

Visual FoxPro 6.0 JI QI CHENG XU SHE JI

Visual FoxPro 6.0 及其程序设计

田爱景 卫春芳 主编



COMPUTER
COMPUTER
COMPUTER

科学出版社

Visual FoxPro 6.0 及其程序设计

田爱景 卫春芳 主编

杜利娟 朱 寄 武 青 副主编

科 学 出 版 社

2002

内 容 简 介

本书为 Visual FoxPro 课程教材,内容包括:关系型数据库系统基本概念,Visual FoxPro 6.0 基础知识,数据表、数据库的操作,项目管理器的使用,查询,视图,结构化程序设计,可视化程序设计。每章末附有习题。

Visual FoxPro 6.0 不仅完全兼容该系列较低的版本,而且也兼容 FoxBase 的应用程序,还同 Visual C++ 6.0、Visual Basic 6.0 等共同组成 Microsoft Visual Studio 98。正因为如此,本书充分考虑了 Visual FoxPro 6.0 与 FoxPro、FoxBase 教学的衔接,有利于实际教学。

本书为大专院校非计算机专业程序设计课程教材,也可供职业学校、成人教育的学生和继续教育、计算机培训班的学员以及等级考试应试者、社会学习者使用。

Visual FoxPro 6.0 及其程序设计

田爱景 卫春芳 主编

杜利娟 朱 寄 武 青 副主编

责任编辑 冯贵层

科 学 出 版 社 出 版

北京东黄城根北街16号
邮政编码:100717

湖北京山金美印刷有限责任公司印刷

新华书店北京发行所发行 各地新华书店经售

*

2002年1月第一版 开本:787×1092 1/16
2002年1月第一次印刷 印张:18
印数:1~10 000 字数:443 000

ISBN 7-03-009987-7/TP·1660

定价:22.80元

前 言

数据库技术在事务处理和办公自动化等方面有着广泛的应用,随着我国现代化进程的推进,它已经成为现代管理必须具备的知识与技能。

Visual FoxPro 6.0 是 Microsoft 公司于 1998 年推出的近几年在中国最流行的微机关系型数据库管理系统。它不仅完全兼容该系列较低的版本,而且也兼容 FoxBase 的应用程序,还同 Visual C++ 6.0、Visual Basic 6.0 等产品共同组成 Microsoft Visual Studio 98。Visual FoxPro 6.0 最大的特点是易学、高效、功能强大,特别适用于中、小型数据库的管理。多年来 Microsoft 公司将 SQL 技术、Active X 技术、OLE 技术、可视化编程技术等一系列先进技术引进到 FoxPro 系列产品之中。Visual FoxPro 6.0 的强大功能主要表现为:既能管理大量复杂的数据信息,又具有很好的安全性;使用数据库标准查询语言 SQL,使查询速度大幅度提高,能从大量数据中快速查询所需信息;具有较强的网络功能,可实现数据的远程访问和存储加工;使用面向对象的开发技术,能开发出专门针对用户需要的应用程序;能快速制作各种形式的报表、标签以及直方图、饼图等图形;具有强大的函数库,并可借助 Visual Basic 和 Visual C++ 等其他工具辅助开发,以解决某些需求不能实现的问题。

本书旨在以 Visual FoxPro 6.0 中文版为蓝本,为读者提供一本系统学习 Visual FoxPro 的教材。在编写过程中,作者力图用深入浅出、清晰简洁的叙述,比较全面地介绍 Visual FoxPro 6.0 的操作使用方法和程序设计方法,同时辅以典型的例题来帮助读者加深理解。

本书有两个明显的特点:一是充分考虑了 Visual FoxPro 6.0 与 FoxPro 乃至 FoxBase 教学的衔接,便于教学的实际操作;二是较为详细地介绍了 Visual FoxPro 6.0 的先进技术,如向导、视图、SQL 查询、可视化程序设计等有关内容,为读者使用 Visual FoxPro 6.0 的强大功能和掌握可视化面向对象的程序设计方法打下良好的基础。基于这两点,本书既可作为大专院校非计算机专业程序设计课程的教材,也可供职业学校、成人教育的学生和继续教育、计算机培训班的学员以及等级考试应试者、计算机爱好者学习使用。

全书共分八章,各章的内容安排是:第一章介绍关系型数据库系统的基本概念;第二章主要介绍 Visual FoxPro 6.0 的基础知识;第三章介绍数据表 and 数据库的操作;第四章介绍项目管理器的使用;第五章介绍查询;第六章介绍视图;第七章介绍结构化程序设计;第八章介绍可视化(Visual)程序设计。

本书由田爱景、卫春芳任主编,杜利娟、朱寄、武青任副主编。其中第一章、第五章和第六章由卫春芳编写,第二章由田爱景和武青编写,第三章由田爱景编写,第四章和第七章由朱寄编写,第八章由杜利娟编写,武青草拟部分提纲,制作部分图片,并参加了部分内容的编写。

Visual FoxPro 6.0 的命令和函数共有 700 多条,限于篇幅和学时,必然要有所取舍和侧重。至于这种取舍是否恰当,详略是否合适,诚恳希望读者不吝赐教。由于时间仓促,水平有限,书中难免会有错误和不当之处,在此敬请读者批评斧正,以便再版时修正。

编者

2001 年 10 月 20 日

JS201803

目 录

第一章 基本概念	(1)
1.1 数据库基础	(1)
1.1.1 数据、数据库、数据库管理系统和数据库系统	(1)
1.1.2 数据库系统的主要特征	(2)
1.1.3 数据模型	(3)
1.1.4 关系型数据库	(5)
1.2 Visual FoxPro 6.0 的发展史和性能特性	(7)
1.2.1 Visual FoxPro 的发展史	(7)
1.2.2 性能指标	(8)
1.3 VFP 中文版的安装	(9)
1.3.1 系统需求	(9)
1.3.2 中文版 VFP 的安装	(9)
1.3.3 从 CD-ROM 上直接安装 VFP	(9)
1.3.4 添加/删除程序	(12)
1.4 VFP 的启动和退出	(13)
1.4.1 VFP 的启动	(13)
1.4.2 VFP 的初始界面	(13)
1.4.3 VFP 的退出	(14)
1.5 VFP 的界面	(14)
1.5.1 VFP 的窗口	(14)
1.5.2 VFP 的操作方式	(15)
1.5.3 VFP 的系统主菜单	(16)
1.5.4 工具栏	(19)
1.5.5 向导	(20)
1.5.6 设计器	(21)
1.5.7 生成器	(22)
1.6 配置 Visual FoxPro 的环境	(23)
1.6.1 设置 Visual FoxPro 的配置	(23)
1.6.2 交互式地设置环境	(23)
1.6.3 修改 Windows 注册表	(24)
习题一	(25)
第二章 VFP 的基础知识	(26)
2.1 VFP 的数据类型	(26)
2.2 VFP 的常量与变量	(29)
2.2.1 常量	(29)
2.2.2 变量	(30)
2.3 表达式与运算符	(34)
2.3.1 字符型表达式和字符运算符	(34)
2.3.2 数值型表达式和货币型表达式及算术运算符	(35)

2.3.3	日期型 日期时间型表达式及其运算符	(36)
2.3.4	关系型表达式与关系运算符	(36)
2.3.5	逻辑型表达式和逻辑运算符	(37)
2.3.6	表达式和运算符小结	(38)
2.4	VFP 的常用函数	(38)
2.4.1	函数的一般格式	(39)
2.4.2	函数的类型	(39)
2.4.3	常用函数简介	(39)
2.5	VFP 的命令结构和书写规则	(45)
2.5.1	VFP 的命令结构	(45)
2.5.2	VFP 命令和短语的书写规则	(46)
2.6	VFP 的文件类型	(46)
2.7	面向对象程序设计的基本概念	(47)
2.7.1	对象(Object)	(47)
2.7.2	类(Class)	(48)
2.7.3	属性(Property)	(49)
2.7.4	事件(Event)	(49)
2.7.5	方法和方法程序	(50)
2.7.6	封装(Encapsulation)、继承性(Inheritance)、隐藏(Hide)	(51)
	习题二	(52)
第三章	数据表 and 数据库的操作	(53)
3.1	数据表与数据库	(53)
3.2	表的建立	(54)
3.2.1	表结构的设计	(54)
3.2.2	自由表的建立	(55)
3.2.3	表和编辑窗口的打开和关闭	(58)
3.2.4	通用型字段的操作	(60)
3.3	表的修改	(61)
3.3.1	表结构的修改	(61)
3.3.2	表数据记录的修改	(63)
3.4	表的创建和维护命令	(66)
3.4.1	磁盘文件操作命令	(66)
3.4.2	创建与修改表结构的命令	(68)
3.4.3	表结构与数据记录的显示	(69)
3.4.4	表结构和数据记录的复制命令	(71)
3.4.5	记录指针及其移动	(73)
3.4.6	表记录的添加与插入	(74)
3.4.7	表记录的删除和恢复	(75)
3.4.8	表数据记录的编辑命令	(77)
3.4.9	过滤器和字段表的设置	(79)
3.5	排序、索引、统计和顺序与索引查询	(80)
3.5.1	排序	(80)
3.5.2	索引	(81)

3.5.3	统计与汇总	(86)
3.5.4	顺序查询与索引查询	(89)
3.6	使用多个表的操作	(92)
3.6.1	工作区	(92)
3.6.2	表文件的关联	(95)
3.6.3	表文件的连接	(98)
3.6.4	表文件间的更新	(99)
3.7	数据库和数据库表的操作	(100)
3.7.1	数据库的设计	(100)
3.7.2	数据库的创建	(102)
3.7.3	向数据库中添加或删除表	(105)
3.7.4	数据词典及其作用	(107)
3.7.5	数据库的删除	(110)
习题三		(111)
第四章	项目管理器	(114)
4.1	创建、打开项目	(114)
4.1.1	创建新项目	(114)
4.1.2	打开已有项目	(114)
4.2	项目管理器的操作	(115)
4.2.1	查看项目中的内容	(115)
4.2.2	使用项目管理器组织文件	(117)
4.2.3	连编一个项目	(119)
4.3	定制“项目管理器”	(121)
4.3.1	改变项目管理器的外观	(121)
4.3.2	使用项目管理器的优点	(122)
习题四		(123)
第五章	查询	(124)
5.1	查询语言 SQL	(124)
5.1.1	数据定义	(124)
5.1.2	数据查询	(127)
5.1.3	数据更新	(130)
5.2	“查询向导”的使用	(131)
5.2.1	查询的设计过程	(131)
5.2.2	使用“查询向导”	(132)
5.3	“查询设计器”的使用	(133)
5.4	查询的定制	(138)
5.4.1	查询的确认	(138)
5.4.2	精确搜索	(139)
5.4.3	联接条件	(140)
5.4.4	查询结果的输出定制	(142)
5.5	查询结果	(143)
5.5.1	交叉表	(143)

5.5.2 图表.....	(146)
5.5.3 使用查询.....	(148)
习题五	(149)
第六章 视图	(150)
6.1 创建本地视图	(150)
6.1.1 使用“视图向导”.....	(150)
6.1.2 使用“视图设计器”.....	(150)
6.2 更新数据	(152)
6.2.1 向表发送更新数据.....	(152)
6.2.2 控制更新冲突的检查.....	(153)
6.3 定制视图	(154)
6.3.1 控制字段显示和数据输入.....	(154)
6.3.2 参数提示.....	(155)
6.3.3 高级选项.....	(155)
6.4 创建远程视图	(156)
6.4.1 建立数据源.....	(157)
6.4.2 连接远程数据.....	(158)
6.4.3 远程视图的创建.....	(160)
6.5 查询和更新多表.....	(162)
6.5.1 对多个表进行查询	(162)
6.5.2 在视图中使用多个表和视图.....	(162)
6.5.3 在视图中使用多个远程表.....	(164)
习题六	(165)
第七章 VFP 结构化程序设计.....	(166)
7.1 VFP 程序的建立、执行与显示	(166)
7.1.1 VFP 程序的概念	(166)
7.1.2 VFP 程序的建立与修改	(167)
7.1.3 VFP 程序的执行	(168)
7.1.4 VFP 程序的显示和打印	(168)
7.2 VFP 程序设计中常用的基本命令	(168)
7.2.1 基本输入/输出命令	(168)
7.2.2 格式输入/输出命令	(171)
7.2.3 程序注释命令和清屏命令.....	(173)
7.2.4 常用环境参数设置命令.....	(174)
7.3 VFP 程序流程的控制命令	(176)
7.3.1 程序的结构与结构化程序设计.....	(176)
7.3.2 条件判断命令与分支结构.....	(176)
7.3.3 循环控制命令与循环结构程序设计.....	(182)
7.4 VFP 中的数组及其应用	(192)
7.4.1 数组的定义与赋值.....	(192)
7.4.2 数组与表之间数据的传递.....	(194)
7.5 VFP 中过程与自定义函数的设计	(197)
7.5.1 过程与过程调用.....	(197)

7.5.2	内存变量的作用域	(201)
7.5.3	过程调用中数据的传递	(203)
7.5.4	自定义函数	(206)
7.6	程序的调试	(208)
7.6.1	程序的语法错误和逻辑错误	(208)
7.6.2	常用的程序调试方法	(209)
7.6.3	VFP 调试工具应用举例	(209)
习题七		(211)
第八章 可视化(Visual)程序设计		(216)
8.1	用表单设计应用程序界面	(216)
8.1.1	创建表单	(216)
8.1.2	表单设计器	(221)
8.1.3	表单管理	(229)
8.1.4	控件使用初步	(229)
8.1.5	表单布局设计技巧	(236)
8.1.6	表单设计示例	(238)
8.2	报表和标签设计	(242)
8.2.1	报表设计的步骤与方法	(243)
8.2.2	报表设计示例	(244)
8.2.3	报表设计器及相关菜单、工具栏	(246)
8.2.4	添加报表控件	(248)
8.2.5	报表数据分组	(252)
8.2.6	定制布局、完善报表	(253)
8.2.7	使用变量及预览、打印报表	(255)
8.2.8	标签文件的创建	(256)
8.3	设计菜单和工具栏	(257)
8.3.1	设计菜单	(258)
8.3.2	使用“菜单设计器”创建菜单系统	(258)
8.3.3	菜单设计器的使用	(260)
8.3.4	创建自定义工具栏	(264)
8.4	数据共享	(267)
8.4.1	访问数据	(268)
8.4.2	限制对数据的访问	(268)
8.4.3	使用数据工作期	(269)
8.4.4	缓冲访问数据	(269)
8.4.5	检测并解决冲突	(270)
8.5	帮助文件的设计	(270)
8.5.1	启动帮助文件	(270)
8.5.2	用 Microsoft HTML Help WorkShop 进行帮助文件设计	(270)
8.5.3	帮助文件的随发布盘发行	(271)
8.6	使用 OLE 和 Active X	(271)
8.6.1	OLE 对象操作	(272)
8.6.2	Active X 控件的应用	(273)

8.7 软件发布	(274)
8.7.1 系统软件发布.....	(274)
8.7.2 软件的安装与删除.....	(276)
习题八	(277)

第一章 基本概念

随着计算机技术的发展,计算机的主要应用已从科学计算逐渐转变为事务处理。在进行事务处理时,并不需要进行复杂的计算,主要从事大量数据的存储、查找、统计等工作。为了有效地使用保存在计算机系统的大量数据,必须采用整套严格合理的数据处理方法,即数据管理。

数据管理是指对数据的组织、存储、维护和使用等。随着计算机技术的发展,数据管理广泛采用数据库技术。目前,数据库技术已在日常生活中获得了非常广泛的应用。它不仅成为计算机科学与工程技术领域的重要研究方向之一,也日益成为计算机与工程技术人员应掌握的必不可少的知识与技能。

1.1 数据库基础

1.1.1 数据、数据库、数据库管理系统和数据库系统

数据、数据库、数据库管理系统和数据库系统是与数据库技术密切相关的四个基本概念。

1. 数据(Data)

数据是数据库存储的基本对象。在大多数人头脑中,对数据的第一个反应就是数字,其实数字只是最简单的一种数据,是数据的一种传统和狭义的理解。从广义上来理解,数据的种类很多,文字、图形、图像、声音、学生的档案记录、货物的运输情况等,这些都是数据。

可以对数据做如下定义:描述事物的符号记录称为数据。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存入计算机。

2. 数据库(DataBase,简称DB)

数据库是存放数据的仓库,只不过这个仓库是在计算机存储设备上,而且数据是按一定的格式存放的。

人们收集并抽取出一个应用所需要的大量数据之后,应将其保存起来以供进一步加工处理,进一步抽取有用信息。在科学技术飞速发展的今天,人们的视野越来越广,数据量急剧增加。过去人们把数据存放在文件柜里,现在人们借助计算机和数据库技术科学地保存和管理大量的复杂的数据,以便能方便而充分地利用宝贵的信息资源。

所谓数据库是以一定的格式存储在计算机的存储设备上的数据集合。数据库中的数据按一定的数据模型组织、描述和储存,具有较小的冗余度、较高的数据独立性和易扩展性,并可为各种用户共享。

3. 数据库管理系统(DataBase Management System,简称DBMS)

了解了数据和数据库的概念,下一个问题就是如何科学地组织和存储数据,如何高效地获取和维护数据。完成这个任务的是一个系统软件——数据库管理系统。

数据库管理系统是位于用户与操作系统之间的一层数据管理软件。数据库管理系统是为数据库的建立、使用和维护而配置的软件,它是建立在操作系统的基础上对数据库进行统一的

管理。用户使用的各种数据库命令以及应用程序的执行都要通过数据库管理系统来完成。除此之外,数据库管理系统还承担数据库的维护工作,以保证数据库的安全性和完整性。它的主要功能包括以下几个方面:

(1) 数据定义功能。DBMS 提供数据定义语言,用户通过它可以方便地对数据库中的数据对象进行定义,如建立或删除数据库、基本表和视图等。

(2) 数据操纵功能。DBMS 还提供数据操纵语言,用户可以使用它操纵数据从而实现对数据库的基本操作,如查询、插入、删除和修改等。

(3) 数据库的运行管理功能。数据库在建立、运用和维护时由数据库管理系统统一管理、统一控制,以保证数据的安全性、完整性、多用户对数据的并发使用及发生故障后的系统恢复。

(4) 数据库的建立和维护功能。它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,数据库的重组功能和性能监视、分析功能等。这些功能通常是由一些实用程序完成的。

数据库管理系统是数据库系统的一个重要组成部分。

4. 数据库系统(DataBase System, 简称 DBS)

数据库系统是指在计算机系统中引入数据库后的系统,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。应当指出的是,数据库的建立、使用和维护等工作只靠一个 DBMS 远远不够,还要有专门的人员来完成,这些人被称为数据库管理员。在一般不引起混淆的情况下,常常把数据库系统简称为数据库。数据库系统如图 1-1 所示。

数据库系统软件主要包括支持 DBMS 运行的操作系统以及 DBMS 本身、应用软件和软件开发工具等。此外,为了开发应用系统,还要有各种高级语言和编译系统,这些高级语言具有与数据库的接口。数据库系统在整个计算机系统中的地位如图 1-2 所示。

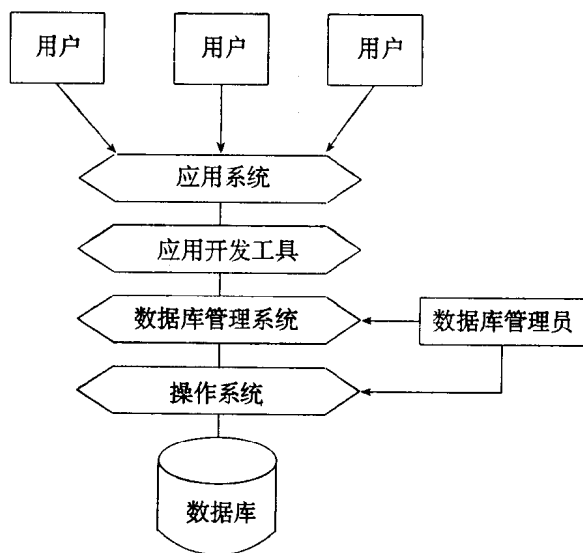


图 1-1 数据库系统

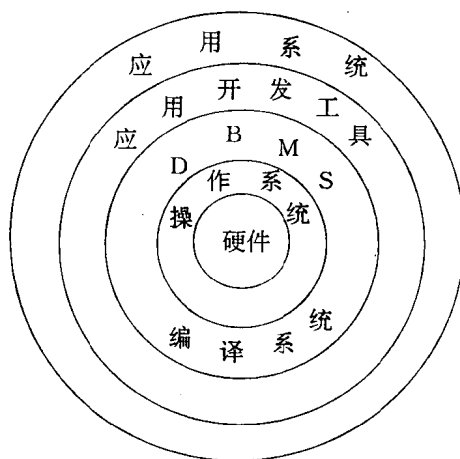


图 1-2 数据库在计算机系统中的地位

1.1.2 数据库系统的主要特征

目前,数据库系统已成为数据处理部门用作数据管理的强有力工具,其有效的管理维护能力显示了极强大的生命力,其主要特征表现如下:

(1) 较高的数据独立性。应用程序不需了解数据实际的存取方式,只通过数据库系统的存取命令就可得到需要的数据,所以,当数据的存储结构发生变化时,仅需改变数据库系统的内部程序,不需改变外部应用程序。

(2) 数据冗余度小。数据冗余度是指在数据库中的数据重复程度。数据冗余度小可以节省存储空间、减少存取时间、避免数据之间的不相容性和不一致性。但是,在实际应用中,有时为了某种原因,在一定条件下的多次存储还是有必要的。

(3) 数据共享。在数据库系统中,所有的程序都存取同一份数据库,数据完全可以达到共享。

(4) 可以避免不一致性。在数据库系统中,当数据需要改变时,仅更改同一份数据,不一致性可以避免。

(5) 安全性保护。数据安全性是指保护数据以防止不合法的使用所造成的数据破坏和泄密,例如设置访问权限、对数据加密等。

(6) 数据完整性控制。数据完整性是指数据的正确性、有效性和相容性。数据库系统提供了必要的功能,保证了数据在输入、修改过程中始终符合原来的数据定义和规定。

(7) 并发控制。并发操作是指多个用户进程在同一时刻期望存取同一数据时发生的事件。为了避免并发进程间相互干扰进而导致错误的结果或破坏数据完整性,必须对多用户的并发操作加以控制和协调。

(8) 故障发现和恢复控制。在数据库系统运行中,由于用户操作失误和硬件及软件的故障,可能使数据库遭到局部性或全局性损坏,但系统能进行应急性处理,把数据库恢复到正确状态。

1.1.3 数据模型

数据模型(Data Model)是描述现实世界的工具,也是实现 DBMS 的基础。它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。数据模型应满足三方面要求:一是能比较真实地模拟现实世界;二是容易为人所理解;三是便于在计算机中实现。一种数学模型要很好地完成这三方面的要求在目前尚很困难,在数据库系统中针对不同的使用对象和应用目的,采用不同的数据模型。根据模型应用的不同目的,可以将这些模型划分为两类,它们分属于两个不同的层次。第一类模型是概念模型,也称信息模型,它是按用户的观点来对数据和信息建模,主要用于数据库设计。另一类模型是数据模型,主要包括网状模型、层次模型、关系模型等,它是按计算机系统的观点对数据建模,主要用于 DBMS 的实现。数据模型是数据库系统的核心和基础。一般来讲,数据模型是严格定义的一些概念的集合,这些概念精确地描述了数据、数据之间的联系、对数据的操作以及有关的语义约束规则,它通常是由数据结构、数据操作和完整性约束三部分组成。

(1) 数据结构。数据结构是指所研究的对象类型的集合,这些对象是数据库的组成部分,有两类:一类是指对象的数据类型、内容、性质;另一类是指数据之间的联系。

(2) 数据操作。数据操作是指对数据库中的各种对象的实例允许执行操作的集合,包括操作以及有关的操作规则。数据库主要有检索和更新(增、删、改)两大类操作。数据模型需要定义这些操作的准确含义、操作符号、操作规则以及实现操作的语句。

(3) 完整性约束。完整性约束规则是定义数据的约束条件,即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,用以限定数据库的状态及变化,以确保数据的正确、有效和

相容。

当前,数据库中所支持的几种主要的数据模型是层次型、网状型、关系型、面向对象型。

1. 层次模型

层次型数据库如树型结构,具有父子关系,如图 1-3 所示。每个父节点可以有許多子节点,但每个子节点仅有一个父节点。

层次型数据库是最基本的、最容易实现的数据库。它易于操作,可利用树状数据结构来完成。每一个节点有其具体的功能,如果需要寻找较远的节点,则必须先往上通过很多父节点,然后再往下寻找另一节点。显然,对于一个较大的数据库将会消耗很多搜索时间,而且如果需要的子节点有很多父节点,或者不同的父节点的子节点需要联系,则无法使用层次型结构。

层次型结构的特点如下:

- (1) 结构简单,易于操作;
- (2) 从上向下寻找数据容易;
- (3) 与日常生活的数据类型相当。

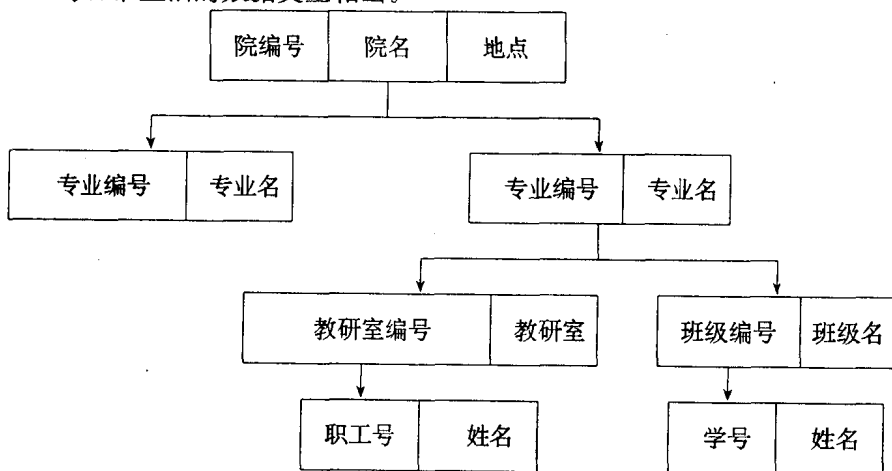


图 1-3 层次型数据库

2. 网状模型

网状数据库的结构如图 1-4 所示。网状数据库比层次型数据库更具有灵活性,允许子节点有多个父节点,并且子节点之间的关系较接近,联系很容易。明显的缺点是路径太多,当加入或删除数据时,涉及相关数据太多,不易维护与重建。网状数据库适合较稳定的数据库,不适合动态数据库,这是因为数据经常变动会造成重建的困难。

网状模型表达能力强,它能反映实体间的多对多的联系,但网状模型在概念上、结构上和使用上都比较复杂,而且对计算机的硬件环境要求较高。

层次模型和网状模型都是用指针来实现两个实体之间的联系的。它们都建立在图论的基础上,通常被称为格式化数据模型。

3. 关系模型

早期的数据库系统都采用格式化模型。1970 年,美国 E. F. Good 提出了关系模型的概念,首次运用数学方法来研究数据库的结构和数据操作,将数据库的设计从以经验为主提高到以理论为指导。关系模型是用表格形式表示实体之间联系的模型,它将数据的逻辑结构归结为满足一定关系的二维表格。关系模型的特点是:

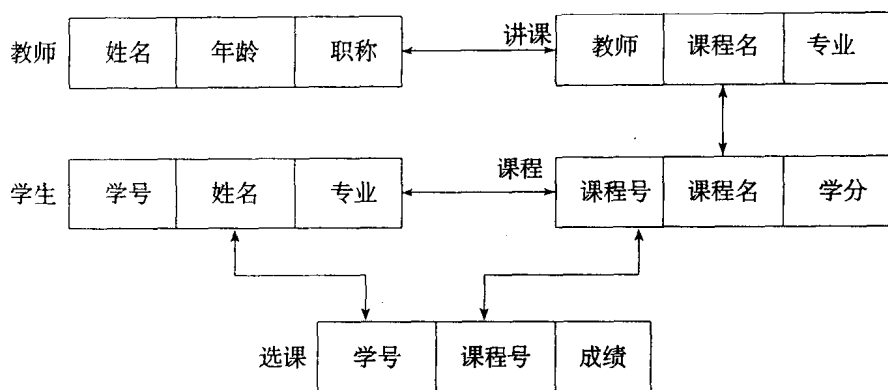


图 1-4 网状数据库

(1) 描述一致性,无论是实体还是实体之间的联系都用关系来表示。

(2) 可以直接表示多对多联系,比如“教师任课表”可表示一个教师担任几个班的教学,也可表示一个班有几个教师任教。

(3) 关系规范化,二维表格中每一栏目都是不可分的数据项,即不允许表中有表。

(4) 数学基础严密。

(5) 概念简单,操作方便,用户对数据的检索是从原来的表中得到一张新表,具体操作无需用户关心,数据的独立性高。

由于关系模型有很强的数据表示能力和坚实的数学基础,最容易被使用者接受,是目前应用最广泛的一种数据模型。Visual FoxPro 6.0 就是基于关系模型的数据库管理系统。

4. 面向对象模型

自 20 世纪 90 年代中期以来,人们看到了关系模型的一些严重缺陷,提出面向对象模型。面向对象模型一方面对数据结构方面的关系结构进行了改良,如 Oracle 8 就提供了面向对象模型的数据结构描述。另一方面,人们对数据操作引入了对象操作的概念和手段,使得今天的数据库管理系统基本上都提供了这方面的功能,本书将在以后的章节中详细介绍这种对象型操作的方法和技巧。一个面向对象模型是用面向对象观点来描述现实世界实体(对象)的逻辑组织、对象间限制、联系等的模型。面向对象模型是当前数据库中最为实用的模型。

1.1.4 关系型数据库

1. 关系数据库

关系型数据库是采用关系模型构成的一种数据库。它是一个表的集合,也就是说关系型数据库是由表组成的。一张表构成一个数据库文件,若干个数据库文件就形成了一个数据库。下面所示的就是一个学生管理数据库,它包含有三张表:学生表、课程表和成绩表。表 1-1 记载了有关学生的基本数据,表 1-2 记载了所开课程的信息,表 1-3 记载了学生选课的成绩信息。

表中的行称为记录,每一个记录包含了一个学生的情况。表中的列称为字段,每一个字段表示学生某方面的情况。表中第一行称为字段名,是各字段类型的集合,构成一个框架,这个框架就叫做数据库结构。在关系型数据库中,这些字段称为属性。作为关系型数据库中的表格应满足以下要求:

- (1) 表中不允许有重复的字段名;
- (2) 表中每一字段中的数据类型必须相同;

表 1-1 学生表

学 号	姓 名	性 别	出生日期	籍贯	入学成绩	英语四级过否	爱好和特长	简历
001001	尚 杰	男	11/20/82	北京	520.5	. T.	下棋	
003054	刘 晓	女	05/13/83	上海	512.0	. T.	看书	
002001	余芳习	女	12/26/82	广州	513.5	. F.	跳舞	
001057	张轶一	男	01/09/82	武汉	612.0	. T.	打乒乓球	
004023	陶红莉	女	02/14/81	青海	439.0	. F.	唱歌	
005437	皇甫俊	男	10/07/82	云南	426.0	. F.	吹笛子	
005412	杨 欣	男	01/01/80	武汉	589.5	. T.	下棋	
001039	马 松	男	04/16/81	新疆	441.0	. F.	踢足球	

表 1-2 课程表

课程编号	课程名称	学时	学分
001	英语	60	3
002	计算机	72	4
003	数学	90	5
004	哲学	60	3
005	文学	72	4
006	经济学	72	4

表 1-3 成绩表

学号	课程号	成绩
001001	001	83
001001	002	86
002006	003	90
002006	005	90
001057	006	79
003054	004	88
003054	002	82
004023	003	66

(3) 表中的记录的次序及字段的次序可任意排列;

(4) 一般表中的字段之间不应相互关联,如有“出生日期”字段,就不应有“年龄”字段;

(5) 主关键字一般应为惟一的。

学生表中包含了 8 条记录,每条记录有 9 个字段。为了区别每一条记录,作为主关键字的字段不能重复,例如学号字段是主关键字段,不可重复,但性别字段和籍贯字段的数据则可以重复。

2. 关系型数据库的三种基本的关系操作

(1) 选择。选择是指依据一定的条件选择若干个符合条件的记录(在二维表格中选择行),它是一种水平方向上的选择。例如,从表 1-1 中找出所有入学成绩大于 580 分的记录,通过选择操作来完成,选择结果如表 1-4 所示。

表 1-4 学生表的选择结果

学 号	姓 名	性 别	出生日期	籍贯	入学成绩	英语四级过否	爱好和特长	简历
001057	张轶一	男	01/09/82	武汉	612.0	. T.	打乒乓球	
005412	杨 欣	男	01/01/80	武汉	589.5	. T.	下棋	

(2) 投影。投影是指从一个数据库文件中选择若干个字段(在二维表格中选择列)进行操作,它是一种垂直方向(即列方向)上的选择。投影关系是指从同一表格中得到一个更小的表格,例如表 1-5 是学生简表,表 1-5 和表 1-1 之间为映射(即投影)关系。

(3) 连接。在关系数据库中,可以将相关的表格通过关键字连接在一起,这种连接方法称为连接关系。利用连接关系可以得到一个新的表格。

表 1-2 是课程表,表 1-3 是学生成绩表,表 1-2 和表 1-3 有一个字段(课程号字段)是相同的,因此,可以利用此字段将这两个数据库连接起来,使成绩表里可以查看课程名称和学分。表 1-6 是连接结果。

表 1-5 投影结果

学号	姓名	性别	入学成绩
001001	尚杰	男	520.5
003054	刘晓	女	512.0
002001	余芳习	女	513.5
001057	张轶一	男	612.0
004023	陶红莉	女	439.0
005437	皇甫俊	男	426.0
005412	杨欣	男	589.5
001039	马松	男	441.0

表 1-6 连接结果

学号	课程号	成绩	课程名称	学分
001001	001	83	英语	3
001001	002	86	计算机	4
002006	003	90	数学	5
002006	005	90	文学	4
001057	006	79	经济学	4
003054	004	88	哲学	3
003054	002	82	计算机	4
004023	003	66	数学	5

表格和表格间的关系完全由使用者决定,有的使用者仅需要表格中的部分字段,但有的使用者会利用多个较小的表格组成更大的表格,进而产生更复杂的关系。

3. 关系数据库的优点

在三种数据库结构中,关系数据库具有突出的优点,主要表现如下:

- (1) 表格式的关系较易建立数据库;
- (2) 使用者较易于理解表格式的关系;
- (3) 层次型与网状型结构可以较易转变成表格式结构;
- (4) 由于极易实现映射和联结关系,因此新关系的建立也易于实现;
- (5) 由于表格式的搜索比线性结构更容易,所以搜索速度快;
- (6) 关系型结构易于修改。

1.2 Visual FoxPro 6.0 的发展史和性能特性

1.2.1 Visual FoxPro 的发展史

Ashton-Tate 公司的 dBASE II 版,经过大力宣传、再开发、维护和培训,到 dBASE III 已经成了微机上最受欢迎的数据库产品之一,被称为“大众数据库”、“工业标准”。人们之所以喜欢这个系统,主要原因是该系统简单、易学、实用、方便、便于掌握和具有再开发性。

为了打破 dBASE 一统天下的局面,各公司推出了一系列称为“XBASE”的兼容产品,这些产品同样很受欢迎。其中最突出的就是 Fox 公司的 FoxBASE,它的最为突出的一个优点就是比 dBASE 快,因为它多用了一百多 KB 的内存空间,从而换来了时间,其他完全与 dBASE 兼容。这个系统一问世,就吸引了 dBASE 的大部分用户。

Fox Software 公司从 1985 年开始,陆续推出了 FoxBASE 1.0、FoxBASE 2.0、FoxBASE+ 2.1、FoxPro 1.0、FoxPro 2.0。1992 年 6 月并入软件巨人 Microsoft 公司后,借助于 Microsoft 丰富的资源,于 1993 年推出了 FoxPro 2.5,1995 年 6 月推出了 Visual FoxPro 3.0,1997 年 4 月推出了 Visual FoxPro 5.0。

Microsoft 公司紧接着又在 1998 年 9 月份推出了 Visual Studio 98,其中包含了 Visual