



新世纪高职高专实用规划教材

● 计算机系列

数据库原理与应用

SHUJUKU YUANLI YU YINGYONG

涂振宇 傅清平 编著



清华大学出版社

新世纪高职高专实用规划教材 计算机系列

数据库原理与应用

涂振宇 傅清平 编著

清华大学出版社

北 京

内 容 简 介

本书介绍了数据库原理的必备知识,并以 SQL Server 2000 为应用平台帮助读者掌握数据库理论在实践中的应用,主要内容包括:数据库系统概述、数据库管理系统、数据模型、关系数据库模型、数据库设计、数据库系统实现、SQL 标准语言、Transact-SQL 基础、Transact-SQL 的高级应用、数据库索引与查询优化、数据库的备份与恢复。本书还介绍了以 Visual Basic 6.0 为前台,SQL Server 2000 为后台开发数据库应用程序的基本方法,使读者可以充分利用 SQL Server 2000 平台深刻理解数据库技术的原理。

本书原理清晰,通俗易懂,强调应用,特别适合作为高职高专学生的教材或参考书,也可供其他高等院校计算机及相关专业学生选用。

版权所有,翻印必究。举报电话:010-62782989 13501256678 13801310933

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

本书防伪标签采用特殊技术,用户可通过在图案表面涂抹清水,图案消失,水干后图案复现;或将表面膜揭下,放在白纸上用彩笔涂抹,图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

数据库原理与应用/涂振宇,傅清平编著. —北京:清华大学出版社,2006.5

(新世纪高职高专实用规划教材 计算机系列)

ISBN 7-302-12779-4

I. 数… II. ①涂…②傅 III. 数据库系统—高等学校:技术学校—教材 IV. TP311.13

中国版本图书馆 CIP 数据核字(2006)第 029575 号

出 版 者:清华大学出版社 地 址:北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编:100084

社 总 机:010-62770175 客 户 服 务:010-62776969

责任编辑:刘建龙

排版人员:李月菊

印 刷 者:北京密云胶印厂

装 订 者:三河市新茂装订有限公司

发 行 者:新华书店总店北京发行所

开 本:185×260 印张:22.25 字数:528 千字

版 次:2006 年 5 月第 1 版 2006 年 5 月第 1 次印刷

书 号:ISBN 7-302-12779-4/TP·8138

印 数:1~4000

定 价:29.00 元

前 言

数据库技术形成于 20 世纪 60 年代,经历了层次数据库、网状数据库、关系数据库 3 个阶段。如今数据库技术已成为计算机信息技术的重要基础。“数据库原理及应用”也是计算机及相关专业学生必修的基础课程。

本书是在作者多年教授“数据库原理与应用”课程的授课讲义的基础之上,结合高等职业教育的特点修改补充而成。在本书的编写过程中,作者努力实现以下写作理念:

(1) 去繁存简、强调应用。

对数据库理论部分以“必需”、“够用”为度,以满足应用技术的教学需要为限,着重强调对学生的数据库技术应用能力的培养。

(2) 着重认识规律。

内容的安排循序渐进,深入浅出,并以具体实例和实际应用引路,分析和阐明数据库技术的概念和原理,尽量避免抽象的理论讲解,由感性到理性地安排和组织教材的内容,以利于学生的掌握和运用。

(3) 采用流行的先进技术平台,引导学生掌握最先进的技术与成果,激发学生的学习热情和兴趣,使学生深入学习相关的知识,掌握和应用相关的理论与技术。

在编写本书时,作者充分考虑到了数据库技术初学者的学习,尤其是高职高专学生的学习和使用。本书由两部分组成:第一部分讲述数据库理论的基础,用通俗易懂的语言阐明了数据库理论中最基本的内容;第二部分阐述了 SQL Server 2000 的基本内容及数据库程序设计的基本方法。

本书编写人员有涂振宇、傅清平、王萍、曾瑄、黄良英、章逸、王樱。全书由涂振宇、傅清平统稿。

本书可作为高职高专计算机应用专业的教材,也可供其他类院校计算机及相关专业的学生选用。欢迎广大读者就书中疏漏之处提出宝贵意见。

编者

目 录

第1章 绪论.....1	1.5.2 多媒体数据库 32
1.1 数据库技术概述1	1.5.3 主动数据库 34
1.1.1 几个基本概念1	1.5.4 并行数据库 34
1.1.2 数据库系统软、 硬件的关系2	1.5.5 workflow数据库 35
1.1.3 数据管理技术的产生 和发展4	1.5.6 数据仓库 36
1.1.4 数据库系统的特点6	1.5.7 数据挖掘 37
1.2 数据库系统的体系结构.....7	1.6 小结 39
1.2.1 数据库系统模式概念8	1.7 思考题 40
1.2.2 数据库系统的三级 模式结构8	第2章 关系数据库理论简介..... 41
1.2.3 数据库的二级映像功能10	2.1 基本概念 41
1.2.4 数据独立性11	2.1.1 关系定义 41
1.3 数据库管理系统12	2.1.2 关系模式 42
1.3.1 数据库管理系统的主要作用12	2.1.3 关系模型 43
1.3.2 数据库管理系统的 程序组成12	2.1.4 关系的性质 44
1.3.3 数据语言13	2.2 关系运算与关系语言 44
1.3.4 数据字典15	2.2.1 关系代数与 ISBL 语言 45
1.3.5 数据库日志16	2.2.2 元组关系演算与 QUEL 语言 57
1.3.6 用户访问数据库的过程16	2.2.3 域关系演算与 QBE 语言 62
1.4 数据模型17	2.3 关系语言的评价 67
1.4.1 数据模型的组成要素 (数据模型的共性).....18	2.4 关系数据库的理论基础..... 68
1.4.2 概念模型19	2.4.1 关系模型评价 68
1.4.3 概念模型的表示方法22	2.4.2 函数依赖 69
1.4.4 层次模型23	2.4.3 关键字 71
1.4.5 网状模型27	2.4.4 关系模式的规范理论基础..... 71
1.4.6 关系模型28	2.4.5 多值依赖与第四范式(4NF)..... 74
1.5 数据库系统的新技术.....31	2.4.6 连接依赖与第五范式(5NF)..... 76
1.5.1 分布式数据库31	2.5 小结 78
	2.6 思考题 78
	第3章 数据库设计 80
	3.1 数据库设计概述..... 80
	3.1.1 数据库设计的特点 80

3.1.2 数据库设计方法学	81	4.2.3 封锁协议	114
3.1.3 数据库设计基本步骤	82	4.2.4 活锁和死锁	116
3.2 需求分析	84	4.2.5 并发调度的可串化	118
3.2.1 需求分析的任务	84	4.2.6 两段封锁法协议	118
3.2.2 需求分析的步骤和方法	84	4.2.7 多粒度封锁法	118
3.3 概念结构设计	87	4.3 数据库的恢复	119
3.3.1 概念结构的主要特征	88	4.3.1 故障种类	120
3.3.2 E-R 模型	88	4.3.2 日志恢复技术	121
3.3.3 概念结构设计方法	90	4.3.3 基于页表的恢复技术	124
3.3.4 数据抽象	91	4.4 数据库的完整性	124
3.3.5 局部 E-R 模型	92	4.4.1 完整性约束条件	125
3.3.6 视图的集成	93	4.4.2 SQL2 中的完整性约束	125
3.4 逻辑结构设计	94	4.4.3 SQL3 中的触发器	131
3.4.1 逻辑结构设计环境	95	4.5 数据库的安全性	134
3.4.2 逻辑结构设计步骤	95	4.5.1 权限	134
3.4.3 E-R 模型转换为关系模型	96	4.5.2 SQL 中的安全性机制	136
3.4.4 优化关系模式	98	4.5.3 数据加密法	138
3.4.5 设计用户外模式	99	4.5.4 统计数据库的安全性	138
3.5 物理结构设计	100	4.5.5 自然环境的安全性	139
3.5.1 物理结构设计的环境	100	4.6 小结	139
3.5.2 物理结构设计步骤	101	4.7 思考题	139
3.5.3 关系模式存取方法的选择	101		
3.5.4 选择关系模式的存储结构	102	第 5 章 SQL Server 数据库的	
3.5.5 物理结构的评价	103	基本应用	141
3.6 数据库的实施和维护	103	5.1 创建及修改数据库	141
3.6.1 数据载入	103	5.1.1 数据库的存储结构	141
3.6.2 试运行	104	5.1.2 创建数据库	143
3.6.3 运行与维护	104	5.1.3 浏览数据库	145
3.7 小结	105	5.1.4 修改数据库	146
3.8 思考题	106	5.1.5 删除数据库	149
第 4 章 数据库系统实现	107	5.2 管理数据库表	150
4.1 事务	107	5.2.1 创建数据库表	150
4.1.1 事务的基本概念	107	5.2.2 修改表	153
4.1.2 事务的性质	107	5.2.3 查看表的属性	154
4.1.3 事务的状态变迁	109	5.2.4 删除表	154
4.2 并发控制	110	5.2.5 表数据的访问	155
4.2.1 事务的并发执行	110	5.3 数据完整性规则	156
4.2.2 两种封锁	113	5.3.1 数据完整性	156
		5.3.2 创建和使用约束	157

5.3.3 规则.....	162	6.6 索引的建立与删除.....	197
5.3.4 默认值.....	165	6.6.1 索引的建立.....	197
5.4 索引.....	165	6.6.2 索引的删除.....	197
5.4.1 索引的概念.....	166	6.7 SQL 数据控制功能.....	198
5.4.2 索引的种类.....	166	6.7.1 使用权的授予.....	198
5.4.3 创建索引.....	167	6.7.2 使用权的收回.....	199
5.4.4 删除索引.....	169	6.8 小结.....	200
5.5 小结.....	170	6.9 思考题.....	200
5.6 思考题.....	170	第7章 Transact-SQL 基础.....	202
第6章 关系数据库标准语言		7.1 Transact-SQL 概述.....	202
——SQL.....	172	7.2 Transact-SQL 语法格式.....	202
6.1 SQL 介绍.....	172	7.3 Transact-SQL 语法中的	
6.1.1 SQL 的历史.....	172	保留关键字.....	204
6.1.2 SQL 特点.....	173	7.4 标识符.....	206
6.1.3 SQL 的体系结构.....	173	7.4.1 标识符定义的规则.....	206
6.1.4 SQL 的组成.....	174	7.4.2 标识符的分类.....	206
6.2 SQL 数据定义语言.....	175	7.4.3 对象命名规则.....	207
6.2.1 创建数据库.....	175	7.5 数据类型.....	208
6.2.2 修改数据库.....	177	7.5.1 字符数据类型.....	208
6.2.3 删除数据库.....	178	7.5.2 整数数据类型.....	210
6.2.4 建立表.....	179	7.5.3 数字数据类型.....	213
6.2.5 修改表结构.....	180	7.5.4 浮点数据类型.....	214
6.2.6 删除表.....	181	7.5.5 日期/时间数据类型.....	215
6.3 SQL 数据查询功能.....	181	7.5.6 货币数据类型.....	216
6.3.1 简单查询.....	183	7.5.7 Unicode 数据类型.....	217
6.3.2 多表查询.....	186	7.5.8 二进制数据类型.....	218
6.3.3 嵌套查询.....	188	7.5.9 自定义数据类型.....	219
6.3.4 组函数.....	190	7.6 常量.....	221
6.3.5 UNION 查询.....	191	7.7 变量.....	223
6.3.6 INTO 子句.....	191	7.7.1 局部变量.....	223
6.4 SQL 数据更新功能.....	192	7.7.2 全局变量.....	224
6.4.1 插入数据.....	192	7.8 运算符.....	226
6.4.2 修改数据.....	193	7.8.1 算术运算符.....	227
6.4.3 删除数据.....	193	7.8.2 赋值运算符.....	227
6.5 建立和使用视图.....	194	7.8.3 位运算符.....	228
6.5.1 视图的建立与删除.....	194	7.8.4 比较运算符.....	228
6.5.2 查询视图.....	195	7.8.5 逻辑运算符.....	229
6.5.3 更新视图.....	196	7.8.6 字符串串联运算符.....	230

7.8.7 一元运算符	230	8.6.5 查看存储过程相关性	260
7.8.8 运算符的优先顺序	231	8.6.6 存储过程的修改	261
7.9 数据类型转换	232	8.6.7 重命名存储过程	262
7.9.1 数据类型的优先顺序	232	8.6.8 删除存储过程	262
7.9.2 数据类型的转换级别	233	8.6.9 使用企业管理器来创建、修改、 重命名和删除存储过程	263
7.9.3 数据类型的转换方式	233	8.7 触发器	264
7.10 小结	233	8.7.1 触发器的类型和使用触发器 的优点	264
7.11 思考题	233	8.7.2 创建触发器	264
第 8 章 Transact-SQL 的高级应用	234	8.7.3 修改触发器	267
8.1 注释语句	234	8.7.4 删除触发器	268
8.2 批处理	235	8.7.5 触发器的其他操作	268
8.3 函数	235	8.7.6 使用企业管理器来管理 触发器	268
8.3.1 数学函数	235	8.8 小结	270
8.3.2 日期和时间函数	236	8.9 思考题	271
8.3.3 字符串函数	238	第 9 章 数据库索引与查询优化	272
8.3.4 用户自定义函数	239	9.1 索引的概念	272
8.4 控制流语句	240	9.2 索引类型	273
8.4.1 BEGIN...END 语句	240	9.2.1 聚簇索引	274
8.4.2 IF...ELSE 语句	241	9.2.2 非聚簇索引	275
8.4.3 CASE 语句	243	9.3 创建索引	276
8.4.4 WHILE 循环控制语句	245	9.3.1 使用 Transact-SQL 创建索引	276
8.4.5 GOTO 跳转语句	247	9.3.2 使用企业管理器创建索引	279
8.4.6 WAITFOR 语句	248	9.3.3 通过索引向导创建索引	280
8.4.7 RETURN 语句	248	9.4 删除索引	282
8.5 游标	249	9.4.1 使用 Transact-SQL 删除索引	282
8.5.1 声明游标	249	9.4.2 使用企业管理器删除索引	283
8.5.2 打开游标	251	9.5 对计算列和视图进行索引	283
8.5.3 游标的定位	251	9.5.1 对计算列进行索引	283
8.5.4 使用游标修改数据	252	9.5.2 索引视图	285
8.5.5 关闭游标	253	9.6 查询优化	287
8.5.6 释放游标	253	9.6.1 查询优化器	288
8.5.7 游标函数	253	9.6.2 优化步骤	288
8.5.8 游标使用的综合示例	255	9.6.3 优化策略	288
8.6 存储过程	256		
8.6.1 创建存储过程	257		
8.6.2 执行存储过程	258		
8.6.3 查看存储过程的定义	259		
8.6.4 查看存储过程信息	260		

9.7 索引与性能	290	10.6.2 批量登记恢复	321
9.7.1 索引选择	290	10.6.3 简单恢复	321
9.7.2 索引设计准则	291	10.6.4 设置恢复方式	321
9.8 小结	293	10.7 超大数据库问题	322
9.9 思考题	294	10.8 小结	322
第 10 章 数据库的备份与恢复	295	10.9 思考题	323
10.1 概述	295	第 11 章 SQL Server 2000 数据库 开发实例	324
10.1.1 备份与恢复的概念	295	11.1 SQL Server 数据库的快速连接	324
10.1.2 故障的种类	295	11.1.1 创建数据库	324
10.2 备份类型	296	11.1.2 数据引擎与数据 绑定控件	325
10.3 备份设备	299	11.1.3 连接 SQL Server 2000 数据库	326
10.3.1 磁盘备份设备	299	11.2 数据库记录的添加、 删除和修改	328
10.3.2 磁带备份设备	299	11.2.1 界面设计	328
10.3.3 命名管道备份设备	299	11.2.2 编写事件代码	328
10.3.4 备份设备名	300	11.2.3 测试运行程序	330
10.3.5 创建数据库备份设备	300	11.3 使用 SQL 语句	330
10.4 执行备份操作	302	11.3.1 界面设计	330
10.4.1 使用 Enterprise Manager 进行备份操作	302	11.3.2 编写事件代码	330
10.4.2 创建备份向导	305	11.3.3 测试运行程序	331
10.4.3 使用 Transact-SQL 语句 备份数据库	306	11.4 动态连接 SQL Server 2000 数据库	331
10.4.4 数据库备份策略	309	11.4.1 ADO 的属性与方法	332
10.5 数据库的恢复	310	11.4.2 ADO 动态连接数据库的 一般方法	336
10.5.1 自动恢复	311	11.5 开发有存储过程的数据库程序	339
10.5.2 手工恢复	311	11.5.1 后台存储过程的编写	340
10.5.3 恢复进程运行期间的 SQL Server 的活动	311	11.5.2 存储过程与前台 程序的连接	340
10.5.4 取得备份信息	312	11.5.3 程序的运行	341
10.5.5 使用 Enterprise Manager 恢复数据库和日志文件	313	11.6 小结	341
10.5.6 使用 Transact-SQL 语句恢复 数据库和日志文件	315	11.7 思考题	342
10.5.7 恢复文件或文件组	318	参考文献	343
10.5.8 恢复系统数据库	319		
10.6 恢复模板	320		
10.6.1 完全恢复	321		

第 1 章 绪 论

20 世纪 60 年代末, 数据库技术开始发展起来。如今, 数据库技术已经在各行各业得到了广泛的应用, 成为存储、使用和更新信息资料的主要手段, 从小型单项事务处理系统到大型信息系统, 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/CAI)、计算机集成制造系统(CIMS)、办公信息系统(OIS)、地理信息系统(GIS)等, 越来越多新的应用领域采用数据库存储并处理信息资源, 产生了巨大的经济效益和社会效益。数据库的建设规模、数据库信息量的大小和使用频度已成为衡量一个国家、一个地区、一个单位信息化程度的重要标志。因此, 数据库课程是计算机科学与技术专业、信息管理专业的一门重要课程。

1.1 数据库技术概述

1.1.1 几个基本概念

在系统地介绍数据库的基本知识之前, 首先给出几个常用的术语和基本概念。

1. 数据

描述事物的符号记录称为数据(Data)。这些符号包括数字、文字、图形、图像、声音等, 这些符号组成的集合称为记录。数据可以有多种描述形式, 数据的描述形式不能完全表达其内容, 需要经过解释。数据的解释说明称为语义, 数据与其语义是不可分割的。

2. 数据库

数据库(Database)是指长期存储在计算机内的、有组织的、可共享的数据集合。也就是说数据库是存放数据的仓库, 该仓库建立在计算机存储设备上, 且数据是按一定的格式存放的。数据库的主要作用是借助计算机保存、管理大量的数据, 以方便充分地利用资源。

数据库的用户分为 3 种。

- 数据库管理员: 他们使用数据库管理系统提供的工具软件对数据库进行各种维护和管理, 以便使其更好地为其他用户服务。
- 应用程序员: 他们使用数据库语言和程序设计语言编写使用 and 操纵数据库中数据的程序。
- 非程序员用户: 他们通过联机终端设备, 使用数据查询语言访问数据库。

3. 数据库管理系统

数据库管理系统(Database Management System, 简称 DBMS)是对数据库进行管理和操

作的系统，它是建立在操作系统基础上的，是位于操作系统与用户之间的一种数据管理软件，负责对数据库的数据进行统一的管理和控制。用户发出的或应用程序中的各种对数据库的操作都要通过 DBMS 来执行。

4. 数据库系统

数据库系统(Database System, 简称 DBS)是引入数据库后的计算机系统。它一般由数据库、数据库管理系统及其开发工具、应用系统、数据库管理员和用户构成，如图 1.1 所示。

数据的重要价值是使用而不是收集，数据库系统就是为了方便使用数据而设计的。它对数据进行集中控制，能有效地维护和利用数据。

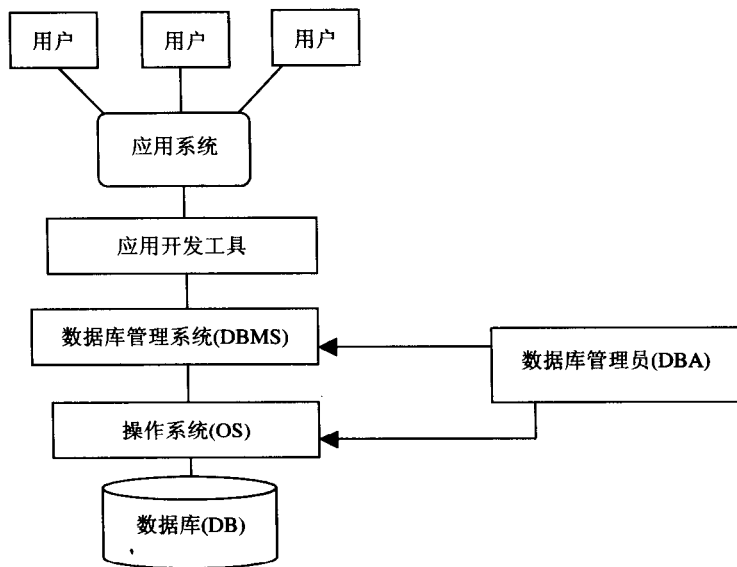


图 1.1 数据库系统的构成

1.1.2 数据库系统软、硬件的关系

数据库系统由支持系统的计算机硬件设备，数据库、相关的计算机软件系统，以及开发管理数据库系统的人员组成。

1. 数据库系统对硬件的要求

数据库系统是建立在计算机硬件基础之上的，必须有相应的硬件资源支持才能运行。支持数据库系统的计算机硬件资源包括 CPU、内存、外存及其他设备。外部设备主要包括某个具体的数据库系统所需的数据通信设备和数据输入/输出设备。

数据库系统数据量大、数据结构复杂、软件内容多，因而要求其硬件设备能够快速处理它的数据。因此在配置数据库系统的硬件时，要满足以下 3 个方面的要求。

(1) 内存要尽量大

数据库系统的软件结构构成复杂,包括操作系统、DBMS、应用程序及数据库。工作时它们都需要占用一定的内存作为程序工作区或数据缓冲区,所以数据库系统与其他计算机相比需要更大的内存。内存的大小对数据库系统性能的影响非常明显,内存大可以建立较多、较大的程序工作区或数据缓冲区,以管理更多的数据文件和控制更多的程序进程,进行较复杂的数据管理和更快地进行数据操作。并且每种数据库系统对内存有一定的要求,如果内存达不到要求,便无法正常运行。

(2) 外存要尽量大

计算机外存主要有软磁盘、磁带和硬盘,其中硬盘为主要的内存设备。数据库系统要求硬盘的容量尽量大,这有三个优点:

- 可以为数据文件和数据库软件提供足够的空间,满足数据和程序的存储要求。
- 可以为系统的临时文件提供存储空间,保证系统能正常运行。
- 数据检索时间缩短,从而加速数据存取速度。

(3) 计算机的数据传输速度要快

由于数据库的数据量大而操作复杂度不大,数据库运行时要经常进行内、外存之间的交换操作,这要求计算机不仅有较强的运算能力,而且数据存取和数据交换的速度要快。对一般的系统来说,计算机的运行速度是由 CPU 和数据 I/O 的传输速度两者决定的,但对于数据库系统来说,数据 I/O 的传输速度是提高运行速度的关键问题,提高数据传输速度是提高数据库系统效率的重要指标。

2. 数据库系统的软件组成

(1) 操作系统

操作系统是所有计算机软件运行的基础,在数据库系统中它支持 DBMS 及主语言系统的工作。

(2) DBMS 和主语言系统

DBMS 是为定义、建立、维护、使用及控制数据库而提供的有关数据管理的系统软件。

主语言系统是为应用程序提供诸如程序控制、数据输入/输出、功能函数、图形处理、计算方法等数据处理功能的系统软件。应用系统的设计与实现,需要将 DBMS 和主语言系统结合才能完成。

(3) 应用开发工具软件

应用开发工具是 DBMS 为应用开发人员和最终用户提供的高效率、多功能的应用生成器等各种软件工具,如报表生成器、表单生成器、查询和视图设计器等,它们为数据库系统开发提供了良好的环境和有效的帮助。

(4) 应用系统及数据库

数据库应用系统包括为特定应用环境建立的数据库、开发的各类应用程序及编写的文档资料,它们是一个有机整体。数据库应用系统涉及到各个方面,如人工智能和计算机控制等。通过运行数据库应用系统,可以实现对数据库中数据的维护、查询、管理和处理操作。

3. 数据库系统的人员组成

数据库系统人员由软件开发人员、软件使用人员和软件管理人员组成。其中软件开发人员包括系统分析员、系统设计员及程序设计员，他们的主要职责是开发数据库系统；软件使用人员是数据库最终用户，他们利用功能菜单、表格及图形用户界面等实现数据查询及数据管理工作；软件管理人员是数据库管理员(DataBase Administrator, 简称 DBA)，他们负责全面的管理并控制数据库系统。数据库系统与计算机软、硬件的关系如图 1.2 所示。

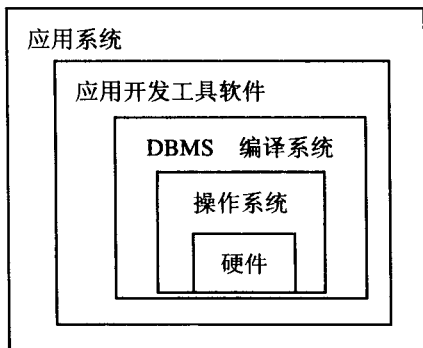


图 1.2 数据库系统与软、硬件关系

1.1.3 数据管理技术的产生和发展

数据管理经过了 3 个阶段：人工管理阶段、文件管理系统阶段和数据库系统阶段。

1. 人工管理阶段(20 世纪 50 年代)

年代特征：以科学计算为主。存储设备是纸带、卡片、磁带等，无操作系统，无管理数据的软件，数据处理方式是批处理。

管理特征：数据不保存，应用程序管理数据，数据不共享，数据不具有独立性。该阶段计算机系统只提供输入/输出操作，数据的逻辑组织与物理组织相同。如图 1.3 所示，设计人员要考虑数据存储方式和组织方法，当数据的物理组织或存储设备改变时，应用程序必须改变，在一般情况下，一组数据只能对应一个应用程序，因此，造成大量的数据重复。

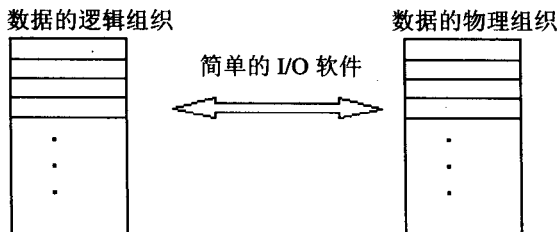


图 1.3 人工管理的数据组织

总之，人工管理阶段的缺点是数据不能保存，没有软件系统对数据进行管理，应用程

序依赖于数据的物理组织与存储形式，数据冗余度大、独立性差。

2. 文件管理系统阶段(20 世纪 50 年代后期~60 年代中期)

年代特征：已有磁盘、磁鼓等直接存取存储设备；操作系统中已有了专门的数据管理软件，称为文件系统；处理方式不仅有批处理，而且能够联机实时处理。

管理特征：数据可以长期保存；由文件系统管理数据(文件名访问、按记录进行存取)；数据共享性差、冗余度大(一个文件对应一个应用)，数据独立性差。

文件管理系统是用户和数据文件之间的一个接口，它负责对数据文件进行专门的管理，其主要特点是：

(1) 数据的逻辑组织与物理组织之间可以有区别，做到了数据的物理独立性，当数据的物理组织改变时，不影响逻辑组织形式。物理组织与逻辑组织之间由专门的存储方法实现转换。用户只需考虑数据的逻辑表示形式，而不用考虑数据的物理组织方法，即数据的物理存储形式的改变不影响其逻辑形式。

(2) 数据可以以文件的形式存储在外存储器上，实现了以文件为单位的数据共享。用户可以对数据反复进行修改、插入、删除和查询等操作。

(3) 数据文件的逻辑组织与应用程序紧密相关。当数据结构需要修改时，应用程序也要做相应的修改；反之当应用程序需修改和扩充时，数据结构也要做相应的改变，这对数据的维护是非常不便的。数据文件的组织形式有多种：顺序文件、索引文件和直接存取文件等，如图 1.4 所示。

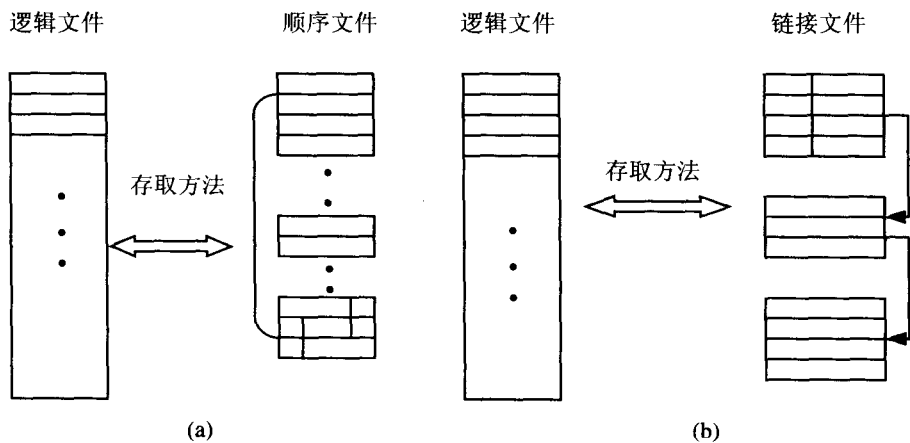


图 1.4 文件管理系统的数据组织

3. 数据库系统阶段(20 世纪 60 年代后期至今)

年代特征：使用大容量磁盘，硬件价格下降，软件价格上升，联机实时处理要求高，开始提出和考虑分布处理。

管理特征：用 DBMS 管理数据，以解决多用户、多应用共享数据的需求，满足更多应用。

随着计算机的广泛应用，人们对数据管理提出了更高的要求，包括：数据共享，更完全的数据独立性，集中统一的数据管理方式，应用系统开发简便，维护容易等。数据库技

术，正是为了满足上述要求而开发的。它提供了广泛的数据共享，高度的数据独立性，最小的数据冗余度，方便的用户接口，集中统一的数据管理等。

1.1.4 数据库系统的特点

1. 数据是结构化的

这是与文件系统的根本区别。目的是节省空间、增强灵活性大可以为多个应用提供服务。

在数据库系统中不仅要考虑某个应用的数据结构，还要考虑整个组织的数据结构，以便为各部门的管理提供必要的记录。

例如：一个学校的管理信息系统中，不仅要考虑学生的人事管理，还要考虑学籍管理、选课管理等，可以按图 1.5 所示方式为该校的管理信息系统组织学生数据。

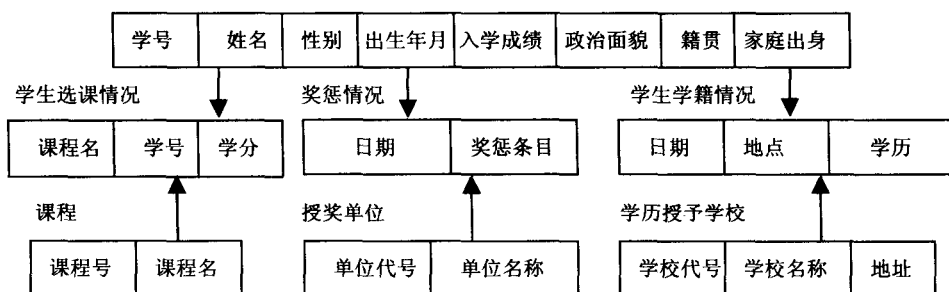


图 1.5 适应多种管理的学生数据记录

这种数据组织方式为多种管理提供必要的记录，使学校的学生数据结构化了。这就要求在描述数据时不仅要描述数据本身，还要描述数据之间的联系。

数据库系统实现整体数据的结构化，这是数据库的主要特征之一。

在数据库系统中，不仅数据是结构化的，而且存取数据的方式也很灵活，可以存取数据库中的某一个数据项、一组数据项、一个记录或一组记录。

2. 尽可能减少数据的冗余度

数据库系统只能尽量减小，而不能消除数据的冗余。这是因为满足某些要求时，同一数据的多次存储是必须的，系统会对这些冗余进行控制，保证不会因此引起数据不一致性。

3. 实现广泛的数据共享

- 某个应用所使用的只是数据库的一个子集，不同的子集可以任意重叠。
- 不同的用户可以并发地访问同一数据。
- 具有广泛的适应性，有多种语言的接口。

4. 保证数据的安全可靠

- 确保数据的安全存取。系统只对有权使用数据的用户授有限使用权，任何一个用

户都不能无限制地使用数据库中的数据。系统提供一套有效的安全性检查功能和控制设施,确保数据的合法使用。

- 保证数据的完整一致。数据完整性在系统应用中是十分重要的,如果某个应用程序破坏了数据的完整性,使其他的应用程序使用了不正确的数据可能产生错误的处理结果,甚至会造成重大的经济损失。因此,为了保证用户所使用的数据是正确、合理和相容的,系统提供了数据完整性约束条件和控制机制。
- 并发控制。不同的应用程序同时访问数据库,就有可能使数据受到损坏而失去完整性。系统提供的并发控制机制,就是用来排除和避免损坏数据的一致性,保证数据库的完整、准确。
- 故障的发现、排除与系统的恢复。数据库系统在运行时,随时都会受到局部或全局性的破坏。系统提供一套完整的中断和后援方案,确保能及时发现并排除故障。

5. 保证数据独立性

当数据库系统的物理性质发生变化(如更换存储设备、改变组织方法等)和逻辑性质发生变化(如改变模式和外模式等)时,都不用修改应用程序。

6. 实现标准化

数据库对数据的集中控制管理便于实现数据的标准化。标准化的实施,有利于行业间、国家间的信息交流与技术协作。

1.2 数据库系统的体系结构

数据库系统结构可以从不同的角度来认识。在数据库系统中,用户看到的数据与计算机中存放的数据是不同的,当然这两种数据之间是有联系的。它们之间实际上是经过了两次变换的(即二级映射):第一次是系统为了减少冗余,实现数据共享,对所有用户的数据进行了综合,抽象成为一个统一的数据视图;第二次是为了提高存取效率,改善性能,将若干个数据视图集合而成为全局视图,并将其数据按照物理组织的最优形式存放。

用户使用的数据视图称为外模型,又叫子模型,外模型是一种局部数据逻辑视图,它表示了用户所理解的实体、实体属性和实体联系。全局的数据逻辑视图称为概念模型,简称为模型,它是数据库管理员所看到的实体、实体属性和实体间的联系。数据存储的模型称为内模型,是数据存放的具体形式。

数据库系统分为3层:外层、概念层和内层,分别对应外模型、概念模型和内模型。用户只能看到外层,即用户级,其他两层是看不到的。外模型可以有多个,而概念模型(概念级)和内模型(物理级)分别只有一个,内模型是整个数据库的最底层。

数据库管理系统尽管产品很多,它们支持不同的数据模型,使用不同的数据库语言,建立在不同的操作系统上,数据结构也各不相同,但它们在体系结构上通常都具有相同的特征,即采用三级模式结构并提供两级映像功能。

1.2.1 数据库系统模式概念

数据模型中有“型”和“值”的概念。型是指对某一类数据的结构和属性的说明，值是型的一个具体赋值。例如：学生记录定义是一个型，而具体的某一条记录则是一个值。

模式：是数据库中全体数据的逻辑结构和特征的描述，它只涉及到型的概念，不涉及具体的值。

实例：模式的一个具体值称为模式的一个实例，它涉及到具体的值。同一个模式可以涉及很多实例。

模式与实例的区别：模式相对稳定，实例是相对变动的。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

掌握数据库的三级结构及其联系与转换关系是深入了解数据库的关键所在。模式是用数据描述语言精确地定义数据模型的程序。定义外模型的模式称为外模式，又称外模式，用外模式定义语言来定义。定义概念模型的模式称为概念模式，简称模式，用模式定义语言来定义。定义内模型的模式称为内模式，又称存储模式，用设备介质语言来定义。

数据库系统的三级结构是指数据库系统由外模式、模式、内模式三级构成，如图 1.6 所示。

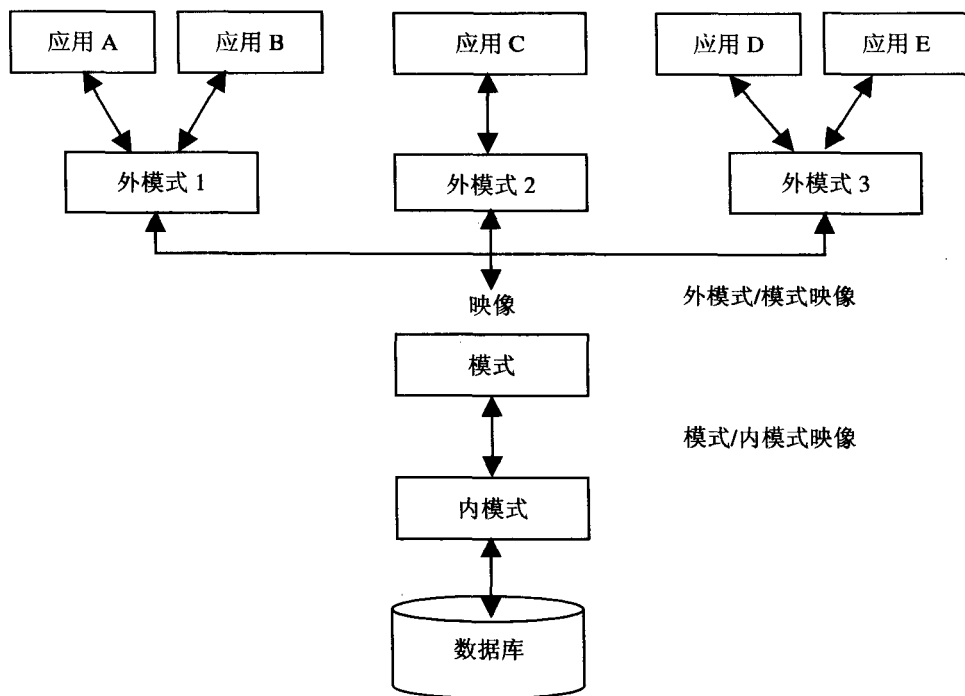


图 1.6 数据库系统的三级模式结构