能概述.....53	
4.2 SQL 支持的数据类型.....54	
4.2.1 数值型.....54	
4.2.2 字符串类型.....55	
4.2.3 日期时间类型.....56	
4.3 数据定义功能.....56	
4.3.1 架构的定义与删除.....57	
4.3.2 基本表.....58	
习题.....62	
<b>第 5 章 数据操作语句</b> .....64	
5.1 数据查询语句的基本结构.....64	
5.2 单表查询.....66	
5.2.1 选择表中的若干列.....66	
5.2.2 选择表中的若干元组.....68	
5.2.3 对查询结果进行排序.....76	

5.2.4	使用聚合函数汇总数据	77
5.2.5	对数据进行分组统计	79
5.3	多表连接查询	84
5.3.1	内连接	85
5.3.2	自连接	89
5.3.3	外连接	90
5.4	使用 TOP 限制结果集行数	93
5.5	CASE 表达式	95
5.5.1	CASE 表达式介绍	95
5.5.2	CASE 表达式应用示例	97
5.6	将查询结果保存到表中	99
5.7	子查询	102
5.7.1	使用子查询进行基于集合的测试	102
5.7.2	使用子查询进行比较测试	106
5.7.3	带 EXISTS 谓词的子查询	110
5.8	数据更改功能	112
5.8.1	插入数据	113
5.8.2	更新数据	114
5.8.3	删除数据	115
习题		117

## 第6章 索引和视图 120

6.1	索引	120
6.1.1	索引的基本概念	120
6.1.2	索引的存储结构及分类	121
6.1.3	创建和删除索引	126
6.2	视图	127
6.2.1	视图的基本概念	127
6.2.2	定义视图	128
6.2.3	通过视图查询数据	130
6.2.4	修改和删除视图	132
6.2.5	视图的作用	133
6.3	物化视图	134
习题		134

## 第7章 触发器和存储过程 136

7.1	触发器	136
7.1.1	创建触发器	136
7.1.2	后触发型触发器	137
7.1.3	前触发型触发器	138
7.1.4	删除触发器	139
7.2	存储过程	139
7.2.1	存储过程的概念	139
7.2.2	创建和执行存储过程	140
习题		144

# 第II篇 设计篇

## 第8章 关系规范化理论 146

8.1	函数依赖	146
8.1.1	基本概念	146
8.1.2	术语和符号	147
8.1.3	函数依赖的推理规则	148
8.1.4	闭包及候选键求解方法	149
8.1.5	极小函数依赖集	152
8.1.6	为什么讨论函数依赖	154
8.2	范式	155
8.2.1	第一范式	156
8.2.2	第二范式	157
8.2.3	第三范式	158
8.2.4	Boyce-Codd 范式	159
8.2.5	规范化小结	161
8.3	关系模式的分解准则	162
习题		165

## 第9章 实体-联系模型 167

9.1	E-R 模型的基本概念	167
9.1.1	实体	167
9.1.2	联系	168
9.1.3	属性	170
9.1.4	约束	173
9.2	E-R 图符号	174
习题		175

## 第10章 数据库设计 177

10.1	数据库设计概述	177
10.1.1	数据库设计的特点	178
10.1.2	数据库设计方法概述	178
10.1.3	数据库设计的基本步骤	179
10.2	数据库需求分析	180
10.2.1	需求分析的任务	180

10.2.2 需求分析的方法	180	10.4.1 功能分析	197
10.2.3 需求分析工具	182	10.4.2 功能设计	198
10.3 数据库结构设计	184	10.4.3 事务设计	198
10.3.1 概念结构设计	184	10.5 数据库实施	198
10.3.2 逻辑结构设计	191	10.6 数据库的运行和维护	200
10.3.3 物理结构设计	195	习题	200
10.4 数据库行为设计	197		

## 第Ⅲ篇 系统篇

### 第 11 章 安全管理

11.1 安全控制概述	204
11.1.1 数据库安全控制的目标	204
11.1.2 数据库安全的威胁	205
11.1.3 数据库安全问题的类型	205
11.1.4 安全控制模型	205
11.1.5 授权和认证	206
11.2 存取控制	207
11.2.1 自主存取控制	207
11.2.2 强制存取控制	209
11.3 审计跟踪	211
11.4 防火墙	211
11.5 统计数据库的安全性	212
11.6 数据加密	212
11.7 SQL Server 提供的安全控制	213
11.8 登录名	214
11.8.1 身份验证模式	214
11.8.2 建立登录名	215
11.8.3 删除登录名	216
11.9 数据库用户	217
11.9.1 建立数据库用户	218
11.9.2 删除数据库用户	220
11.10 权限管理	220
习题	228

### 第 12 章 事务与并发控制

12.1 事务	230
12.1.1 事务的基本概念	230
12.1.2 事务的特性	231
12.1.3 事务处理模型	232
12.1.4 事务日志	233
12.2 并发控制	234

### 系统篇

12.2.1 并发控制概述	235
12.2.2 可交换的活动	237
12.2.3 调度	237
12.2.4 可串行化调度	237
12.3 并发控制中的加锁方法	238
12.3.1 锁的粒度	239
12.3.2 封锁协议	240
12.3.3 活锁和死锁	242
12.3.4 两阶段锁	244
12.4 乐观的并发控制方法	247
12.4.1 乐观的并发控制方法中的三个阶段	247
12.4.2 乐观的并发控制方法的优缺点	247
习题	248

### 第 13 章 数据库恢复技术

13.1 恢复的基本概念	250
13.2 数据库故障的种类	251
13.3 数据库恢复的类型	252
13.3.1 向前恢复(或重做)	252
13.3.2 向后恢复(或撤销)	253
13.3.3 介质故障恢复	255
13.4 恢复技术	256
13.4.1 延迟更新技术	256
13.4.2 立即更新技术	258
13.4.3 镜像页技术	260
13.4.4 检查点技术	260
13.5 缓冲区管理	262
习题	262

### 第 14 章 查询处理与优化

14.1 概述	264
14.2 关系数据库的查询处理	264

14.2.1	查询处理步骤	265	14.4.1	选择操作的实现和优化	272
14.2.2	优化的一个简单示例	265	14.4.2	连接操作的实现和优化	273
14.3	代数优化	267	14.4.3	投影操作的实现	277
14.3.1	转换规则	267	14.4.4	集合操作的实现	277
14.3.2	启发式规则	269	14.4.5	组合操作	277
14.4	物理优化	272	习题		278

## 第IV篇 发展篇

### 第15章 大规模数据库架构

15.1	分布式数据库	280
15.1.1	分布式数据库系统概述	280
15.1.2	分布式数据库目标与数据分布策略	280
15.1.3	分布式数据库系统体系结构	282
15.1.4	分布式数据库的相关技术	283
15.2	并行数据库	284
15.2.1	并行数据库概述	284
15.2.2	并行数据库系统结构	284
15.2.3	数据划分与并行算法	286
15.3	NoSQL 数据库	288
15.3.1	NoSQL 数据库概述	288
15.3.2	NoSQL 数据库的分类	288
15.3.3	NoSQL 数据库的发展现状及挑战	289
15.4	云计算数据库架构	289
15.4.1	云计算概述	289
15.4.2	云数据库体系结构	290
15.4.3	云数据库与传统数据库的比较	293
15.4.4	云数据库展望	294
15.5	XML 数据库	294
15.5.1	XML 数据库概述	294
15.5.2	SQL Server 2008 与 XML	295

### 附录 A SQL Server 2008 基础

A.1	SQL Server 2008 的版本和组件	299
A.1.1	SQL Server 2008 的版本	299

### 发展篇

A.1.2	SQL Server 2008 的组件	300
A.1.3	安装 SQL Server 2008 需要的软硬件环境	300
A.1.4	实例	301
A.2	安装和配置 SQL Server 2008	302
A.2.1	安装 SQL Server 2008	302
A.2.2	设置 SQL Server 服务启动方式	309
A.2.3	SQL Server Management Studio 工具	310
A.3	创建数据库	311
A.3.1	数据库的组成	311
A.3.2	数据库文件的属性	312
A.3.3	创建数据库	312
A.4	删除数据库	315

### 附录 B 一些常用的系统函数

B.1	日期函数	317
B.2	字符串函数	319
B.3	类型转换函数	321

### 附录 C 上机实验

C.1	第4章上机实验	323
C.2	第5章上机实验	324
C.3	第6章上机实验	326
C.4	第7章上机实验	326
C.5	第11章上机实验	327



# 第 I 篇

---

## 基础篇

本篇主要介绍数据库的基本概念和基础知识，是读者进一步学习后续章节的基础。本篇由 7 章组成。

第 1 章，数据库概述。介绍了文件管理数据与数据库管理数据的本质区别、数据独立性的含义以及数据库系统的组成。

第 2 章，数据模型与数据库结构。介绍了数据库技术发展过程中使用过的数据模型、数据独立性的概念。本章介绍的知识是读者进一步学习关系数据库及相关知识的基础。

第 3 章，关系数据库。介绍了关系数据库采用的数据模型的特点，同时介绍了关系数据库基于的理论基础——关系代数和关系演算。读者在学习完本章和第 5 章的数据操作语句之后，可以对关系代数、关系演算、SQL 查询语句之间的功能及表达方法进行比较。本章介绍的关系代数也是学习第 14 章查询优化的基础。

第 4 章，SQL 基础及数据定义功能。SQL 部分介绍了常用的数据类型，由于不同的数据库管理系统提供的数据类型不完全相同，因此本章主要介绍的是 SQL Server 数据库管理系统提供的数据类型，这部分内容是定义关系表的基础。数据定义功能部分，介绍了架构和基本表的概念和定义语句，同时介绍了数据完整性约束的定义方法。

第 5 章，数据操作语句。其内容主要包括查询、添加、删除和更改数据的 SQL 语句，同时介绍了一些高级查询功能，包括 CASE 表达式、嵌套子查询和相关子查询等。这一章使用第 4 章建立的数据表，运用实际的数据，通过描述问题的分析思路以及用图示的方法展示查询语句的执行结果，使读者能够准确理解和掌握查询语句的功能。

第 6 章，索引和视图。索引部分除了介绍索引的概念的定义方法外，还用图示的方法详细讲述了索引的构建过程以及利用索引的查找过程，使读者能够从系统内部了解索引的作用。视图部分，介绍了视图的概念和定义语句，并简单介绍了物化视图的概念和作用。

第 7 章，触发器和存储过程。触发器用于实现复杂的完整性约束和业务规则。本章介绍了触发器的概念和使用方法。存储过程是一段封装好的代码块，这个代码块可供应用程序调用使用，存储过程提供了代码共享的功能。

# 第 1 章

## 数据库概述

数据库是管理数据的一种技术。现在，数据库技术已经被广泛应用到我们日常生活中的方方面面。本章首先介绍数据管理技术发展的过程，然后介绍使用数据库技术管理数据的特点和好处。

### 1.1 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，如信用卡购物，飞机、火车订票系统，商场的进销存，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，其主要研究内容是如何对数据进行科学管理，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，但它的功能不仅仅只是保存数据，而且还提供了对数据进行各种管理和处理的功能，如安全管理、数据共享的管理、数据查询处理等。

本章将介绍数据库的基本概念，包括数据管理的发展过程、数据库系统的组成等。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。

### 1.2 基本 概念

在系统地介绍数据库技术之前，首先介绍数据库中最常用的一些术语和基本概念。

#### 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应用于科学计算领域，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，如文本、图形、图像、音频、视频、商品销售情况等都是数据。

我们可以将数据定义为：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，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。因此，数据和数据的语义是不可分的。

在日常生活中，人们一般直接用自然语言来描述事物，如描述一门课程的信息：数据库系统基础，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。

## 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。

## 3. 数据库管理系统

在了解了数据和数据库的基本概念后，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(Database Management System, DBMS)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，其主要功能包括以下几个方面。

### (1) 数据库的建立与维护功能

此功能包括创建数据库及对数据库空间的维护、数据库的转储与恢复功能、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

此功能包括定义数据库中的对象，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库定义语言(Data Definition Language, DDL)实现的。

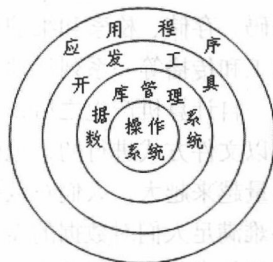


图 1-1 数据库管理系统在  
计算机系统的位置

### (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、存取方式和存储位置,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,如索引查找、Hash 查找、顺序查找等。

### (4) 数据操作功能

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

### (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,而且在数据库发生故障时能够对数据库进行正确恢复。

### (6) 其他功能

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。

## 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、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软件外,还需要一个(或一些)专门人员来对数据库进行维护,这个专门人员就称为数据库管理员(Database Administrator, DBA)。我们将在1.5小节详细介绍数据库系统的组成。

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。

# 1.3 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。数据管理是指对数据进行分类、组织、编码、存储、检索和维护,它是数据处理的核心,而数据处理则是指对各种数据的收集、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,也就是用户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。

本节将介绍文件管理和数据库管理在管理数据上的主要差别。

## 1.3.1 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。在本章后面的讨论中,为描述简单,我们将忽略操作系统中的文件管理系统,假定应用程序直接对磁盘文件进行操作。

用户通过编写应用程序来管理存储在自定义文件中的数据的操作模式如图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,并针对这些数据文件构建对学生及选课情况进行管理的系统。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,各系管理自己的学生基本信息。学生基本信息管理只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。图1-3所示为用文件存储并管理数据的实现示例(图中省略了操作系统部分)。

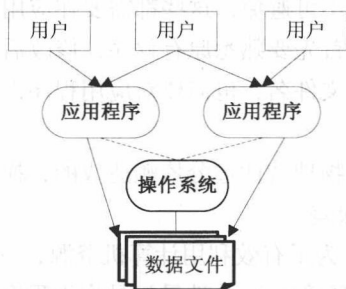


图 1-2 用文件存储数据的操作模式

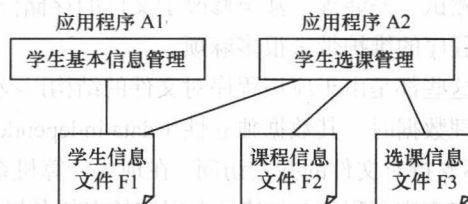


图 1-3 用文件存储并管理数据的实现示例

假设文件F1、F2和F3分别包含以下信息。

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”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,判断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,就会发现用文件方式管理数据有以下缺点。

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、修改等操作,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,如图 1-3 所示的“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,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 文件中应做同样的修改。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。当发生数据不一致时,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对,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的结构。定义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,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,这显然给程序的维护带来很多麻烦。

所有这些都是由于应用程序对文件的结构以及文件的物理特性过分依赖造成的,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 Word 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,必须支持多个用户对数据的并发访问,否则就不会有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程序来实现。例如,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 文件中的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,那么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,将用户不关心的数据屏蔽,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3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3个文件中读取的数据进行比较,然后组合成一行有意义的信息。例如,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,这个过程非常复杂。因此,这种复杂信息的查询,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,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。例如,在学生选课管理中,我们不允许学生修改其考试成绩,但允许他们查询自己的考试成绩。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。

## 1.3.2 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和应用程序的成本相应地不断增加。在数据处理方式上,对联机实时处理的需求越来越多,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1.3.1小节的介绍可以看到,在数据库管理系统出现前,人们对数据的操作是通过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。有了数据库管理系统后,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。

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,只需知道存放数据的场所——数据库即可。

本质上讲,即使在有了数据库技术后,数据最终还是以文件的形式存储在磁盘上的(这点我们将在本书附录 A 中的“创建数据库”部分可以体会到),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,而不再是每个用户通过编写应用程序实现。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(就像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,他只针对数据库进行操作,无需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。

有了数据库技术后,用户只需要知道存放所需数据的数据库名,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 1.3.1 小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实现,其实现方式如图 1-5 所示。

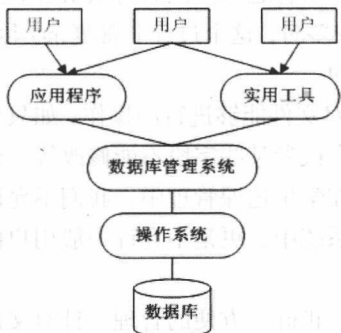


图 1-4 用数据库进行管理的操作模式

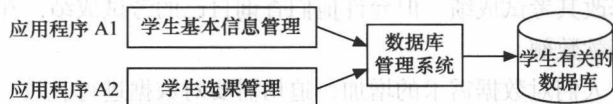


图 1-5 用数据库存储数据的实现示例

与用文件管理数据相比,用数据库技术管理数据具有以下特点。

① 相互关联的数据集合。在用数据库技术管理数据时,所有相关的数据都被存储在一个数据库中,它们作为一个整体定义,因此可以很方便地表达数据之间的关联关系。例如,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,即学生选课中的“学号”的取值范围在学生基本信息的“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参照完整性实现的。

② 较少的数据冗余。由于数据被统一管理,因此可以从全局着眼,对数据进行最合理的组织。例如,将 1.3.1 小节中文件 F1、F2 和 F3 的重复数据挑选出来,进行合理的管理,这样就可以形成以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、选课类型、选课时间、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在后边介绍),重