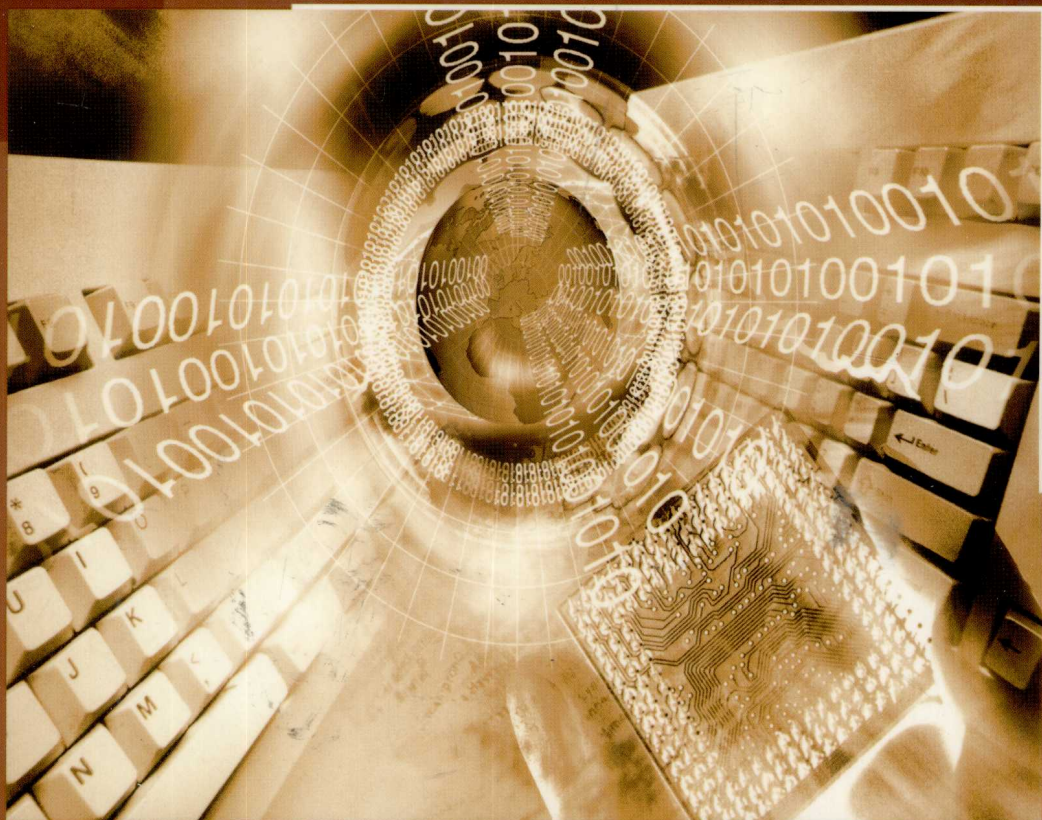


Visual FoxPro

数据库管理系统教程



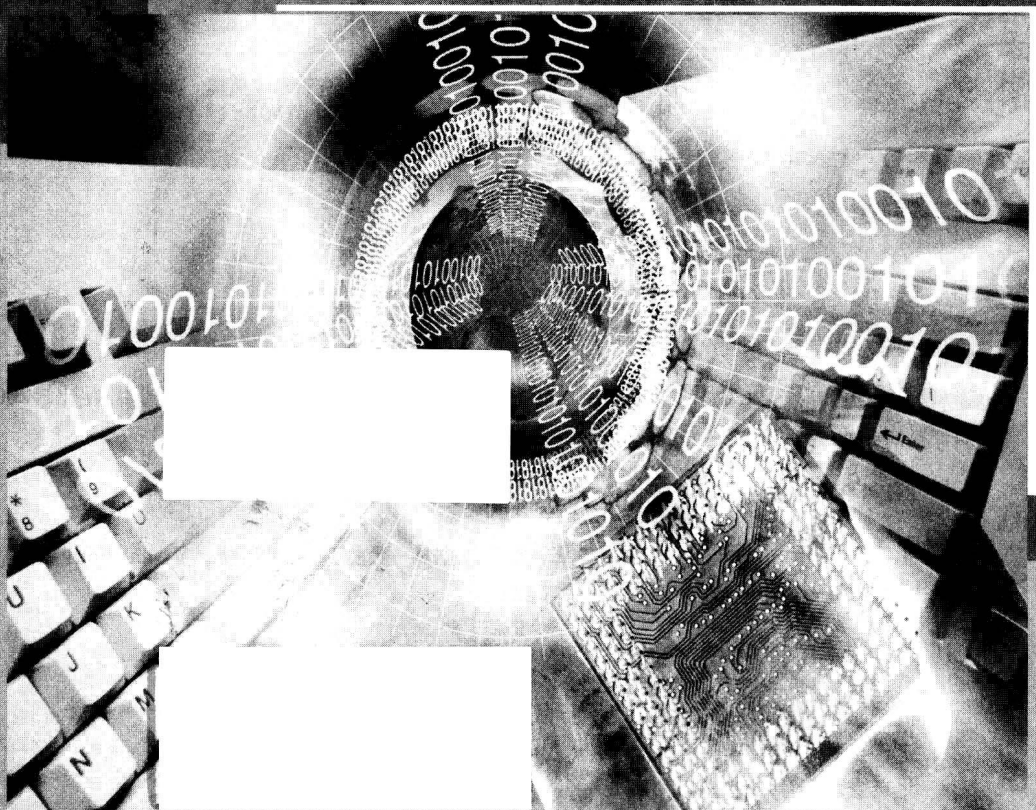
程 玮 陆 晶 主 编
杨晓红 李 静 副主编

本书为教师配有电子教案

普通高等院校计算机课程规划教材

Visual FoxPro

数据库管理系统教程



程 玮 陆 晶 主 编
杨晓红 李 静 副主编



机械工业出版社
China Machine Press

本书以目前广泛使用的Visual FoxPro 8.0中文版为主,系统地介绍了数据库基础知识、关系型数据库管理系统的基本原理、数据库设计的基本方法、数据库的基本操作、关系型数据库标准查询语言SQL、结构化程序设计、面向对象的可视化编程方法以及数据库应用系统的开发等。附录部分提供了Visual FoxPro的命令、函数、文件类型与扩展名,基类和控件对象的名称、属性、事件、方法等。

本教程以教育部提出的非计算机专业计算机基础教学的三个层次为依据,根据高等学校计算机应用技术基础层次的教学基本要求,结合作者多年从事计算机应用技术基础教学的经验编写完成,可以作为高等学校非计算机专业本、专科学生的教材。本书的编写也考虑到计算机等级考试的基本要求,覆盖了“全国计算机等级考试二级Visual FoxPro”考试大纲的主要内容,因此也可以作为培训教材或者计算机等级考试参考书。对于计算机应用人员本书也是一本较好的参考书。

版权所有,侵权必究。

本书法律顾问 北京市展达律师事务所

图书在版编目(CIP)数据

Visual FoxPro数据库管理系统教程 / 程玮, 陆晶主编. —北京: 机械工业出版社, 2008.1
(普通高等院校计算机课程规划教材)

ISBN 978-7-111-22967-4

I. V… II. ①程… ②陆… III. 关系数据库—数据库管理系统, Visual FoxPro—程序设计—高等学校—教材 IV. TP311.138

中国版本图书馆CIP数据核字(2008)第007955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: 姚 蕾

北京京北印刷有限公司印刷

2008年12月第1版第2次印刷

184mm × 260mm · 15.75印张

标准书号: ISBN 978-7-111-22967-4

定价: 26.00元

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换
本社购书热线电话: (010) 68326294

前 言

当今世界已进入信息时代,以计算机技术为代表的高新技术飞速发展,计算机的应用正向人类生产、生活的各个领域迅速渗透。是否掌握一定的计算机应用能力已成为衡量新世纪技术人才的重要标准之一。

信息管理是计算机应用最广泛的领域之一。数据库技术是当前计算机信息管理采用的主要技术手段。为了适应数据库技术的新发展,满足高等院校非计算机专业计算机应用技术教育的需要,我们组织编写了这本教材。

Visual FoxPro是小型数据库管理系统的代表。它以强大的功能,完整而又丰富的工具,较高的处理速度和友好的界面等特点,受到广大用户的欢迎。Visual FoxPro提供了一个集成化的系统开发环境,它使数据的组织与操作变得简单方便。它不仅支持传统的结构化程序设计,而且支持面向对象的程序设计。利用它提供的功能强大的可视化程序设计工具和向导等,用户可以快速地创建数据库、表,创建表单、菜单、查询,打印输出报表等。与其他数据库管理系统比较,Visual FoxPro的最大特点是自带编程工具,将数据库管理系统和程序设计语言有机地结合在一起,很适合初学者学习。特别是各类非计算机专业的学生,学习计算机课程的学时有限,又要培养他们具备一定的计算机应用能力,采用Visual FoxPro作为数据库管理系统和程序设计语言的教学软件已成为一个较好的选择。

本教程以教育部提出的非计算机专业计算机基础教学的三个层次为依据,根据高等学校计算机应用技术基础层次的教学基本要求,结合我们多年从事计算机应用技术基础教学的经验,并参考国内外大量文献资料后编写完成。全书共分九章和八个附录,以目前广泛使用的Visual FoxPro 8.0中文版为主,系统地介绍了数据库基础知识、关系型数据库管理系统的基本原理、数据库设计的基本方法、数据库的基本操作、关系型数据库标准查询语言SQL、结构化程序设计、面向对象的可视化编程方法以及数据库应用系统的开发等。附录部分提供了Visual FoxPro的命令、函数、文件类型与扩展名,基类和控件对象的名称、属性、事件、方法等,方便读者查阅。

本教程可以作为高等学校非计算机专业本、专科学生的教材。带有“*”号的章节为选学内容。教学过程中,教师可根据学制、专业、学时数、教学要求等实际情况对讲授内容进行取舍。

本书的编写也考虑到计算机等级考试的基本要求,覆盖了“全国计算机等级考试二级Visual FoxPro”考试大纲的主要内容,因此也可以作为培训教材或者计算机等级考试参考书。对于计算机应用人员,本书也是一本较好的参考书。

为了方便教学和读者进行上机操作练习,作者还同步编写了《Visual FoxPro数据库管理系统教程学习与实验指导》一书,作为与本教程配套使用的实验教材。其中的学习指导、习题及参考答案等,为读者自学提供了方便。另外,本书还有配套的教学课件可以提供教师使用。

本书由程玮教授、陆晶副教授任主编,杨晓红教授、李静老师任副主编。其中程玮编写了第1章,孙延民编写了第2章,陆晶编写了第3章和第4章,杨晓红编写了第5章和第6章,李

静编写了第7章，姚贵敏编写了第8章，王福滨编写了第9章，陈丽萍编写了附录I~附录VIII。全书最后由程玮总纂统稿，陆晶校对。

本书在编写过程中，始终得到山东财政学院教务处、教材科等各级领导的关心与支持，计算机信息工程系各教研室、系办公室、学院实验中心的老师也给予了大力支持和帮助。南京大学徐洁磐教授详细审阅了本教材的编写大纲和全部书稿，提出了许多宝贵的意见和建议。机械工业出版社的编辑也为本书的顺利出版做了许多工作，在此向他们表示衷心的感谢。

本书的编写力求做到概念清晰、结构合理、层次分明、深入浅出、通俗易懂、删繁就简、详略得当。但因编写时间的仓促和编者学识水平所限，书中难免有疏漏和错误之处，敬请广大读者不吝赐教，批评指正。

需要教学课件的教师，请与编者联系。Email: chengwei@sdfi.edu.cn

编 者

2007年11月

目 录

前言

第1章 数据库基础知识	1
1.1 数据库和数据库系统	1
1.1.1 数据、信息和数据处理	1
1.1.2 数据库和数据库系统	5
1.2 关系型数据库	6
1.2.1 数据模型概述	6
1.2.2 关系型数据库	9
1.3 数据库设计基础	20
1.3.1 数据库设计的基本原则	20
1.3.2 数据库设计的过程	21
1.4 Visual FoxPro系统概述	23
1.4.1 Visual FoxPro简介	23
1.4.2 Visual FoxPro的用户界面	26
1.4.3 Visual FoxPro命令简介	32
1.4.4 Visual FoxPro项目管理器	34
1.4.5 Visual FoxPro向导、设计器和 生成器	39
第2章 Visual FoxPro语言基础	41
2.1 数据类型	41
2.2 常量、变量与函数	42
2.2.1 常量	42
2.2.2 变量	44
2.2.3 函数	49
2.3 运算符与表达式	49
2.3.1 算术运算符和数值表达式	49
2.3.2 字符串运算符及其表达式	50
2.3.3 日期时间运算符及其表达式	50
2.3.4 关系运算符及其表达式	51
2.3.5 逻辑运算符及其表达式	53
2.4 常用函数	54
2.4.1 数值运算函数	54
2.4.2 字符串操作函数	56

2.4.3 日期时间函数	58
2.4.4 数据类型转换函数	59
2.4.5 测试函数	60
*2.5 常用Visual FoxPro系统设置命令	62
第3章 数据库与数据表的基本操作	65
3.1 数据库的基本操作	65
3.1.1 创建数据库	65
3.1.2 打开、关闭和设置数据库	66
3.1.3 修改、删除数据库	68
3.2 数据库表的基本操作	68
3.2.1 数据库表结构的设计	69
3.2.2 创建数据库表	70
3.2.3 打开与关闭数据库表	72
3.2.4 修改数据库表结构和属性	73
3.3 数据库表记录的基本操作	75
3.3.1 使用浏览窗口	75
3.3.2 添加记录	77
3.3.3 显示记录	78
3.3.4 修改记录	79
3.3.5 删除和恢复记录	81
3.3.6 定位记录指针	83
3.3.7 其他操作	84
3.4 索引	85
3.4.1 基本概念	85
3.4.2 创建索引	86
3.4.3 使用索引	89
3.4.4 索引定位命令	90
3.4.5 删除索引命令	90
3.5 自由表的操作	91
3.5.1 自由表与数据库表	91
3.5.2 自由表和数据库表的转换	91
3.6 多数据表操作	92
3.6.1 工作区的概念	92

3.6.2 不同工作区的互访	92	6.3 程序设计辅助命令	130
3.6.3 数据表之间的关联	93	6.3.1 常用输入输出命令	130
3.6.4 永久关系与参照完整性设置	94	6.3.2 其他常用程序命令	132
第4章 数据查询与视图	97	6.4 程序的基本控制结构	134
4.1 查询	97	6.4.1 顺序结构	134
4.1.1 创建查询	97	6.4.2 分支结构	134
4.1.2 使用查询	102	6.4.3 循环结构	137
4.2 视图	102	6.5 子程序、过程文件与自定义函数	141
4.2.1 视图的概念	102	6.5.1 子程序	141
4.2.2 创建视图	103	6.5.2 过程与过程文件	142
4.2.3 连接与远程视图	104	6.5.3 自定义函数	144
4.2.4 参数化视图	105	6.5.4 变量的作用域	145
4.2.5 视图与数据更新	106	6.6 程序调试	148
4.2.6 使用视图	107	6.6.1 程序调试的一般方法	148
第5章 关系数据库结构化查询语言SQL	108	6.6.2 Visual FoxPro调试器的使用	148
5.1 SQL语言概述	108	第7章 面向对象的程序设计基础	154
5.2 SQL数据定义功能	109	7.1 面向对象的概念	154
5.2.1 创建数据库和表	109	7.1.1 对象与类	154
5.2.2 表的修改	110	7.1.2 属性、方法和事件	155
5.2.3 表的删除	111	7.1.3 Visual FoxPro中的基类	155
5.3 SQL数据查询功能	111	7.2 表单的基本操作	156
5.3.1 基本查询	112	7.2.1 创建表单	156
5.3.2 条件查询	113	7.2.2 操作表单	160
5.3.3 连接查询	114	7.2.3 表单属性、事件和方法	162
5.3.4 嵌套查询	116	7.2.4 表单数据环境的设置	165
5.3.5 数据统计查询	119	7.3 表单常用控件	166
5.3.6 查询结果排序	122	7.3.1 控件的基本操作	167
5.3.7 查询结果处理	123	7.3.2 标签控件	167
5.3.8 视图语句	123	7.3.3 按钮类控件	167
5.4 SQL数据修改功能	124	7.3.4 文本框和编辑框控件	169
5.4.1 插入数据	124	7.3.5 选项组和复选框控件	172
5.4.2 更新数据	125	7.3.6 列表框和组合框控件	174
5.4.3 删除数据	125	7.3.7 表格控件	178
第6章 程序设计基础	126	7.3.8 页框控件	181
6.1 结构化程序设计概述	126	7.3.9 计时器控件	183
6.1.1 程序的概念	126	7.3.10 微调控件	183
6.1.2 程序设计和算法	126	7.3.11 图像控件	183
6.2 程序文件的建立、编辑与执行	128	7.3.12 容器控件	184
6.2.1 程序文件的建立与编辑	128	7.4 Visual FoxPro中类的建立	184
6.2.2 程序文件的调用执行	129		

第8章 菜单、报表与标签设计	188
8.1 菜单设计	188
8.1.1 创建下拉式菜单	188
8.1.2 菜单设计器	189
8.1.3 “快速菜单”命令	193
8.1.4 创建快捷菜单	193
8.2 报表设计	195
8.2.1 创建报表	195
8.2.2 设计报表	197
8.2.3 报表控件	199
8.2.4 报表输出	201
8.3 标签设计	203
8.3.1 使用标签向导	203
8.3.2 使用标签设计器	204
8.3.3 标签输出	205
第9章 应用系统开发简介	206
9.1 应用系统开发的基本步骤	206
9.2 “教师管理信息系统”的开发简介	207
9.2.1 系统需求分析	207

9.2.2 系统总体规划设计	207
9.2.3 数据库、表设计	208
9.2.4 应用系统功能模块设计	209
9.2.5 应用系统主控程序	210
9.3 “教师管理信息系统”应用程序简介	211
9.4 应用系统的连编	218
附 录	223
附录I Visual FoxPro 8.0命令索引表	223
附录II Visual FoxPro 8.0常用函数索引表	231
附录III Visual FoxPro 8.0常用文件类型 及其扩展名	233
附录IV Visual FoxPro 8.0的基类	234
附录V Visual FoxPro 8.0的对象	234
附录VI Visual FoxPro 8.0的对象属性	234
附录VII Visual FoxPro 8.0的对象事件	241
附录VIII Visual FoxPro 8.0的对象方法	242
参考文献	244

第1章 数据库基础知识

自从20世纪40年代中期诞生了具有巨大存储能力，可以高速、准确、自动地进行数值运算和逻辑运算的数字电子计算机以来，在短短的六十余年里，计算机技术日新月异地迅猛发展，计算机的应用领域不断扩展，几乎已渗透到整个人类经济活动和社会生活的各个方面。在计算机的众多应用中，最广泛、最直接、最普及、最重要的应用领域就是数据处理。

计算机技术的应用从单纯的科学计算发展到数据处理是一个划时代的转折。它使计算机从科学殿堂的神坛上走下来，成为广大的科技工作者和管理人员工作中不可缺少的得力助手，成为普通人工作、生活、娱乐的有用工具。计算机技术在数据处理中的应用和发展为人类社会进入信息时代打下了坚实的基础，提供了最有力的工具，并形成了一个计算机科学与技术的重要分支——数据库技术。

本章首先从数据库的基本概念出发，介绍一些最常用的数据库基础知识和术语。然后简要介绍Visual FoxPro数据库管理系统的一般概况。

1.1 数据库和数据库系统

在数据处理中，人们首先遇到的基本概念是数据和信息，它们是两个不同的术语，但又有着密不可分的联系。

1.1.1 数据、信息和数据处理

1.1.1.1 数据

数据(Data)指的是人们用于表达、描述、记录客观世界事物与现象属性的某种物理符号。数据不仅具有描述事物特性的具体数据内容，而且具有能够记录和存储的某种具体表现形式。

例如，可以用“王丽，女，1981年11月20日，1.68m，55kg，助教，1200元，照片(图像)”等数据分别表示“姓名，性别，出生年月日，身高，体重，职称，基本工资和相貌”等，则上述数据就描述出某个具体人的一些基本特征。如“女”是用汉字符记录、描述的性别特征；“1.68m，55kg”则是用十进制数字符号和国际标准单位记录、描述的身高、体重特征；照片则是用图像的形式记录、描述的相貌特征。

数据不仅包括数字、文字、字母和各种特殊符号等文字数据，还包括图形、图像、动画、影像、声音等各种多媒体数据。但使用最多、最基本的仍然是文字数据。

计算机能够接收、识别、存储、处理的数据是二进制数，因此，必须将各种形式的数据转换成二进制数据后才能用计算机对其进行加工处理。

1.1.1.2 信息

信息(Information)是客观世界事物与现象属性的反映，是经过加工处理，并对人类的客观行为产生影响的具有知识性的有用数据。

数据与信息在很多场合被认为是同义的，但实际上它们之间是有区别的。数据是信息的载体，是记录信息的符号，任何客观事物与现象的属性都可以用数据来表示。信息依靠数据

来表达,是对数据具体含义的解释。数据经过加工处理具有了知识性,并能对人类的活动产生决策作用,从而形成有用的数据——信息。

例如,某个人的“出生日期”是其不可改变的基本特征之一,称为原始数据。通过用某个具体的年月与出生日期进行相减这样的简单计算,可以得到其“年龄”信息。再根据年龄、性别、职称以及离退休年龄的有关规定,即可确定此人何时应当办理离退休手续。

1.1.1.3 数据处理

由于“数据”和“信息”这两个术语之间的密切联系,人们在使用中往往对这两个词不加区别,所以“数据处理”也称为“信息处理”。所谓数据处理,其真正的含义是为了产生信息而对原始数据进行的加工处理。数据处理通常包括数据的采集、接收、传递、转换、存储、整理、分类、排序、索引、统计、计算、检索等一系列的活动过程。

数据处理的目的是从大量的原始数据中获得人们所需的有用数据,为做出正确的决策提供依据。而数据处理的核心是数据管理。

计算机技术的发展和运用,使数据管理发生了划时代的变革。而数据库技术的出现与发展,又使数据管理跨入了一个崭新的阶段。随着计算机硬件、软件技术和应用范围的不断发展,计算机数据管理技术的发展也经历了人工管理、文件管理、数据库系统管理等几个阶段。

1. 人工管理阶段

人工管理阶段是指计算机技术应用于数据处理的初期。在20世纪50年代中期以前,外存储器只有卡片、纸带、磁带等非直接存取数据的存储设备,也没有专门管理数据的软件系统,用户只能直接在计算机裸机上进行操作,数据处理方式是批处理。在此阶段数据管理的主要特点是:

(1) 人工实施数据管理

因为没有管理数据的软件系统,程序设计人员不仅需要设计数据的逻辑结构,还要负责设计数据的物理结构,包括确定数据在计算机中的存储结构、存取方法和输入输出方式等,工作负担极重。

(2) 数据不能共享

因为数据与程序密切结合、不可分割,一组数据对应一个程序,所以数据不能在各程序之间互相传递。多个应用程序使用相同的数据时,只能各自定义,无法共享,这使程序之间存在着大量的冗余数据。

(3) 数据不能保存

因为数据由对其进行处理的程序自行携带,只是在进行数据处理时才将数据输入,所以一旦处理完毕,程序运行结束,数据即丢失而不能保存。

(4) 数据未结构化

当数据的逻辑结构或物理结构发生变化时,必须对应用程序做相应的修改以适应其改变,反之亦然,数据与程序的独立性差。

在人工管理阶段,应用程序与数据之间的关系如图1-1所示。

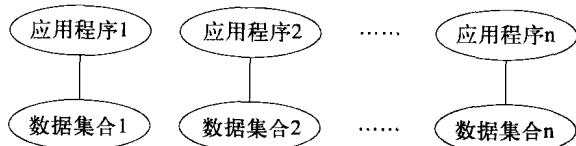


图1-1 人工管理阶段应用程序与数据间的对应关系

2. 文件管理阶段

20世纪50年代中后期至60年代中后期,可以随机访问、直接存取数据的磁盘成为计算机的主要外存储器,软件出现了高级程序设计语言和操作系统。在操作系统中包含了专门对外存储器中的数据进行管理的文件管理系统。在此阶段,数据管理的主要特点是:

(1) 数据管理由软件实现

数据由专门的软件(文件管理系统)进行管理。文件管理系统将数据组织成相互独立的数据文件,数据的结构、存取方法等均由文件管理系统负责,程序设计人员的负担大大减轻。

(2) 数据可以保存

数据以文件的形式组织起来,有了一定的独立性。数据和程序可以分开,能够长期保存。

(3) 数据共享性较差

在文件管理系统中,基本上是一个数据文件对应一个应用程序。当不同的应用程序使用的数据有相同部分时,也只能建立各自的数据文件,而不能实现相同数据的共享,造成数据的冗余度大。这不仅浪费了大量的存储空间,而且对数据进行修改和维护比较困难,容易造成数据的不一致性。

(4) 数据的结构化差

在文件内部,数据实现了在记录范围内的结构化,但各数据文件之间彼此独立、互不相关,在整体上并未实现结构化,因此数据文件的逻辑结构仅适用于与其对应的应用程序,而其他的应用程序则难以使用。若要扩充系统的应用,则必须修改数据文件的结构和与之对应的应用程序。所以,数据文件与应用程序之间仍然缺乏独立性。

在文件管理阶段,应用程序与数据之间的关系如图1-2所示。

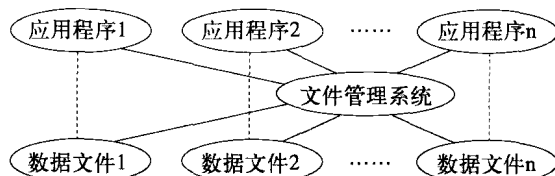


图1-2 文件管理阶段应用程序与数据间的对应关系

3. 数据库系统管理阶段

20世纪60年代以后,需要计算机管理的数据量急剧增加,对数据共享的要求日益增强,文件管理系统的数据库管理方法已无法满足应用系统的需求。为了实现计算机对数据的统一管理,达到数据共享的目的,出现了数据库系统。

数据库系统对数据的管理方式与文件管理系统不同,它面向整个系统,用整体的观点统一规划、组织、管理数据,形成一个数据管理中心,构建一个保存所有数据的数据库。数据库中的数据由专门的系统软件——数据库管理系统进行管理,并能满足所有用户的各种不同要求,供不同的用户共享。在数据库系统中,应用程序不再与一个孤立的数据文件相对应,而是通过数据库管理系统从数据库中取用自己所需的数据。

在数据库系统管理阶段,应用程序与数据之间的关系如图1-3所示。

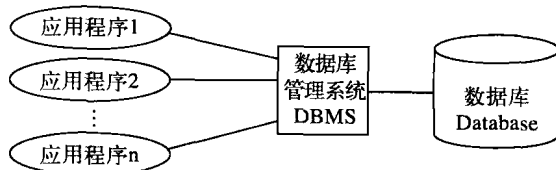


图1-3 数据库系统管理阶段应用程序与数据间的对应关系

在近四十年的发展史中,数据库技术也经历了三个发展阶段。

(1) 第一代数据库系统

第一代数据库系统又称非关系型数据库系统,其中包括“层次”与“网状”两种数据库系统。这两种数据库系统均以记录作为基本的数据结构,它们的差别主要体现在记录之间的联系上。层次型数据库系统中的记录之间只能有单线的联系,而网状数据库系统中的记录之间允许有两种或两种以上的联系。因为非关系型数据库系统的结构复杂,数据存取路径需要用户一一指定,因此数据查询和数据操纵困难,不易使用。随着关系型数据库系统的出现,非关系型数据库系统逐渐被淘汰。

(2) 第二代数据库系统

第二代数据库系统是关系型数据库系统(Relational DataBase System, RDBS),其问世于20世纪70年代。关系数据库技术以严格的关系数学理论为基础,采用二维表(关系)作为基本的数据结构。在关系型数据库系统中,每个关系都是一个二维表,用于表示实体以及实体之间的联系,描述实体的数据本身能够自然地反映出实体之间的联系。数据操作的对象和结果都是关系。

关系型数据库系统因其数学概念简单、清晰,数据描述的一致性,说明性的数据查询和数据操纵语言易学易懂、使用方便,而受到欢迎。它自20世纪70年代中期推向市场后,已发展成为应用最广泛的数据库系统。

(3) 第三代数据库系统

伴随着数据库系统应用范围的不断扩展,计算机新技术的不断涌现,人们提出了新的数据管理需求和认识世界、描述世界的新方法论,这些都推动了新一代数据库技术的研究与发展。

① 分布式数据库系统

分布式数据库系统(Distributed DataBase System, DDBS)是数据库技术与计算机网络技术紧密结合的产物。20世纪80年代以前,数据库系统多数是集中式的。随着计算机网络通信技术的迅速发展,为数据库提供了分布式的运行环境。数据库系统从主机/终端系统结构发展到客户/服务器(Client/Server)系统结构。目前,分布式数据库管理系统已进入市场应用,但尚不能完全解决异构数据和系统方面的许多问题。

② 多媒体数据库系统

随着多媒体应用的扩大,人们已经不满足于数据库系统仅能存储、处理数字、字符等文本数据,还需要存储、处理诸如声音、图形、图像、视频等多媒体信息。与数字、字符等格式化数据不同,声音、图形、图像、视频等非格式化数据具有数据量庞大、处理方法复杂等特点。多媒体数据库系统(Multimedia DataBase System, MDBS)既要实现对格式化的文本数据的存储和处理,还要实现对非格式化的多媒体数据的存储、处理。市场上现在已经有可以存储、处理一些如简短的声音片段、较小的图形、图像等多媒体信息的数据库系统,但还存在着许多要进一步研究解决的问题。

③ 面向对象数据库系统

面向对象是一种新的认识、描述事物的方法论。它起源于程序设计语言。面向对象的方法就是按照人们认识世界的习惯,对现实世界的客观事物(对象)进行最自然、最有效的抽象和表达,同时又以各种严格高效的行为规范和机制,实施对客观事物的有效模拟和处理,而且把对客观事物的表达(对象属性结构)和对它的操作处理(对象行为特征)结合成为一个有机的整体,使事物完整的内部结构和外部行为机制得到充分的反映和体现。

面向对象数据库系统(Object Oriented DataBase System, OODBS)是数据库技术与面向

对象程序设计相结合的产物，是面向对象方法在数据库领域的实现和应用。在关系模型的基础上进行扩展，引入面向对象特性的对象-关系型数据库系统（Object Relational DataBase System, ORDBS），已有实用产品推向市场。全新意义的面向对象数据库系统也有产品问世。

④ 其他第三代数据库系统

除了上述几种区别于第一、第二代数据库系统的新型数据库系统外，还有其他一些新一代数据库系统也在研究和发展之中。如可以在并行计算机系统上运行的并行数据库系统；与人工智能技术结合的知识数据库系统；面向特定应用领域的数据库新技术（如数据仓库、工程数据库、统计数据库、空间数据库等）都属于新一代数据库系统。

1.1.2 数据库和数据库系统

1.1.2.1 数据库

所谓数据库（Database），指的是以一定的组织方式存储在计算机存储设备上，与应用程序彼此独立、能为多个用户所共享、结构化的相关数据的集合。它是数据库系统的核心和管理对象。

在数据库中，数据按照一定的数据模型组织、描述和存储，具有较小的数据冗余度、较高的数据独立性、完整性和一致性，可为多个用户所共享。

1.1.2.2 数据库管理系统

数据库管理系统（DataBase Management System, DBMS）是为了数据库的建立、使用和维护而配置的软件系统。它建立在操作系统基础上，实现对数据库的统一管理和控制。DBMS既要向不同用户提供各自所需的数据，还要承担数据库的维护工作，保证数据库的安全性和完整性。

数据库管理系统的主要功能包括以下几个方面：

1. 数据定义功能

DBMS一般提供数据描述语言（Data Description Language, DDL）来定义数据库中的数据对象，如数据库、表的结构和视图等。

2. 数据操纵功能

DBMS还提供数据操纵语言（Data Manipulation Language, DML）实现对数据库的基本操作，如数据的插入、修改、删除和查询等。

3. 数据库的运行管理功能

这是DBMS运行时的核心功能，包括存取数据时根据约束条件对数据进行控制和检查，数据使用的并发控制，查询优化，以及发生故障后的系统恢复等。所有这些操作都必须在数据库管理系统的统一管理、统一控制下进行，以保证事务处理的正确性、数据库的有效性、数据的安全性和完整性。

4. 数据库的建立和维护功能

该功能包括数据库初始数据的输入、转换功能；数据库的转储、恢复功能；数据库的重组功能 and 系统的性能监视、分析功能。

数据库管理系统是数据库系统的重要组成部分，属于系统软件。

1.1.2.3 数据库系统

数据库系统（DataBase System, DBS）指的是引入数据库技术后的整个计算机系统，一般由计算机硬件系统、软件系统、数据集合和用户四个部分组成。在软件系统中包括操作系统、数据库管理系统以及用数据库管理系统和程序设计语言开发的数据库应用系统。在用户中包

括数据库管理员、专业的应用系统开发人员和数据库最终用户。数据库系统的组成结构如图1-4所示。

与人工管理和文件管理系统相比,数据库系统的主要特点如下:

1. 数据的结构化

在数据库系统中,数据不再针对某个应用,而是面向整个系统按照某种数据模型统一组织。数据库系统不仅可以表示事物内部各数据项之间的联系,而且可以表示事物与事物之间的联系,从而反映出现实世界事物之间的联系,实现了整体数据的结构化。数据的结构化是数据库系统与文件系统的本质区别。

2. 数据的共享性

数据库系统从整体的角度对数据进行组织,因此可以提供给多个用户、多个应用程序共享使用。数据的共享使用可以大大减少数据冗余,节约存储空间,并提高数据的一致性、完整性和安全性。

3. 数据的独立性

在数据库系统中,对数据的定义和描述已经从应用程序中分离出来。应用程序与存储在数据库中的数据相互独立。数据在数据库中的物理存储位置、结构、存取方式等均由数据库管理系统负责,而与应用程序无关。应用程序要处理的只是数据的逻辑结构。

数据与程序的彼此独立,既方便了数据库的维护,也简化了应用程序的设计,大大减小了对它们进行维护和修改的难度。

4. 统一的数据管理控制功能

数据库中的数据可以被多个用户或应用程序共享,其数据的存取往往是并发的,即多个用户可以同时使用同一个数据库中的数据。为此,数据库管理系统必须提供必要的数据管理控制功能和安全保护措施。这些功能主要有数据的并发访问控制功能、数据的安全性保护功能、数据的完整性检查功能和出现故障时数据库的恢复功能。

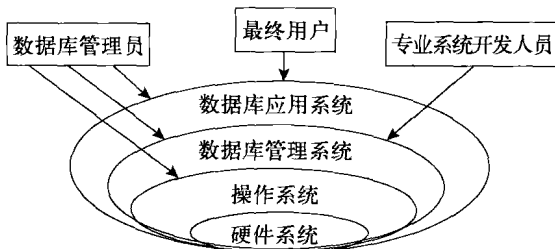


图1-4 数据库系统组成结构示意图

1.2 关系型数据库

1.2.1 数据模型概述

在现实世界中存在着各种各样的事物。每个事物都有自己的特性,事物与事物之间也存在着各种各样的联系。客观存在的事物特性及其联系反映到人的头脑,经过认识、抽象、归纳和总结,可以转化成某种形式的对事物特性及其联系的表达与描述,构建成我们的信息世界。

1.2.1.1 信息世界涉及到的主要概念

1. 实体

现实世界中客观存在并可互相区分的事物称为实体(Entity)。实体可以是实际的事物,如某个人物、某件商品、某个单位等;也可以是抽象的事件、行为,如体育比赛、文艺表演、会议、购物等。实体既可以指事物本身,也可以指事物与事物之间的联系。

2. 实体的属性

实体所具有的特性称为属性(Attribute)。一个实体可以用若干个属性来描述并和其他的实体相区别。例如,某个单位实体可以用单位名称、地址、行业、隶属关系、法人代表等属

性来描述；而某个教师实体可以用教师编号、姓名、性别、年龄、职称、部门、工作简历、照片等属性来表征。

3. 实体集

性质相同的同类型实体的集合称为实体集 (Entity Set)。例如，在教师实体集中，一组具体的教师属性表征一个具体的教师，而所有的教师都可以用相同的教师实体属性来表示。

4. 型与值

实体和属性都有型 (Type) 和值 (Value) 之分。所谓“型”指的是对某一类数据的结构和属性的说明，而“值”指的是在“型”约束下的一个具体数据。例如，一个只有表头的空表称为实体型，又叫框架。如表1-1所示即为一个用“教师编号”、“姓名”、“性别”、“出生日期”、“学历”、“职称”、“部门”等属性构成的教师实体型。

表1-1 教师实体型

教师编号	姓名	性别	出生日期	学历	职称	部门

在实体型下，按照表中的栏目要求填入的具体数据就是实体值。例如，表1-2中所示的“199806”、“茅丽燕”、“女”、“08/21/1959”、“研究生”、“教授”、“经济系”就是该教师实体型的一个实体值。

表1-2 教师实体型的实体值

教师编号	姓 名	性别	出生日期	学 历	职 称	部 门
199806	茅丽燕	女	08/21/1959	研究生	教 授	经济系
199006	张玉萍	女	03/21/1966	大 学	副教授	会计系

属性也同样具有型和值之分。如在表1-2中，“教师编号”、“姓名”、“性别”、“出生日期”、“学历”、“职称”、“部门”等栏目是属性的型，在栏目下所填的“199806”、“茅丽燕”、“女”、“08/21/1959”、“研究生”、“教授”、“经济系”等具体数据就称为对应属性的值。

5. 实体标识符

能够唯一、最小标识实体的属性的集合称为实体标识符，也称关键字 (Key)。如“教师编号”能唯一地标识一个教师实体，可以作为实体标识符。

6. 实体间的联系

在现实世界中，事物内部和事物之间都存在着联系 (Relationship)。这种联系也必然反映在信息世界中，构成实体内部和实体之间的联系。实体内部的联系指的是组成实体的各个属性之间的联系。实体之间的联系指的是不同实体集之间的联系。实体之间的联系可以分为三类：

(1) 一对一联系

如果对于实体集A中的每一个实体，实体集B中至多只有一个实体与其有联系，反之亦然，则称实体集A与实体集B之间具有一对一的联系，记为1:1。

例如，在一个单位里，职工与工资之间就是一对一的联系。

(2) 一对多联系

如果对于实体集A中的每一个实体，实体集B中有n个实体 ($n \geq 0$) 与其有联系，反之，对于实体集B中的每一个实体，实体集A中至多只有一个实体与之联系，则称实体集A与实体集B之间具有一对多的联系，记为1:n。

例如，在一个单位里，部门与职工之间具有一对多的联系。

与一对多的联系正好相反的是多对一的联系。例如，职工与部门之间的联系即为多对一的联系。一对多联系和多对一联系之间的差别取决于以谁为主。

(3) 多对多联系

如果对于实体集A中的每一个实体，实体集B中有 n 个实体 ($n \geq 0$) 与其有联系，反之，对于实体集B中的每一个实体，实体集A中也有 m 个实体 ($m \geq 0$) 与之有联系，则称实体集A与实体集B之间具有多对多的联系，记为 $m:n$ 。

例如，在学校里，学生与课程之间具有多对多的联系。

在三种联系中，一对多的联系是一种最普遍、最基本的联系。实际上，一对一的联系可以看作一对多联系的一个特例，而一对多的联系又可以看作多对多联系的一个特例。

1.2.1.2 数据模型简介

经过抽象得到的、概念化的对事物特性以及事物之间相互联系的表达与描述的集合称为数据模型。数据模型确定了数据库中数据的组织结构框架，表示出了数据之间的联系。数据模型通常由数据结构、数据操作和数据的完整性约束条件三个部分组成。

1. 数据结构

数据结构是对系统静态特性的描述，是所研究的对象类型的集合。这些对象通常分为两类，一类是与数据类型、内容、性质有关的对象，一类是与数据之间的联系有关的对象。

数据结构是刻画一个数据模型性质最重要的方面，因此在数据库系统中，通常都按照数据结构的类型来命名数据模型。如将层次结构、网状结构和关系结构的数据模型分别命名为层次模型、网状模型和关系模型。

2. 数据操作

数据操作是对系统动态特性的描述，是指对数据库中的各种对象允许执行的操作的集合，包括操作以及有关的操作规则。数据库中的数据操作主要有检索和更新（包括存储、插入、删除、修改等）两大类。数据模型必须定义这些操作的确切含义、操作符号、操作规则（如优先级）以及实现操作的语言。

3. 数据的完整性约束条件

数据的完整性约束条件是一组完整性规则的集合。完整性规则是给定的数据模型中数据及其联系所具有的制约和依存规则，用于限定符合数据模型的数据库状态以及状态的变化，以保证数据的正确、有效和相容。

任何一种数据库管理系统都是基于某种数据模型在计算机上实现的。采用的数据模型不同，建立的DBMS也就不同。在数据库系统中，主要的数据模型有三种：层次模型、网状模型和关系模型。

1.2.1.3 常用数据模型

1. 层次模型

层次模型是在数据库系统中最早应用的数据模型。层次模型用树形结构来表示实体以及实体之间的联系。层次模型具有以下特点：

- (1) 有且只有一个结点，且这个结点没有双亲，该结点称为根结点；
- (2) 除根以外的其他结点有且仅有一个双亲。

在层次模型中，每个结点表示一个记录（实体）类型，记录（实体）之间的联系是一对多的联系。因此，层次模型只能反映实体间的一对多联系。现实世界中有许多实体之间的联系就是一种层次关系。如图1-5所示一个单位的组织机构就是一个层次模型。

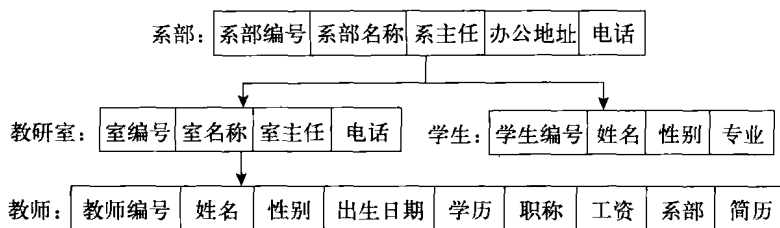


图1-5 层次模型示意图

层次结构具有层次分明、结构清晰的优点，其缺点是只能反映事物间的一对多联系。

2. 网状模型

网状模型用网状结构表示实体以及实体之间的联系。网状模型具有以下特点：

- (1) 允许一个以上的结点无双亲；
- (2) 一个结点可以有一个以上的双亲。

网状模型是一种比层次模型更具普遍性的结构，它突破了层次模型的两点限制，既允许多个结点无双亲，又允许一个结点有多个双亲结点。

与层次模型一样，网状模型中的每个结点也是表示一个记录（实体）类型，但记录（实体）之间的联系既可以是一对多的联系，也可以是多对多的联系。如图1-6所示的厂商与零部件的联系就是一个网状模型。

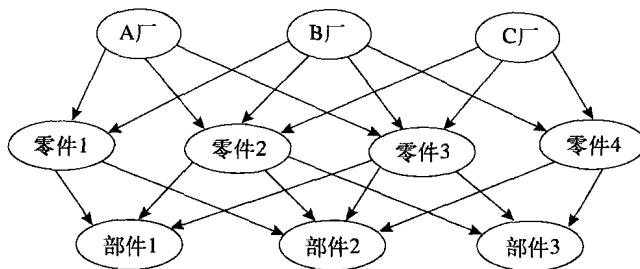


图1-6 网状模型示意图

网状模型表达能力强，可以更直接地描述现实世界，方便地表示实体之间多对多的联系，但其结构复杂，实现与使用均较困难。

3. 关系模型

关系模型用二维表格结构来描述实体和实体之间的联系。关系模型以严格的关系数学理论为基础，数据结构简单，模型概念清楚，格式描述统一，能直接反映实体之间一对一、一对多和多对多的联系，操作对象和结果均为二维表结构，易学习、易理解，且符合我们一般的使用习惯。

1.2.2 关系型数据库

关系型数据库系统采用关系模型作为数据的组织方式。20世纪80年代以来所推出的数据库管理系统几乎都支持关系模型，其已成为目前应用最为广泛的数据库系统。

1.2.2.1 关系术语与关系数据库

1. 关系

一个关系（Relation）对应一张二维表，每个关系有一个关系名。

在Visual FoxPro中，一个关系就是一个表（或称数据表文件，扩展名为.dbf）。如表1-3所