

Visual FoxPro

数据库程序设计

主编 刘秋生

Visual FoxPro

数据库程序设计

主 编 刘秋生

副主编 徐红梅 刘晓松 赵广凤

 江苏大学出版社
JIANGSU UNIVERSITY PRESS

镇 江

中国大学图书馆

(C280M48-1120) (2011) 第1120号

图书在版编目(CIP)数据

Visual FoxPro 数据库程序设计 / 刘秋生主编. —
镇江: 江苏大学出版社, 2017. 9
ISBN 978-7-5684-0507-2

I. ①V… II. ①刘… III. ①关系数据库系统—程序
设计 IV. ①TP311.138

中国版本图书馆 CIP 数据核字(2017)第 144176 号

Visual FoxPro 数据库程序设计

Visual FoxPro Shujuku Chengxu Sheji

主 编/刘秋生

责任编辑/徐 婷

出版发行/江苏大学出版社

地 址/江苏省镇江市梦溪园巷 30 号(邮编: 212003)

电 话/0511-84446464(传真)

网 址/http://press. ujs. edu. cn

排 版/镇江华翔票证印务有限公司

印 刷/句容市排印厂

开 本/787 mm×1 092mm 1/16

印 张/19.25

字 数/459 千字

版 次/2017 年 9 月第 1 版 2017 年 9 月第 1 次印刷

书 号/ISBN 978-7-5684-0507-2

定 价/42.00 元

如有印装质量问题请与本社营销部联系(电话:0511-84440882)

前 言

两化深度融合深刻地影响我国经济建设和管理学的发展,然而,信息化建设完全依赖于数据库系统的应用水平与能力。自数据库的概念形成以来,随着信息技术的高速发展和信息系统的广泛应用,数据库管理系统的理论与方法不断完善。Visual FoxPro 作为数据库管理工具软件不仅简便易懂,而且已经成为经济管理类学生必须掌握的基本技能。

本书是为经济管理类专业学生培养信息素质和信息管理技能,构建课程体系知识链的核心环节,是作者在多年从事数据库系统应用和数据库系统理论与方法研究,以及数据库系统在管理信息系统研发实践和教学经验积累的基础上编写而成的。

本书的主要特点是:

1. 内容精练。在内容上作了精心的安排,以目前操作简便,容易掌握、理解,实用性强,应用面广的面向对象的关系型数据库管理系统 Visual FoxPro 作为数据库系统的开发工具。围绕 Visual FoxPro 的基本概念、基本操作,由浅入深、系统地介绍关系型数据库管理系统的功能和数据处理的方法。

2. 突出重点。全书围绕三个中心,分别重点介绍数据库的基础知识、数据库的基本操作和数据库系统应用编程设计。

3. 实用性强。强调了理论与实践相结合,把学生容易掌握理解的学生学籍管理作为典例,并采用图文并茂的方式,便于阅读理解。

4. 系统性强。从数据库系统的应用着手,认识数据库在信息系统的作用,全面介绍了数据的收集、传输、存储、加工、维护和使用的的基础知识和基本操作。

5. 适用面广。本书是面向经济管理类专业学生的教学用书,也可以作为计算机专业学生和其他工程技术人员的自学用书。

全书共分 10 章。总课时为 64 学时,其中上机实验课为 20 学时。有条件的情况下,可安排多媒体教室上课 30 学时,同时还应该适当安排学生利用课余时间独立上机完成数据库操作练习。各院校可以根据实际情况按上述比例压缩或增加学时。

本书的构思是由江苏大学刘秋生教授完成,江苏大学刘秋生、刘晓松、徐红梅和赵广凤共同参与编写。江苏大学管理学院信息系承担数据库程序设计课程的老师对本书组稿、复核、数据处理等提出了宝贵建议,在此一并表示衷心感谢!

在编写过程中立足以实用、易懂、突出重点为准绳,在内容上反复提炼,精益求精。文字上反复推敲,语言上立足通俗,采用最简练的语言,介绍较先进的技术。然而由于作者水平有限,错误难免发生,请读者提出宝贵意见!

编 者

2017 年 5 月于镇江

目 录

第1章 数据库基础知识	(001)
1.1 数据库基本概念	(001)
1.1.1 数据库	(001)
1.1.2 数据模型	(004)
1.1.3 数据库系统	(005)
1.1.4 数据库技术发展	(010)
1.2 关系型数据模型	(011)
1.2.1 关系型数据库概念	(011)
1.2.2 关系型理论基础	(014)
1.2.3 关系的范式	(017)
1.3 数据库系统的研究与应用	(019)
1.3.1 数据库技术研究范围	(019)
1.3.2 数据库技术应用领域	(021)
1.3.3 数据库技术发展动态	(023)
本章小结	(024)
第2章 数据库管理系统	(026)
2.1 Visual FoxPro 产品与特点	(026)
2.1.1 Visual FoxPro 产品	(026)
2.1.2 Visual FoxPro V6.0 的特点	(027)
2.2 Visual FoxPro 安装、启动与退出	(028)
2.2.1 Visual FoxPro 安装	(028)
2.2.2 Visual FoxPro 启动	(029)
2.2.3 Visual FoxPro 退出	(029)
2.3 Visual FoxPro 用户界面及其规则	(030)
2.3.1 Visual FoxPro 界面	(030)



2.3.2	Visual FoxPro 操作方式	(033)
2.3.3	Visual FoxPro 命令书写规定	(034)
2.3.4	Visual FoxPro 文件类型	(035)
2.3.5	Visual FoxPro 系统环境设计	(035)
2.4	Visual FoxPro 数据类型、常数、变量	(037)
2.4.1	Visual FoxPro 数据类型	(037)
2.4.2	常数	(038)
2.4.3	变量	(039)
2.4.4	数组	(042)
2.5	Visual FoxPro 运算符与表达式	(043)
2.5.1	Visual FoxPro 空值	(043)
2.5.2	运算符与表达式	(044)
2.6	Visual FoxPro 函数	(048)
2.6.1	数学运算函数	(048)
2.6.2	字符处理函数	(050)
2.6.3	日期与时间函数	(053)
2.6.4	转换函数	(054)
2.6.5	其他常用函数	(056)
2.7	Visual FoxPro 项目管理器	(059)
2.7.1	项目管理器的创建与修改	(059)
2.7.2	项目管理器的选项卡	(060)
2.7.3	项目管理器的定制	(060)
2.7.4	项目管理器的操作	(061)
	本章小结	(062)
第3章	数据表、库设计	(063)
3.1	数据表设计	(063)
3.1.1	数据表的分类	(063)
3.1.2	数据表的组成	(063)
3.1.3	数据表的设计过程	(064)
3.2	数据表结构操作	(066)
3.2.1	创建数据表结构	(066)
3.2.2	显示数据表结构	(071)

3.2.3 修改数据表结构	(071)
3.3 数据表记录操作	(073)
3.3.1 给数据表录入记录	(073)
3.3.2 输出数据表记录	(075)
3.3.3 修改数据表记录	(076)
3.3.4 删除数据表记录	(079)
3.3.5 设置记录过滤条件和字段筛选	(080)
3.4 数据表操作	(080)
3.4.1 打开与关闭数据表	(081)
3.4.2 数据表的复制	(082)
3.4.3 数据表记录定位操作	(082)
3.4.4 数据表数值统计	(090)
3.4.5 工作区	(090)
3.4.6 多用户模式	(091)
3.4.7 多工作区环境下数据表操作	(094)
3.5 数据库的设计与基本操作	(096)
3.5.1 数据库设计过程	(096)
3.5.2 数据库的创建	(097)
3.5.3 数据库的打开、选择和关闭	(099)
3.5.4 创建数据库表	(099)
3.5.5 添加数据库表	(100)
3.5.6 移去数据库表	(101)
3.6 数据库表的设计	(102)
3.6.1 设置字段显示属性	(102)
3.6.2 设置表属性	(107)
3.6.3 创建表之间的关系	(109)
3.6.4 设置参照完整性	(110)
本章小结	(111)
第4章 查询与视图设计	(113)
4.1 查询与视图的作用和同异	(113)
4.1.1 查询与视图的作用	(113)
4.1.2 查询与视图的联系与区别	(113)



4.2 查询设计	(114)
4.2.1 查询设计的主要方法与步骤	(114)
4.2.2 查询设计中常用的函数	(114)
4.2.3 用查询设计器设计查询	(116)
4.2.4 用 SELECT-SQL 命令设计查询	(123)
4.2.5 用查询向导设计查询	(127)
4.2.6 查询文件的使用	(130)
4.3 视图设计	(130)
4.3.1 创建视图	(131)
4.3.2 修改视图	(131)
4.3.3 视图的使用	(132)
本章小结	(134)
第5章 程序、过程设计	(135)
5.1 程序设计基础	(135)
5.1.1 建立与编辑程序文件	(136)
5.1.2 程序文件的调试与运行	(137)
5.1.3 程序中常用的命令	(138)
5.1.4 结构化程序设计概念及顺序程序结构	(141)
5.2 分支程序设计	(142)
5.2.1 简单分支语句	(142)
5.2.2 二分支语句	(143)
5.2.3 分支嵌套	(144)
5.2.4 多分语句	(145)
5.2.5 分支语句格式说明	(147)
5.3 循环程序设计	(147)
5.3.1 DO WHILE 循环语句	(148)
5.3.2 FOR 循环语句	(149)
5.3.3 SCAN 循环语句	(150)
5.3.4 循环嵌套	(150)
5.3.5 LOOP 和 EXIT 语句的使用	(152)
5.4 过程与自定义函数	(153)
5.4.1 过程的种类	(153)

5.4.2	过程文件的使用	(155)
5.4.3	自定义函数的定义	(155)
5.4.4	过程及函数的调用方法	(156)
5.4.5	过程及函数调用中的参数传递	(156)
5.4.6	程序运行中的变量作用域	(159)
	本章小结	(163)
第6章 面向对象程序设计基础		(164)
6.1	类和类的特性	(164)
6.1.1	类	(164)
6.1.2	基类	(164)
6.1.3	类的特性	(165)
6.2	对象	(166)
6.2.1	属性	(166)
6.2.2	事件	(167)
6.2.3	方法	(167)
6.3	对象的引用、属性值的设置和方法的调用	(168)
6.3.1	对象的引用	(168)
6.3.2	属性值的设置	(168)
6.3.3	方法的调用	(169)
6.4	常用事件的触发顺序和常用方法	(169)
6.4.1	对象的层次关系	(169)
6.4.2	主要事件的触发顺序	(170)
6.4.3	控制事件循环	(171)
6.4.4	常用方法	(171)
	本章小结	(172)
第7章 表单与类设计		(173)
7.1	表单的设计	(173)
7.1.1	表单设计步骤	(173)
7.1.2	表单设计实例	(174)
7.2	表单的创建及管理	(175)
7.2.1	用向导创建表单	(175)



7.2.2	用表单设计器创建表单	(176)
7.3	表单控件的设计	(182)
7.3.1	控件与数据的关系	(182)
7.3.2	表单控件的设计和应用	(183)
7.4	类的设计和应用	(192)
7.4.1	设计和创建子类	(192)
7.4.2	类的应用	(197)
	本章小结	(198)
第8章 报表和标签设计 (199)		
8.1	报表与标签设计	(199)
8.1.1	报表与标签的组成	(199)
8.1.2	报表的类型	(200)
8.1.3	报表与标签的设计步骤	(200)
8.2	报表的创建及调用	(201)
8.2.1	报表的创建	(201)
8.2.2	报表文件的修改	(208)
8.2.3	报表的调用	(208)
8.3	标签的创建及调用	(209)
8.3.1	标签的创建	(209)
8.3.2	标签的调用	(211)
8.3.3	标签格式文件的修改	(212)
	本章小结	(212)
第9章 菜单设计 (213)		
9.1	菜单设计过程	(213)
9.1.1	菜单的分类	(213)
9.1.2	菜单的组成	(214)
9.1.3	菜单的规划	(214)
9.2	创建菜单系统	(217)
9.2.1	普通菜单的创建	(217)
9.2.2	快捷菜单的创建	(222)
9.2.3	SDI 菜单的创建	(223)

9.3 菜单的使用	(224)
9.3.1 菜单的调试与运行	(224)
9.3.2 菜单的维护	(225)
9.3.3 Visual FoxPro 系统菜单的配置	(226)
本章小结	(227)
第 10 章 数据库系统综合应用	(228)
10.1 学生成绩管理	(228)
10.1.1 学生成绩管理组织结构系统分析	(228)
10.1.2 学生成绩管理主要业务流程分析	(229)
10.1.3 学生成绩管理总体结构分析	(231)
10.2 学生成绩管理功能实现	(232)
10.2.1 用户登录模块	(232)
10.2.2 人员管理	(235)
10.2.3 课程管理	(238)
10.3.4 成绩管理	(240)
10.3 系统设置与项目编译过程	(241)
10.3.1 屏幕设置	(241)
10.3.2 状态设置	(243)
10.3.3 项目编译过程	(250)
本章小结	(252)
附录 A 函数表	(253)
附录 B 文件类型	(269)
附录 C 控件和对象表	(270)
附录 D 事件表	(272)
附录 E 方法表	(275)
附录 F 属性表	(278)
附录 G 环境配置	(289)
附录 H 系统变量	(292)
参考文献	(295)

第 1 章

数据库基础知识

数据库的理念提出至今历经了 60 多年的发展,形成了坚实的理论基础和独特的数据库技术,其内涵不断深入,应用广泛,技术飞速发展。数据不仅成为信息的基本载体,而且数据库技术推动了现代管理技术的进步,成为现代化管理的基石,深刻地影响着人们的生活方式、工作方式和社会环境,改变着人们的思想观念和行为规范。管理现代化的需求给数据库系统提供了宽广的应用途径,也给数据库技术提出了新的课题。数据库技术、管理技术、信息系统相互渗透,促进了企业信息化、全球数字化、资源一体化的快速发展。

1.1 数据库基本概念

1.1.1 数据库

1.1.1.1 数据

数据是描述事物特征特定的符号(也称为数据项或字段),是数据库组织和数据处理中最基本的单元。它不仅包括人们日常工作中所熟悉的数字,还包含在描述事物过程中经常采用的文字、图像、图形、声音等形式,这些也属于数据库中的数据范畴。数据是人们传达思想、进行信息交流的载体。

(1) 数据的定义

通过数据将事物的信息及时、正确、全面地描述或记录下来是数据处理过程中的关键。描述一个事物往往涉及许多方面的概念和理论。

描述事物的用途直接影响着事物属性的定义。对于不同的用户,因其需求不同,侧重点不同,需要的信息自然不同,因此,在描述事物时使用的属性也不相同。例如,把人作为描述的对象时,首先要确定描述人的用途,若描述学生,则将涉及学生的学习情况、政治思想表现、工作能力和学习能力等方面的基本内容;若描述一个职工,则将涉及职工的工资、保险、工作能力和技术等级等。当描述一个学生的学习情况时,针对的是一个较具体的数据处理需求,可以通过学生的姓名、学号、专业、年级、开课时间、课程名称、考试类别和考试成绩等属性反映每一位学生的每一门课程的学习成果,这些属性通过人们日常交流中所约定的符号而被转换成数据。

数据来源于人们的日常工作和生活,存储于各种媒体中,经过加工、传送为人们的工



作、生活服务。

(2) 数据的描述

数据的描述是从客观事物出发,经过概念、规则或逻辑推理转换成数据,这一过程经历了三个领域即三个世界的转换(现实世界、概念世界和数据世界),如图 1.1 所示。

① 现实世界。这是存在于人们头脑之外的客观世界,由不以人的意志为转移的客观实体(事物)组成。所在事物都可分成“对象”与“性质”两大类,又可分为“特殊事物”与“共事物”两个重要级别。

② 概念世界。这是人们对现实世界中事物的规范、约定和习俗等在人们头脑中的反映;由一切定义、定理、规则等组成,也称为逻辑世界。现实世界的事物往往通过各种属性的表达转换成概念世界。

③ 数据世界。这是概念世界中信息的数据化过程。现实世界事物属性值和用数据模型描述概念世界的定义、定理、规则及联系将转换成数据世界。

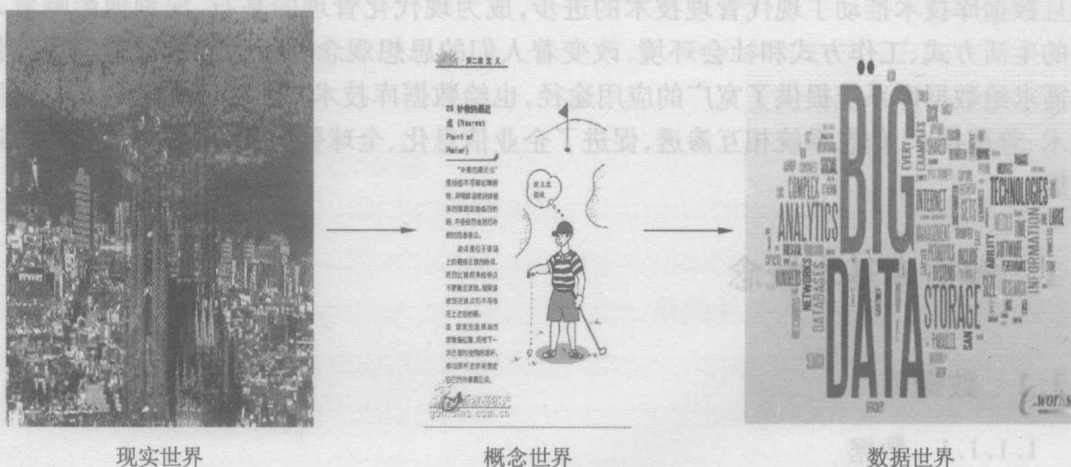


图 1.1 数据三大领域转换示意图

(3) 数据模型

这是一种描述数据与数据之间的联系及有关的语义约束规则的方法,也是数据库设计遵循的基本原则。目前使用的数据模型大体可分为两种类型:一类是独立于任何计算机实现的数据模型,如实体—联系模型、语义网络模型等。这类数据模型完全不涉及信息在计算机系统内的表示方式、方法,只用来描述某个特定的企业组织所关心的信息结构,因而又常常被称为信息模型或概念数据模型;另一类数据模型则直接面向数据库中数据的逻辑结构,这是我们常遇到的数据模型。为了与概念数据模型相区别,不妨把它们称作基本数据模型,这也是本书重点讨论的数据模型。

按照著名的数据库专家 E. F. Codd 的理论,一个基本数据库模型实质上是一组向用户提供的规则,这组规则规定数据结构如何组织,以及应当允许进行任何操作。一般来说,一个数据库的基本数据模型至少包含以下三个组成部分:数据结构、数据操作和数据的完整性约束。

① 数据结构。数据结构可以看成是数据集合的描述,它主要包含两部分的内容,一是数据集合的元素,即数据类型、内容、性质;二是数据之间的联系。对于不同的联系方

式,相应的数据集中元素的含义也有所不同。按数据之间的联系方式,数据模型可分成层次型、网状型和关系型三种。

② 数据操作。这是指对数据库中各对象实例允许执行的操作集合,包括操作及有关的操作规则。数据库日常操作主要有检索和维护两类。在数据模型中需定义这些操作的含义、操作符号和操作规则等。

③ 数据的完整性约束。数据的完整性约束是完整性规则的集合。它是给定的数据模型中数据及其联系所具有的约束和依存规则,用于限定符合数据模型的数据库状态及状态变化,以保证数据正确、有效和相容。

采用不同的数据模型,数据库的数据结构、数据操作和数据的完整性约束也不相同。

1.1.1.2 数据库

(1) 数据库的内涵

数据库是有用数据的有序集合,从形式上看数据库是数据的仓库,对数据进行组织、存储和管理。

数据库中数据组织的最小单元是数据项,这是独立的不可分割的处理单元,描述了事物的某一属性。描述一个事物某一用途的全部属性的数据集称为记录,如一个学生一门课程或一个学期的学习成绩等,即记录是由数据项组成;描述一类相同属性事物的记录集称为数据文件,如学生成绩表,在数据文件中数据存取的最小单位是记录,数据文件是由记录组成的,在数据库组织中数据文件被形象地称为表;数据库是由数据文件组成的,在数据库中记载了各张表的特征和表间的联系。一个数据库可以记录、加工、传递知识,网络教育、电子商务、信息系统等无不建立在数据库的基础之上。

(2) 数据库的特性

数据库技术自产生以来形成了较完善的理论体系和强大的数据处理功能。数据处理进入数据库技术阶段以前,经历了人工管理阶段、文件管理阶段。数据库具有如下特点:

① 数据共享性。一个数据库可以供多种不同的用户使用,如学生成绩数据表,可以供学生、学校和用人单位等多种用户使用。在数据库里,数据与程序独立,提高了数据的使用价值,同时简化了程序设计,提高了程序的灵活性,方便了用户的操作。

② 数据的一体化和结构化。数据库按某种模型组织、存储和处理数据,不仅使内部数据之间彼此相关,而且文件之间在结构上也有有机地联系在一起,整个数据库形成一个整体,即数据库的一体化,这样使数据库具有较大的适应性,易于维护与扩充,应用数据灵活方便。

③ 较少的冗余度。数据库的数据组织是从描述事物的整体出发,数据的冗余大大减少,在数据文件中除了作为表间联系的关键字和为了数据安全、可靠所采取的备份副本之外,存储的数据冗余度保持在尽可能小的程度。

④ 数据独立性好。数据库系统提供了数据的映射功能,当需要改变存储结构时,逻辑结构可以不改变,从而避免了不必要的程序修改工作。

⑤ 对数据进行集中统一的控制。系统提供统一的数据定义、预处理、查询及维护等手段,并统一控制数据的安全性、完整性、保密性和并发性,使得对数据的应用更加有效和可靠。

1.1.2 数据模型

数据模型揭示了现实世界中各类事物之间的联系,根据其联系的方式和描述的手段可以将数据模型分成层次型数据模型、网状型数据模型和关系型数据模型。这里只简单介绍前两种,关系型数据模型将在下一节介绍。

(1) 层次型数据模型

层次型数据模型是数据库技术中应用最早的、与人们日常生活联系最密切的一种数据模型。它把数据库结构描述成一个有序树的集合,这棵树的每一个结点是由若干数据项组合而成的逻辑记录。用层次型数据模型描述的主要特点是自然、直观、结构简单、层次清晰、易于理解。

事实上层次型结构描述了管理体系结构中的一种方式,它是从总体到具体的一个细化过程,对总体的描述我们把它作为0层,逐步细化分成1层、2层等。上下层之间形成父子关系。在这种结构中,0层结点没有父结点,其他层只允许有一个父结点,可以有若干个子结点,最底层没有子结点。在层次模型中,每个结点表示一个记录类型,结点之间的连线表示记录类型之间的联系,这种联系只能是一对多联系。每个记录类型可包含若干个字段,记录类型用来描述实体,字段用来描述实体的属性。

在现实世界中,具有层次型联系的事物很多,行政管理机构、产品结构、家族结构、教学体系(如图1.2所示学生学籍管理数据模型)等都属于层次型结构。用层次模型对具有一对多层次关系的部门进行描述非常自然、直观和容易理解。这是层次数据库的突出优点,但层次模型有两个缺点:其一,层次型数据模型所能体现的记录联系只限于函数型的,即一对多的联系方式(如学生对课程,教师对学生),在现实世界中事物之间的联系还有多对多的联系,这些复杂的数据联系在层次型中受到限制。其二,层次型数据模型中全部数据只能以有序树的形式组织起来,这种单一固定的数据模型使得数据库适应变化的能力弱,同时增加了系统开销。这一缺陷使得它很难适应许多信息系统需求动态变化的应用环境。

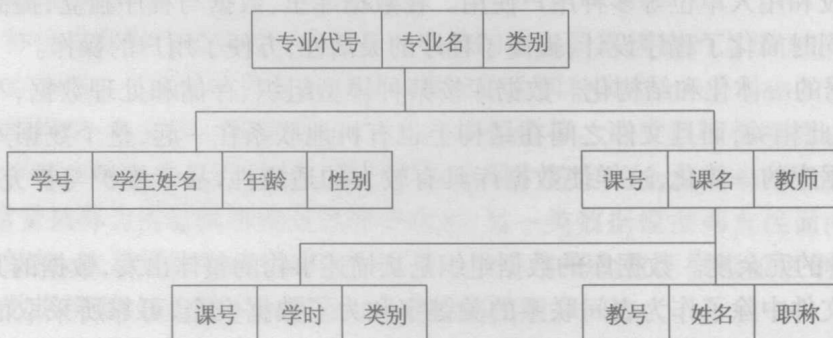


图 1.2 学生学籍管理数据模型

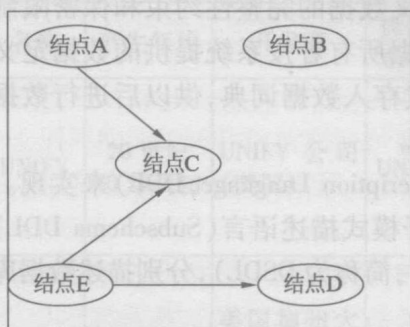
基于层次模型的数据库系统称为层次数据库系统。1968年IBM公司推出的IMS系统是典型的层次模型系统,在20世纪70年代得到了广泛应用。

(2) 网状型数据模型

在事物之间存在的多种对应关系被称为多对多的联系,如制造企业对新产品研制开

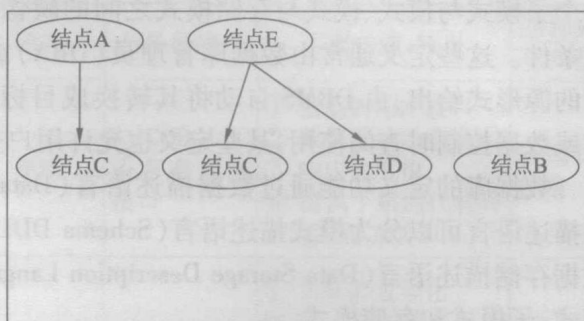
发工作的组织管理,它不仅受到产品生命周期的影响,同时也受到企业各个职能部门的影响,这项工作同时受到多方面的管理。在这种情况下,记录之间是以网络方式实现联系,因此采用网状型数据模型是最恰当的方法。

网状模型用网状结构表示各类实体及其之间的联系。在网状型数据模型中最基本的数据结构仍然是逻辑记录。记录之间用系型来组织,分成单从系型、多从系型和奇异系型三种类型。每一个系型是从一般模式中分离出来的一个二级树,其中只有一个记录处于突出的地位,称为主记录,其余的记录处于从属地位,称为从记录,如图 1.3 和图 1.4 所示。在这种方式下记录之间的联系是通过系型实现的,一个系型被定义为记录的一个非空集合。



单从系

图 1.3 一般网状模型



多从系

奇异系

图 1.4 系型的种类

网状模型是一种比层次模型更具普遍性的结构,它可以更直接地描述现实世界,但是网状模型的数据结构比较复杂,数据独立性较差。基于网状模型的数据库系统称为网状数据库系统。1969年美国数据系统语言委员会(CODASYL)下属的数据库任务组(DBTC)提出的报告中的数据模型是网状模型的主要代表。20世纪70年代也出现了许多基于网状模型的系统,如Cullinet公司的IDMS、HP公司的IMAGE/3000等。

在网状型数据模型中,记录之间的函数性联系隐含着以下一些固有的结构约束。

- ① 在任何时候任何一个系值中至多只有一个主记录。
- ② 任何一个记录在同一系型中至多只能参与其中一个系值(但同一记录可以联系到不同系型中的几个系上)。
- ③ 允许用户自由地定义一些专用过程来对数据库的数据进行处理所必需的其他语义约束。

1.1.3 数据库系统

1.1.3.1 数据库管理系统

数据库管理系统(Data Base Management System, DBMS)是指数据库系统中对数据进行管理的软件系统,它是数据库系统的核心部分。数据库系统的一切操作,包括查询、维护及各种控制都是通过DBMS进行的。

DBMS是基于某种数据模型上的。因此,可以把它看成是某种数据模型在计算机系统上的具体实现。根据采用的数据模型不同,DBMS可分为层次型数据库管理系统(HDBMS)、网状型数据库管理系统(NDBMS)和关系型数据库管理系统(RDBMS),但



对于不同的计算机系统,由于缺乏统一的标准,即使相同类型的 DBMS,它们在用户接口、系统功能等方面也常常是不同的。对于相互兼容的相同 DBMS 建立的数据库之间可以直接交换数据,而不同的数据模型的 DBMS 建立的数据库文件之间是不能直接交换数据的。

(1) 数据库管理系统的功能

DBMS 是组织、存储和处理数据的规则和实现这些功能的软件。使数据成为一个可管理的资源不仅易于实现共享,也增强了数据的安全性、完整性和可用性,并可提供高度的数据独立性。具体来说,一个比较完善的 DBMS 至少应该具有如下功能。

① 数据库定义功能。这是指定义数据库的结构,包括模式、存储模式和子模式,以及每个子模式与模式、模式与存储模式之间的映像;定义数据的完整性约束和保密限制约束条件。这些定义通常由数据库管理员(DBA)或数据所有者按系统提供的数据库定义语言的源形式给出,由 DBMS 自动将其转换成目标形式存入数据词典,供以后进行数据操作或数据控制时查阅使用,某些定义也允许用户查阅。

数据库的定义功能通过数据描述语言(Data Description Language, DDL)来实现。数据描述语言可以分为模式描述语言(Schema DDL)、子模式描述语言(Subschema DDL)和数据存储描述语言(Data Storage Description Language, 简称为 DSDL),分别描述数据库的模式、子模式和存储模式。

② 数据库操纵功能。它包括数据初始装入、对数据的存取和维护操作、数据库结构的维护和重新组织、数据转存等,系统提供统一的数据操纵语言,允许用户根据需要在授权的范围内自由地进行上述操作。

数据库的操纵功能是通过数据操纵语言(Data Manipulation Language, DML)来描述的,它提供了用户或应用程序访问数据库系统的接口。DML 是一般集合型的操作,是一种高级的数据处理语言。应用这些语句,可以让用户很方便地对数据库中的记录进行各种插入、删除、修改、统计和查询处理,我们将在 Visual FoxPro V6.0 数据库应用开发工具中详细介绍这些语句的应用。

③ 数据控制功能。DBMS 对数据库的控制主要包括三个方面:数据安全性控制、数据完整性控制及在多用户环境下的并发控制等。

数据安全性控制是对数据库的一种保护,它的作用是防止数据库中的数据未经许可地被用户访问,并防止用户有意或无意地对数据库造成破坏性修改。

数据完整性控制是 DBMS 对数据库提供保护的另一个重要方面。我们知道,数据的价值在于它们的正确性,在于它们正确地表达了现实世界中客体的信息,而这些客体的各种信息往往具有某些固定联系,这些联系体现为数据本身的内涵和各种关联关系,与内涵相矛盾的数据显然是无意义的,因而是必须避免的。完整性控制的目的是保持进入数据库中的存储数据的正确性和有效性,防止任何操作对数据造成违反其本意的改变。

并发控制的策略包括封锁单位大小的确定,死锁的预防、检测和解除等。DBMS 还具有系统缓冲区的管理及数据存储的某些自适应调节机制等其他控制功能。

④ 数据通信功能。提供与操作系统的联机处理、与分时系统及远程作业输入的相应接口、与网络软件的通信功能等。