



高等学校应用型系列规划教材

GAODENGXUEXIAO
YINGYONGXING
XILIEGUIHUAJIAOCAI

Visual FoxPro 程序设计与应用

Visual FoxPro CHENGXU SHEJI YU YINGYONG

陈维默 王榕国 编著



北京理工大学出版社

BEIJING INSTITUTE OF TECHNOLOGY PRESS

Visual FoxPro 程序设计与应用

陈维默 王榕国 编著

 **北京理工大学出版社**
BEIJING INSTITUTE OF TECHNOLOGY PRESS

内 容 简 介

本书由数据库技术基础、语言基础、程序设计、系统应用四部分内容组成,全面系统地介绍了利用中文 Visual FoxPro 6.0 进行程序设计的方法与工具。全书共分 12 章,主要内容包括:数据库技术基础、数据库程序设计中的数据类型、函数与表达式、数据库与表的基本操作、结构化查询语言 SQL、查询与视图创建、程序设计基础知识、面向对象程序设计、表单及控件的设计、报表设计、菜单设计、应用软件开发过程及综合实例。

本书除了作为计算机等级(二级)考试的教材之外,还可以作为高等院校及其他各类计算机培训班的微机数据库课程教学用书。对于计算机应用人员和计算机爱好者,本教程也是一本实用的自学参考书。

版权专有 侵权必究

图书在版编目(CIP)数据

Visual FoxPro 程序设计与应用/陈维默,王榕国编著. —北京:北京理工大学出版社, 2006.8
ISBN 7-5640-0679-X

I.V… II.①陈… ②王… III.关系数据库—数据库管理系统, Visual FoxPro—程序设计 IV.TP311.138

中国版本图书馆 CIP 数据核字(2006)第 069203 号

出版发行 / 北京理工大学出版社

社 址 / 北京市海淀区中关村南大街 5 号

邮 编 / 100081

电 话 / (010)68914775(总编室) 68944990(批销中心) 68911084(读者服务部)

网 址 / <http://www.bitpress.com.cn>

经 销 / 全国各地新华书店

印 刷 / 北京市业和印务有限公司

开 本 / 787 毫米×1092 毫米 1/16

印 张 / 18.5

字 数 / 381 千字

版 次 / 2006 年 8 月第 1 版 2006 年 8 月第 1 次印刷

责任校对 / 郑兴玉

定 价 / 26.00 元

责任印制 / 母长新

图书出现印装质量问题,本社负责调换

前 言

由于现代科学技术特别是电子计算机技术的迅猛发展,数据库技术在各个领域中的应用日趋普遍。为适应 21 世纪培养人才的需要,作者结合多年的教学实践,在为各类本科生讲授的《Visual FoxPro 6.0 程序设计与应用》讲义的基础上,针对数据库技术在各领域应用所需的基本理论知识和操作技能,编写了这本教材。本书力图传授知识与培养能力并重,具有如下 3 个特点:

1. 注重实用性兼顾理论性。从工程实用性出发,本书所举的例子均精选于中文 Visual FoxPro 6.0 技术应用于工程领域中的实际例子。

2. 力求深入浅出。对于一些较为抽象、难以理解的概念、理论,一方面从内容的编排上尽量循序渐进地深入;另一方面尽可能地给出具体的实例予以解释。

3. 注重能力的培养。对于大学生来说,学习《Visual FoxPro 6.0 程序设计与应用》这门课程,重点在于如何应用数据库技术解决生产实践中的问题。因此,掌握中文 Visual FoxPro 6.0 的系统功能,开发一个简单的工程领域中的数据库管理系统,就成为本书的主线。这可使学习者获得较为完整的开发数据库管理系统的技术技能的训练,从而明显地提高学习者应用计算机解决问题的能力,达到掌握知识与能力培养并重的目的。

本书共分 12 章:第 1 章“数据库技术基础”、第 2 章“Visual FoxPro 概述”、第 3 章“数据、函数与表达式”、第 4 章“表与数据库的基本操作”、第 5 章“结构化查询语言 SQL”、第 6 章“查询与视图”、第 7 章“程序设计基础”、第 8 章“面向对象程序设计”、第 9 章“表单控件”、第 10 章“报表设计”、第 11 章“菜单设计”、第 12 章“应用软件开发过程及一个综合实例”。本书体现了计算机等级(二级 Visual FoxPro)考试大纲的要求,力求做到内容既不超纲,又不降低水平;内容的编排简明扼要,层次分明;理论联系实际,面向应用。在每一章的后面均附有练习题,供读者复习参考。

限于编者对数据库原理和技术及中文 Visual FoxPro 6.0 的系统功能的把握水平和经验,书中难免有不妥和疏漏之处,恳请读者批评指正。

编者

2006 年 3 月

目 录

第 1 章 数据库技术基础1	2.5.4 项目管理器的使用25
1.1 数据管理技术的发展.....1	2.6 VFP 的设计器、生成器与向导工具 ...27
1.1.1 数据与数据处理.....1	2.6.1 设计器27
1.1.2 计算机数据管理的发展.....1	2.6.2 向导28
1.2 数据模型.....3	2.6.3 生成器29
1.2.1 三个世界的含义.....4	2.7 习题.....30
1.2.2 数据模型的有关概念.....5	第 3 章 数据、函数与表达式32
1.2.3 层次数据模型.....6	3.1 常量与变量.....32
1.2.4 网状数据模型.....6	3.1.1 常量32
1.2.5 关系数据模型.....7	3.1.2 变量34
1.3 数据库系统的组成与模式结构.....9	3.1.3 内存变量常用命令36
1.3.1 数据库系统.....9	3.2 常用函数.....40
1.3.2 数据库系统的三级模式结构...10	3.2.1 数值函数40
1.3.3 数据库系统研究的范围.....11	3.2.2 字符函数42
1.4 习题.....12	3.2.3 日期和时间函数46
第 2 章 Visual FoxPro 概述14	3.2.4 数据类型转换函数47
2.1 VFP 的功能与特点14	3.2.5 测试函数50
2.1.1 VFP 概述14	3.2.6 数据库函数52
2.1.2 Visual FoxPro 6.0 的特点14	3.3 表达式.....55
2.2 VFP 的启动与退出15	3.3.1 数值、字符与日期时间 表达式55
2.2.1 启动.....15	3.3.2 关系表达式57
2.2.2 退出.....16	3.3.3 逻辑表达式59
2.3 VFP 的应用开发环境16	3.4 习题.....60
2.3.1 主界面.....16	第 4 章 表与数据库的基本操作63
2.3.2 菜单操作.....16	4.1 表的创建.....63
2.3.3 工具栏的使用.....17	4.1.1 表的设计过程63
2.3.4 命令操作.....19	4.1.2 表结构的创建63
2.3.5 VFP 的环境设置20	4.1.3 表记录的输入66
2.4 VFP 的文件类型22	4.1.4 修改表结构66
2.5 项目管理器.....22	4.2 数据库的基本操作.....67
2.5.1 项目管理器的作用.....22	4.2.1 基本概念67
2.5.2 项目文件的创建.....23	4.2.2 建立数据库67
2.5.3 项目管理器的组成.....23	

4.2.3 打开数据库.....	68	5.1.8 利用 NULL 值查询.....	99
4.2.4 修改数据库.....	68	5.1.9 别名与自联接查询.....	99
4.2.5 删除数据库.....	71	5.1.10 超联接查询.....	101
4.3 表的基本操作及操作命令.....	71	5.1.11 VFP 专用的 SQL SELECT 的特殊选项.....	103
4.3.1 VFP 命令的一般结构.....	72	5.2 SQL 操作功能.....	104
4.3.2 创建表文件结构.....	73	5.2.1 数据的插入.....	105
4.3.3 表的打开或关闭.....	73	5.2.2 数据的更新.....	106
4.3.4 使用浏览器操作表.....	73	5.2.3 数据的删除.....	106
4.3.5 增加记录的命令.....	74	5.3 SQL 定义功能.....	107
4.3.6 删除记录.....	75	5.4 习题.....	109
4.3.7 修改记录.....	75		
4.3.8 显示记录.....	76	第 6 章 查询与视图	114
4.3.9 查询定位.....	76		
4.4 表的索引.....	77	6.1 查询与统计命令.....	114
4.4.1 基本概念.....	77	6.1.1 数据筛选命令.....	114
4.4.2 在表设计器中建立索引.....	78	6.1.2 查询定位命令.....	115
4.4.3 用命令建立索引.....	79	6.1.3 统计命令.....	116
4.4.4 使用索引.....	80	6.1.4 多表操作命令.....	117
4.4.5 筛选记录.....	81	6.2 使用查询设计器建立查询.....	119
4.4.6 查看索引结果.....	81	6.2.1 查询设计器.....	119
4.5 数据完整性.....	82	6.2.2 建立查询.....	120
4.5.1 实体完整性与主关键字.....	82	6.2.3 运行查询.....	123
4.5.2 域完整性与约束规则.....	82	6.2.4 用“查询向导”建立查询.....	125
4.5.3 参照完整性与表之间 的关联.....	83	6.3 视图.....	127
4.6 多表操作.....	85	6.3.1 视图的概念.....	127
4.6.1 工作区的概念.....	85	6.3.2 使用视图设计器建立视图.....	128
4.6.2 使用不同工作区的表.....	86	6.3.3 使用 SQL 命令建立视图.....	129
4.6.3 表之间的关联.....	87	6.3.4 视图与数据更新.....	131
		6.3.5 使用视图.....	132
第 5 章 结构化查询语言 SQL	89	6.3.6 用“视图向导”建立视图.....	133
5.1 SQL 数据查询.....	90	6.4 习题.....	136
5.1.1 简单查询.....	91	第 7 章 程序设计基础	138
5.1.2 简单的联接查询.....	93		
5.1.3 嵌套查询.....	93	7.1 程序文件的建立与运行.....	138
5.1.4 几个特殊运算符.....	95	7.1.1 程序文件的概念.....	139
5.1.5 排序.....	96	7.1.2 程序文件的建立与编辑.....	139
5.1.6 简单的计算查询.....	96	7.2 VFP 常用的命令.....	141
5.1.7 分组与计算查询.....	98	7.2.1 简单的输入输出命令.....	141
		7.2.2 环境设置语句.....	144

7.3 程序的基本结构.....	144	8.6 表单向导.....	194
7.3.1 选择结构.....	144	8.6.1 打开表单向导	194
7.3.2 循环结构.....	148	8.6.2 通过与向导的对话 创建表单	194
7.3.3 编程实例.....	152	8.7 习题.....	198
7.4 过程.....	156	第9章 表单控件.....	202
7.4.1 过程的定义.....	156	9.1 输出类控件.....	202
7.4.2 过程文件的定义.....	157	9.1.1 标签(Label).....	202
7.4.3 过程的调用.....	158	9.1.2 图像、线条、形状	203
7.5 参数传递与变量的作用域.....	159	9.2 输入类控件.....	205
7.5.1 参数传递.....	159	9.2.1 文本框(TextBox).....	205
7.5.2 变量的作用域及其定义.....	161	9.2.2 编辑框(Editbox)	211
7.6 自定义函数.....	164	9.2.3 列表框(Listbox)与 组合框(Combox).....	213
7.6.1 自定义函数的定义.....	164	9.2.4 微调控件(Spinner)	220
7.6.2 自定义函数的调用.....	165	9.3 控制类控件.....	221
7.6.3 自定义函数的参数传递.....	165	9.3.1 命令按钮(Commandbutton)....	221
7.7 习题.....	166	9.3.2 命令按钮组(Commandgroup). 222	
第8章 面向对象程序设计.....	173	9.3.3 复选框(Checkbox).....	225
8.1 面向对象程序设计的基本要素.....	173	9.3.4 选项按钮组(OptionGroup)	226
8.1.1 对象.....	173	9.3.5 计时器控件(Timer)	227
8.1.2 属性.....	174	9.4 多重容器类.....	228
8.1.3 方法.....	174	9.4.1 容器(Container).....	229
8.1.4 事件.....	174	9.4.2 表格(Gird)	231
8.1.5 事件驱动的工作方式.....	174	9.4.3 页框(Pageframe)与 页面(Page).....	236
8.2 表单设计器.....	175	9.4.4 表单集(Formset).....	240
8.2.1 表单设计器的功能与特点.....	175	9.5 连接类控件.....	241
8.2.2 打开表单设计器.....	175	9.5.1 ActiveX 绑定控件 (Oleboundcontrol).....	241
8.2.3 表单设计工具.....	176	9.5.2 超级链接(Hyperlink).....	243
8.3 使用表单设计器创建表单.....	177	9.6 习题.....	244
8.4 VFP 的事件代码编写	184	第10章 报表设计	250
8.4.1 在容器分层结构中 引用对象.....	185	10.1 使用报表设计器创建报表.....	250
8.4.2 编程设置属性和调用 方法程序.....	186	10.1.1 报表的带区.....	250
8.4.3 带参事件编写.....	188	10.1.2 在报表中使用控件	253
8.4.4 VFP 的事件序列	189	10.1.3 报表输出和保存	257
8.5 用户定义属性和方法程序.....	190		
8.5.1 用户定义属性.....	190		
8.5.2 用户定义方法程序.....	192		

10.1.4 报表设计示例.....	258	11.2.8 生成菜单程序.....	272
10.2 使用报表向导创建报表.....	260	11.2.9 运行菜单.....	272
10.2.1 启动报表向导.....	261	11.3 快捷菜单设计.....	272
10.2.2 利用向导创建 一对一报表.....	261	11.3.1 建立快捷菜单的 方法和过程.....	273
10.2.3 利用向导创建 一对多报表.....	262		
10.3 快速报表.....	262		
第 11 章 菜单设计	265	第 12 章 应用软件开发过程及 一个综合实例	275
11.1 菜单设计器的使用与操作.....	265	12.1 开发 VFP 数据库应用系统的 一般步骤.....	275
11.1.1 打开“菜单设计器”.....	265	12.1.1 软件需求分析.....	275
11.1.2 “菜单设计器”简介.....	266	12.1.2 系统设计.....	276
11.2 创建菜单系统.....	267	12.1.3 编码与测试.....	277
11.2.1 规划菜单系统.....	268	12.1.4 运行和维护.....	279
11.2.2 创建菜单.....	268	12.1.5 应用程序的基本 性能要求.....	279
11.2.3 创建子菜单.....	269	12.2 综合应用实例.....	280
11.2.4 创建过程.....	270	12.2.1 软件需求分析.....	280
11.2.5 为菜单或菜单项 指定任务.....	271	12.2.2 系统设计.....	282
11.2.6 设定快捷方式.....	271	12.2.3 运行与维护.....	286
11.2.7 预览菜单设计结果.....	271		

第 1 章 数据库技术基础

随着计算机技术的蓬勃发展, 计算机应用从科学计算、过程控制进入数据处理, 使计算机从少数科学家手中的专业工具成为人们日常工作中处理数据的得力助手和亲密伙伴。当今世界, 在计算机的三大主要应用领域(科学计算、过程控制和数据处理)中, 数据处理迅速上升为计算机应用的主要方面。在本章中, 我们将从数据处理开始介绍数据处理的发展、数据模型和数据库以及所涉及的基本概念, 作为后面各章学习的基础。

1.1 数据管理技术的发展

数据库管理系统是处理数据的有效工具, 在本节中首先需要了解数据、数据处理的基本概念和计算机数据管理的发展历程。

1.1.1 数据与数据处理

1. 数据

数据是指一切计算机可以接受并能处理的符号序列。数据的概念包括三个方面:

- (1) 描述事物特性的数据内容。
- (2) 存储在某一种媒体上的数据形式。
- (3) 不仅包括数字、字母、文字和其他特殊字符组成的文本形式的数据, 而且还包括图形、图像、动画、影像、声音等多媒体数据。

2. 数据处理

数据处理是指将数据转换成信息的过程, 对数据的组织、分类、编码、存储、检索和维护提供操作手段。通过处理数据可以获得信息, 通过分析和筛选信息可以产生决策。

1.1.2 计算机数据管理的发展

计算机数据管理技术的发展经历了人工管理、文件系统和数据库系统等发展阶段。

1. 人工管理阶段

20 世纪 50 年代中期以前, 外存储器只有卡片、纸带、磁带, 没有像磁盘这样的可以随机访问、直接存取的外部存储设备。软件方面, 没有专门管理数据的软件, 数据由计算或处理它的程序自行携带。数据管理任务, 包括存储结构、存取方法、输入输出方式等完全由程序设计人员自负其责。其特点为:

- (1) 数据不保存。
- (2) 数据与程序不具有独立性。
- (3) 数据冗余大。

在人工管理阶段, 一组数据对应一组程序, 数据不长期保存, 程序运行结束后就退出计算机系统, 一个程序中的数据无法被其他程序利用, 因此程序与程序之间存在大量的重复数据, 这种情况称为数据冗余。

2. 文件系统阶段

20 世纪 50 年代后期至 60 年代中后期, 计算机开始大量地用于管理中的数据处理工作。大量的数据存储、检索和维护成为紧迫的需求。可直接存取的磁盘成为联机的主要外存。在软件方面, 出现了高级语言和操作系统。操作系统中的文件系统是专门管理外存储器的数据管理软件。其特点为:

- (1) 数据以文件的形式长期保留在外存中, 可以被多次存取。
- (2) 文件的物理结构和逻辑结构间有简单的变换。
- (3) 文件组织已多样化, 如对文件进行索引、排序等。
- (4) 数据基本上是面向应用, 文件系统中的数据文件是为了满足特定业务领域或某部门的专门需要而设计的, 服务于某一特定应用程序, 数据和程序相互依赖。同一数据项可能重复出现在多个文件中, 导致数据冗余度大, 不能实现数据共享。
- (5) 文件之间彼此孤立, 不能反映现实世界中事物之间的复杂的相互关系。

3. 数据库系统阶段

20 世纪 60 年代后期开始, 需要计算机管理的数据且急剧增长, 并且对数据共享的需求日益增强; 为了实现计算机对数据的统一管理, 达到数据共享的目的, 发展了数据库技术。其特点为:

- (1) 数据结构化。
- (2) 提高了数据的共享性, 使多个用户能够同时访问数据库中的数据。
- (3) 减小了数据的冗余度, 以提高数据的一致性和完整性。
- (4) 由 DBMS 对数据库进行统一的数据管理和控制功能。

Visual FoxPro 就是一种在微机上运行的数据库管理系统软件。在数据库管理系统的支持下, 数据与程序之间的关系如图 1.1 所示。

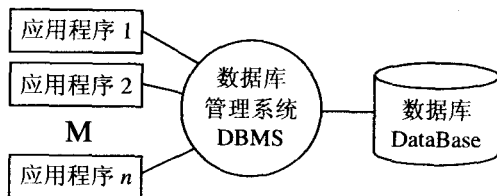


图 1.1 数据库系统中应用程序与数据库之间的关系

4. 分布式数据库系统

分布式数据库系统是数据库技术和计算机网络技术紧密结合的产物。20 世纪 70 年代

后期之前,数据库系统多数是集中式的。网络技术的进展为数据库提供了分布式运行环境,从主机—终端系统结构发展到客户/服务器系统结构。具体分为下列3种结构形式。

(1) 物理上分布、逻辑上集中的分布式数据库结构

一个逻辑上统一、地域上分布的数据集合,是计算机网络环境中各个节点局部数据库的逻辑集合,同时受分布式数据库管理系统的统一控制和管理,即把全局数据模式按数据来源和用途,合理分布在系统的多个节点上,使大部分数据可以就地或就近存取,而用户感觉不到数据的分布。

(2) 物理上分布、逻辑上分布的分布式数据库结构

把多个集中式数据库系统通过网络连接起来,各个节点上的计算机可以利用网络通信功能访问其他节点上的数据库资源。它一般由两部分组成:一是本地节点的数据,二是本地节点共享的其他节点上的有关数据。在这种运行环境中,各个数据库系统的数据库由各自独立的数据库管理系统集中管理,节点间的数据共享由双方协商确定。这种数据库结构有利于数据库的集成、扩展和重新配置。

(3) 开放式数据库连接(ODBC, Open DataBase Connectivity)

用于数据库服务器的一种标准协议。可以安装多种数据库的 ODBC 驱动程序,从而使 Visual Foxpro 能够与该数据库相连,访问其中的数据。

5. 面向对象数据库系统

面向对象方法是一种认识、描述事物的方法论,它起源于程序设计语言。面向对象程序设计是 20 世纪 80 年代引入计算机科学领域的一种新的程序设计技术和范型,它的发展十分迅猛,影响涉及计算机科学及其应用的各个领域。

通俗地讲,面向对象的方法就是按照人们认识世界和改造世界的习惯方法对现实世界的客观事物、对象进行最自然的、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)结合成为一个有机整体,事物完整的内部结构和外部行为机制被反映得淋漓尽致。

面向对象数据库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中的实现和应用,它既是一个面向对象的系统,又是一个数据库系统。Visual FoxPro 不但仍然支持标准的过程化程序设计,而且在语言上还进行了扩展,提供了面向对象程序设计的强大功能和更大的灵活性。

1.2 数据模型

根据某种数据模型,人们可以用数据世界来合理表示现实世界的某一部分。因此,数据模型是数据库研究的基础,这也正是数据模型的重要意义。

1.2.1 三个世界的含义

获得一个数据库管理系统所支持的数据模型的过程, 是一个从现实世界的事物出发, 经过抽象, 以获得人们所需要的概念模型和数据模型的过程。信息在这一过程中经历了三个不同的世界: 现实世界、信息世界和数据(机器)世界。

1. 现实世界

人们管理的对象存在于现实世界之中。在现实世界里, 事物及事物之间存在着联系, 这种联系是客观存在的。

2. 信息(概念)世界

概念世界是现实世界在人们头脑中的反映, 是对客观事物及其联系的一种抽象描述。它不是现实世界的简单录像, 而要经过选择、命名、分类等抽象过程产生概念模型。概念模型是现实世界到机器世界必然经过的中间层次。信息世界涉及的概念主要有:

(1) 实体(Entity): 客观存在并可相互区别的事物称为实体。实体可以是实际事物, 也可以是抽象事件。同一类实体的集合称为实体集。例如, 全体职工的集合、全馆图书等。我们用命名的实体型表示抽象的实体集, 例如, 实体型“职工”表示全体职工的概念, 并不具体指职工甲或职工乙。以后在不致引起混淆的情况下, 实体就是指实体型而言。

(2) 属性(Attribute): 描述实体的特性称为属性。属性的具体取值称为属性值, 用以刻画一个具体实体。

(3) 关键字(KeyWord): 如果某个属性或属性组合的值能够唯一地标识出实体集中的每一个实体, 则可以选作关键字。用作标识的关键字也称为码。

(4) 联系(Relationship): 实体集之间的对应关系称为联系, 它反映现实世界事物之间的相互关联。联系分为两种, 一种是实体内部各属性之间的联系。例如, 相同职称的有很多人, 但一个职工当前只有一种职称; 另一种是实体之间的联系。例如, 一位读者可以借阅若干本图书, 同一本书可以相继被多个读者借阅。在现实世界中, 实体之间的联系可分为三种类型: “一对一”的联系, “一对多”的联系, “多对多”的联系, 如图 1.2 所示。

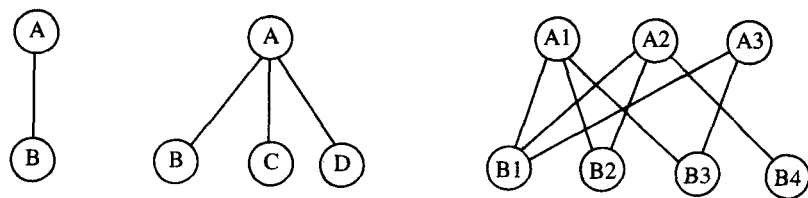


图 1.2 实体间的联系

① 一对一联系(1:1): 设 A, B 为两个实体集。若 A 中的每个实体至多和 B 中的一个实体有联系, 反过来, B 中的每个实体至多和 A 中的一个实体有联系, 则称 A 对 B 或 B 对 A 是 1:1 联系。

② 一对多联系(1:n): 如果 A 中的每个实体可以和 B 中的几个实体有联系, 而 B 中的每个实体至多和 A 中的一个实体有联系, 那么 A 对 B 属于 1:n 联系。

③ 多对多联系($m:n$): 若 A 中的每个实体可以和 B 中的多个实体有联系, 反过来, B 中的每个实体也可以与 A 中的多个实体有联系, 则称 A 对 B 或 B 对 A, 是 $m:n$ 联系。

3. 数据 (机器) 世界

数据库中的数据是有结构的, 这种结构用数据模型表示。数据模型将概念世界中的实体及实体间的联系进一步抽象成便于计算机处理的方式。任何一个数据库管理系统都是基于某种数据模型的。有关术语在三个世界中的变化见表 1.1。

表 1.1 有关术语在三个世界中的对照表

现实世界	信息 (概念) 世界	数据 (机器) 世界
组织	实体及其联系	数据库
事物类	实体集	文件
事物	实体	记录
特征	属性	数据项

1.2.2 数据模型的有关概念

广义地讲, 模型是对客观世界中复杂对象的抽象描述, 获取模型的抽象过程叫做建模 (Modeling)。而数据模型(Data Model)是数据库系统中用于提供信息表示和操作手段的形式构架。可分为:

概念模型: 概念模型是在信息世界中形成的, 它是按用户的观点对数据进行建模, 主要用于表示数据的逻辑特性, 如实体、属性和联系。概念模型只是在概念上表示在数据世界(数据库)中将要存储一些什么信息, 而不管这些信息是怎样实现存储的。最常见的概念模型是实体—联系(E-R)模型。

实施模型: 实施模型侧重于数据的表达方式和数据库结构的实现方法, 主要有层次模型、网状模型和关系模型。对应于这三种数据模型, 我们可以建立层次型数据库、网状型数据库与关系型数据库。

1. 数据模型的两个要素

(1) 数据结构: 用于描述系统的静态特性, 是所研究的对象类型的集合。对象一般分为两大类, 一类是与数据类型、内容、性质有关的对象; 另一类是与数据之间联系有关的对象。

(2) 数据操作: 用于描述系统的动态特性, 是对数据库中各种对象及实例(可理解成对象的值)允许执行的操作集合。数据库主要有检索和更新(包括插入、删除、修改等)两类操作。

(3) 完整性: 指数据的正确性和相容性。数据库中的数据都要遵守一组预先规定的条件, 这种条件在数据库中称为完整性约束条件。数据库管理系统必须提供一种功能来保证数据库中数据的完整性。这种功能亦称为完整性检查, 即系统用一定的机制来检查数据库中的数据是否满足规定的条件。

2. 数据模型应满足的要求

数据模型应能比较真实地模拟现实世界，容易为人们所理解，并且便于在计算机上实现。

1.2.3 层次数据模型

用树形结构表示实体及其之间联系的模型称为层次模型。在这种模型中，数据被组织成由“根”开始的“树”，每个实体由根开始沿着不同的分支放在不同的层次上。如果不再向下分支，那么此分支序列中最后的节点称为“叶”。上级节点与下级节点之间为一对多的联系，图 1.3 给出了一个层次数据模型的例子。

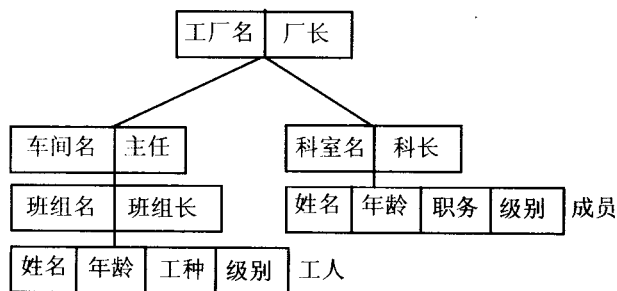


图 1.3 层次数据模型

层次数据模型的最大特点是，有且仅有一个根节点无父节点，其他节点有且仅有一个父节点。

1.2.4 网状数据模型

用网状结构表示实体及其联系的模型称为网状模型。网中的每一个节点代表一个实体类型。网状模型突破了层次模型的两点限制：允许节点有多于一个的父节点；可以有一个以上的节点没有父节点。因此，网状模型可以方便地表示各种类型的联系。但网状模型在概念、结构和使用上都比较复杂，而且对计算机的硬件环境要求较高。图 1.4 给出了一个网状数据模型的例子。

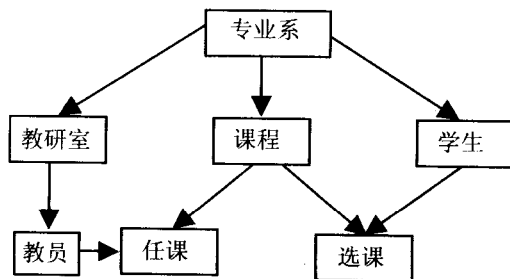


图 1.4 网状数据模型

1.2.5 关系数据模型

用二维表结构来表示实体以及实体之间联系的模型称为关系模型。关系数据模型是以关系数学理论为基础的,在关系模型中,操作的对象和结果都是二维表,这种二维表就是关系。图 1.5 给出一个关系数据模型的例子。

关系模型与层次模型、网状模型的本质区别在于数据描述的一致性,模型概念单一。在关系型数据库中,每一个关系都是一个二维表,无论实体本身还是实体间的联系均用称为“关系”的二维表来表示,使得描述实体的数据本身能够自然地反映它们之间的联系。而传统的层次和网状模型数据库是使用链接指针来存储和体现联系的。

关系数据库以其完备的理论基础、简单的模型、说明性的查询语言和使用方便等优点得到最广泛的应用。下面介绍关系型数据库中的常用术语、特点和性质。

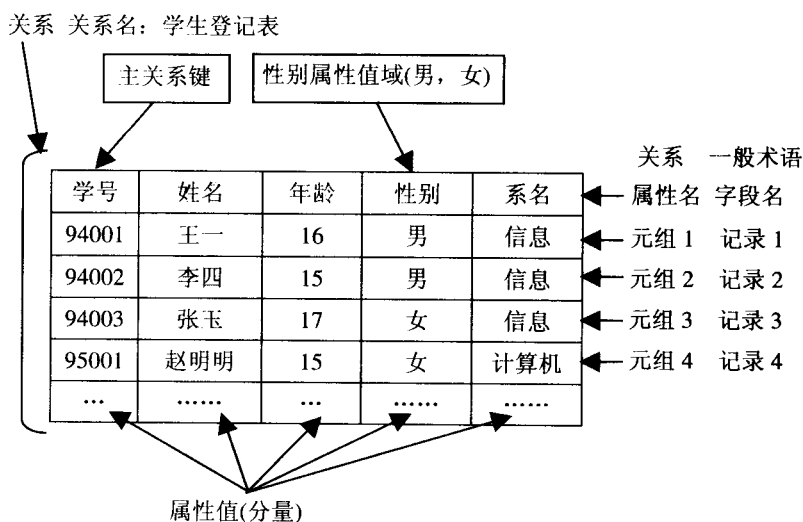


图 1.5 关系数据模型

1. 关系术语

(1) 关系: 一个关系就是一张二维表, 每个关系有一个关系名。一个关系存储为一个扩展名为 .dbf 的文件, 称为表。

对关系的描述称为关系模式, 一个关系模式对应一个关系的结构。其格式为:

关系名(属性名 1, 属性名 2, ……, 属性名 n)

在 VFP(Visual FoxPro 的简称)中表示为表结构:

表名(字段名 1, 字段名 2, ……, 字段名 n)

(2) 元组: 在一个二维表 (一个具体关系) 中, 水平方向的行称为元组, 每一行是一个元组。元组对应存储文件中的一个具体记录。

(3) 属性: 二维表中垂直方向的列称为属性, 每一列有一个属性名, 与前面讲的实体

属性相同,在 VFP 中表示为字段名。每个字段的数据类型、宽度等在创建表的结构时规定。

(4) 值域:属性的取值范围,不同元组对同一个属性的取值所限定的范围。

(5) 主码:属性或属性的组合,其值能够唯一地标识一个元组。在 VFP 中表示为字段或字段的组合,构成主码的诸属性称为该关系的主属性。

(6) 外码:如果表中的一个字段不是本表的主码,而是另外一个表的码,这个字段(属性)就称为外码,起各表之间的联系作用。

2. 关系的特点

(1) 模型概念单一,一律为表格框架,通过公共属性可建立关系之间的联系。

(2) 关系数据库由若干相互关联的表组成。

(3) 存取路径对用户透明。

(4) 关系必须规范化:关系模型中的每一个关系模式都必须满足一定的要求,最基本的是每个属性必须是不可分割的数据单元,即表中不能再包含表。

例如,手工制表中经常出现表 1.2 所示的复合表。

表 1.2 工资表

姓名	职称	应发工资			应扣款			实发工资
		基本工资	奖金	津贴	房租	水电	托儿费	

这种表格不是二维表,不能直接作为关系来存放,必须去掉表中的应发工资和应扣工资这两个表项才可以。

3. 关系的基本性质

(1) 在同一个关系中不能出现相同的属性名,即相同的字段名。

(2) 关系中不允许有完全相同的元组,即冗余。

(3) 在一个关系中元组的次序无关紧要。也就是说,任意交换两行的位置并不影响数据的实际含义。

(4) 在一个关系中列的次序无关紧要。任意交换两列的位置也不影响数据的实际含义。

(5) 关系中每一列上,属性值(数据项)必须取自同一个值域。

4. 关系运算

(1) 选择 (Selection)运算

选择运算是单个关系施加的运算,它是一种水平方向(即行的方向)上的选择,其目的是在关系 R 上,把满足条件的元组抽出来构成新的关系,这个新关系是原关系 R 上的一个子集。

(2) 投影 (Projection)运算

投影运算也是对单个关系施加的运算,它是一种垂直方向(即列的方向)上的运算。其基本思想是:从一个关系中选择所需要的属性,并重新排列组成一个新关系。因投影后属性个数要减少,故形成新的关系型,因此,应重新对这个新关系命名。

(3) 连接(Join)运算

从两个或多个关系中选取属性间满足一定条件的元组，组成一个新的关系。

1.3 数据库系统的组成与模式结构

数据库系统是一个整体的概念，它主要包括数据库、数据库管理系统和数据库应用系统。而从数据库管理系统的角度来看，数据库系统通常采用三级模式结构。本节先介绍数据库系统的组成，再介绍数据库系统三级模式结构。

1.3.1 数据库系统

数据库系统(DataBase System, 简称 DBS)是一个复杂的系统，它由硬件、软件(操作系统、数据库管理系统和编译系统等)、数据库和用户构成。

1. 数据库 (DataBase)

数据库是存储在计算机存储设备上，结构化的相关数据集合。它不仅包括描述事物的数据本身，而且还包括相关事物之间的联系。其数据结构独立于使用数据的程序，对于数据的增加、删除、修改和检索由系统软件进行统一的控制。

2. 数据库管理系统

为了让多种应用程序并发地使用数据库中具有最小冗余度的共享数据，必须使数据与程序具有较高的独立性。这就需要有一个软件系统对数据实行专门管理，提供安全性和完整性等统一控制机制，方便用户以交互命令或程序方式对数据库进行操作。

3. 数据库应用系统

数据库应用系统是系统开发人员利用数据库系统资源开发出来的，面向某一类实际应用的应用软件系统。无论是面向内部业务和管理的管理信息系统，还是面向外部、提供信息服务的开放式信息系统，从实现技术角度而言，都是以数据库为基础和核心的计算机应用系统。

4. 数据库系统的广义定义

引进数据库技术后的计算机系统，实现了有组织地、动态地存储大量相关数据，提供了数据处理和信息资源共享的便利手段。数据库系统由五部分组成：硬件系统、数据库集合、数据库管理系统及数据库应用系统、数据库管理员 DBA(专门管理维护数据库)和用户(专业用户编写处理程序；终端用户在数据库中查询信息)。各层次之间的相互关系如图 1.6 所示。