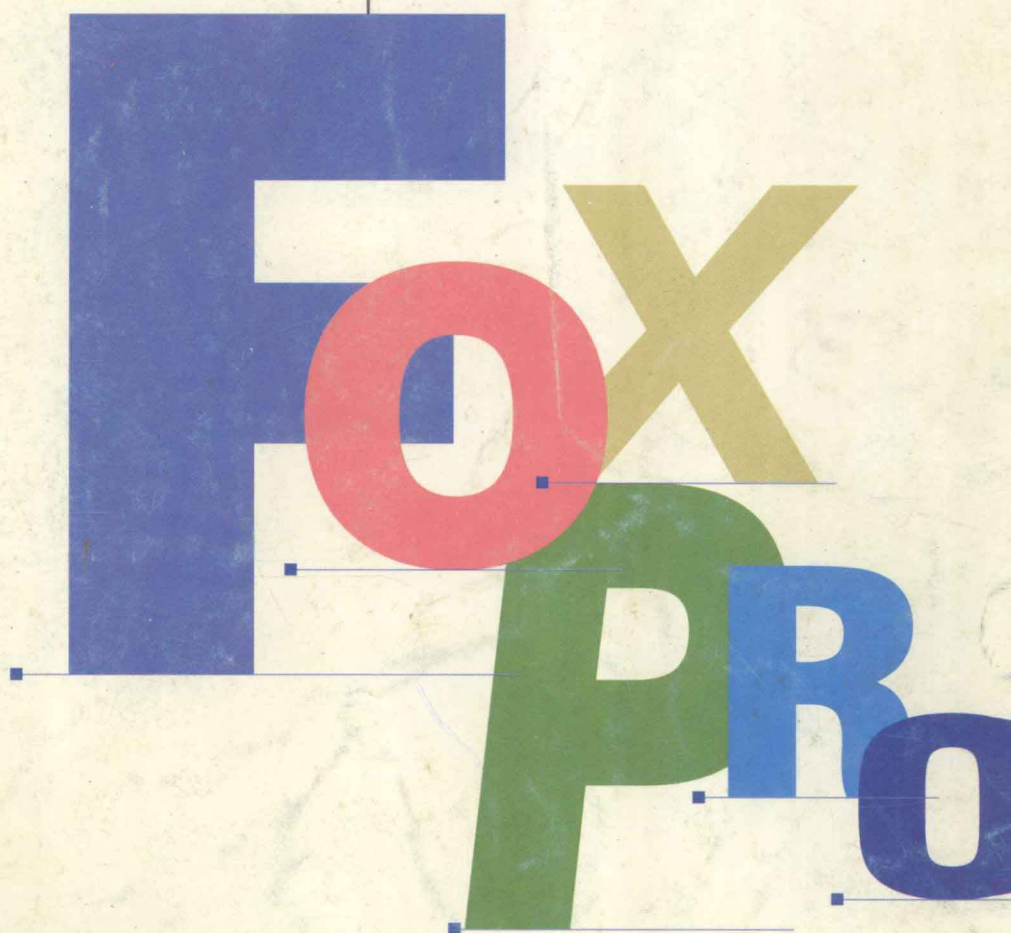




宗子安 许第洪
湛霖霖 编著

及其在工程中的运用



湖南大学出版社

FoxPro 及其在工程中的运用

宗子安 许第洪 湛霖霖 编著

江苏工业学院图书馆
藏书章

湖南大学出版社

2000年·长沙

内 容 提 要

本书是为适应高等院校工科专业普遍开设的数据库技术课程改革、更新及广大工程技术人员学习、掌握、应用数据库技术的需要而编写的,以淡化专业、拓宽应用基础为总的思路,力争反映数据库最新发展动态,着重培养读者解决实际问题和创新的能力,力求从基本理论和实际应用能力的培养两方面为读者提供帮助和支持。

除全面介绍 FoxPro 系统的基本概念及基本操作以外,重点放在实用程序设计及应用系统开发方面。通过精心组织典型工程应用例题让读者掌握数据库技术与工程实际相结合的方法与技巧,并与后续的 CAD 技术等课程的内容有机地结合起来,提高读者综合应用计算机技术的能力。

本书行文简洁、流畅,在内容的安排上由浅入深、循序渐进、图文并茂,各章配有精心组织的习题,附录给出了 FoxPro 系统的所有命令及函数的索引以供读者查阅,十分便于教学与自学。本书可作为高等院校工科专业数据库技术课程的教材或教学参考书。

图书在版编目(CIP)数据

FoxPro 及其在工程中的应用/宗子安,许第洪,谌霖霖编.

—长沙:湖南大学出版社,2000.8

ISBN 7-81053-320-7

I.F... II.宗...②谌...③许... III.关系数据库-数据库
库管理系统,FoxPro IV.TP311.13

中国版本图书馆 CIP 数据核核字(2000)第 68939 号

FoxPro 及其在工程中的运用

Fox Pro JiQi Zai Gongcheng zhong de Yunyong

宗子安 许第洪 谌霖霖 编著

☐责任编辑 戴东宁

☐出版发行 湖南大学出版社

社址 长沙岳麓山 邮码 410082

电话 0731-8821691 0731-8821315

☐经 销 湖南省新华书店

☐印 装 湖南大学印刷厂

☐开本 787×1092 16 开 ☐印张 17 ☐字数 435 千

☐版次 2000 年 8 月第 1 版 ☐2000 年 8 月第 1 次印刷

☐印数 1-4 000 册

☐书号 ISBN 7-81053-320-7/TP·18

☐定价 24.00 元

(湖南大学版图书凡属印装差错,请向承印厂调换)

前 言

在现代社会中,信息与数据量急剧增长,已经成为一种重要的资源。如何开发、利用这些资源已成为一项推动社会进步、促进科技发展的重要课题。而数据库系统是到目前为止利用计算机技术进行信息管理和数据处理的最重要的工具与手段。数据库技术已成为计算机应用领域内最重要的一个方面。因此对于高等院校工科专业的学生及广大工程技术人员来说,掌握及应用数据库技术是必不可少的基本素质之一。本书正是为适应高等院校工科专业普遍开设的数据库技术课程改革、更新及广大工程技术人员学习、掌握、应用数据库技术的需要而编写的,力求从基本理论和实际应用能力的培养两方面为读者提供帮助和支持。

数据库技术是一种新型的信息管理和数据处理技术。在工程领域已成为人们完成工程分析、计算、数表和图表处理、实现生产组织、管理自动化、科学化的重要工具。因此数据库技术已成为高等院校工科专业一门重要的技术基础课。但是,由于数据库技术发展十分迅速,内容更新很快。为了适应这种形势的需要,必须对本课程从教学方式到教学内容都进行大力改革,教材建设就是其中重要的内容之一。目前数据库技术课程的教材有的内容陈旧,已不能反映当前发展动态;有的篇幅太大,内容过于繁杂;有的则过于简单。为此,编写一本能反映本课程最新发展动态且内容精练、实用的教材是十分必要的。同时由于关系型数据库管理系统 FoxPro 性能优越,对运行环境要求较低,是目前微机上最流行的、应用最广泛的数据库软件。因此本书介绍、讨论的是 FoxPro 2.6 For Windows。

本书的编写以淡化专业、拓宽应用基础为总的思路,力争反映数据库技术最新发展动态,着重培养读者解决实际问题和创新的能力。在编写时除全面介绍 FoxPro 系统的基本概念及基本操作以外,重点放在实用程序设计及应用系统开发方面。通过精心组织典型工程应用例题让读者掌握数据库技术与工程实际相结合的方法与技巧,并与后续的 CAD 技术等课程的内容有机地结合起来,提高读者综合运用计算机技术的能力。同时本书行文简洁、流畅,在内容的安排上由浅入深、循序渐进、图文并茂,各章配有精心组织的习题,附录给出了 FoxPro 系统的所有命令及函数的索引以供读者查阅,因此十分便于教学与自学。本书可作为高等院校工科专业数据库技术课程的教材或教学参考书。

本书第一章、第二章、第三章、第八章由宗子安编写;第四章、第六章由湛霖霖编写;第五章、第七章由许第洪编写;第九章由宗子安、许第洪编写;附录由湛霖霖编写。由宗子安负责全书的策划与统稿。由于编者水平有限,书中难免有不当之处,敬请读者批评指正。

本书在编写过程中参阅了有关文献,在此向这些文献的作者表示衷心的感谢。

编 著 者

2000 年 3 月于岳麓山

目 次

第一章 概 论

1.1 信息、数据及数据处理	(1)
1.2 数据处理技术的发展	(1)
1.2.1 数据管理技术的发展阶段	(1)
1.2.2 数据库系统处理数据的特点	(2)
1.3 信息结构及数据描述	(3)
1.3.1 信息结构的三个世界及其相互关系	(3)
1.3.2 信息世界与数据世界中的基本概念	(3)
1.3.3 实体之间的相互联系	(5)
1.4 数据模型	(6)
1.4.1 层次模型	(6)
1.4.2 网状模型	(6)
1.4.3 关系模型	(7)
1.4.4 关系代数	(9)
1.5 数据库系统基础知识	(11)
1.5.1 数据库系统	(11)
1.5.2 数据库的分级结构	(12)
1.5.3 数据库管理系统(DBMS)	(12)
1.6 FoxPro 概述	(13)
1.6.1 FoxPro For Windows 简介	(13)
1.6.2 FoxPro 基本知识	(19)

第二章 数据库的建立与基本操作

2.1 数据库的建立	(26)
2.1.1 定义数据库的结构	(26)
2.1.2 建立数据库	(27)
2.2 数据库的基本操作	(30)
2.2.1 数据库的打开与关闭	(30)
2.2.2 记录数据的添加与插入	(31)
2.2.3 数据库结构与记录的显示	(33)
2.2.4 数据库结构与记录的编辑	(35)
2.2.5 数据库结构与数据库文件的复制	(43)
2.3 数据库的辅助操作	(47)
2.3.1 有关内存变量操作的命令	(47)

2.3.2 有关文件操作的命令	(50)
2.3.3 其他辅助命令	(54)

第三章 数据库的使用

3.1 数据库的重新组织	(57)
3.1.1 数据库的分类	(57)
3.1.2 数据库的索引	(58)
3.2 数据库的查询	(64)
3.2.1 顺序查询	(64)
3.2.2 快速查询	(66)
3.3 数据库的统计与汇总	(69)
3.3.1 统计记录个数的命令	(69)
3.3.2 求和命令	(69)
3.3.3 求平均值的命令	(70)
3.3.4 综合统计命令	(70)
3.3.5 分类求和命令	(71)
3.4 多重数据库操作	(72)
3.4.1 工作区与工作区的选择	(73)
3.4.2 数据库文件之间建立关联	(74)
3.4.3 数据库文件之间的连接	(79)
3.4.4 数据库文件之间的更新	(80)

第四章 函 数

4.1 数值函数	(83)
4.2 字符函数	(85)
4.3 日期与时间函数	(89)
4.4 转换函数	(91)
4.5 数据库函数	(93)
4.6 测试函数	(94)
4.7 窗口函数	(96)
4.8 菜单及系统函数	(98)

第五章 基本程序设计

5.1 FoxPro 程序的建立与执行	(100)
5.1.1 FoxPro 程序设计的特点	(100)
5.1.2 命令文件的建立和编辑	(102)
5.1.3 命令文件的执行	(103)
5.2 常用程序设计命令	(103)
5.2.1 输入、输出命令	(103)
5.2.2 交互输入命令	(105)

5.2.3	数据的格式化输出	(107)
5.2.4	数据的格式化输入	(108)
5.2.5	宏代换和名表达式	(110)
5.2.6	程序终止命令	(111)
5.2.7	预处理命令	(111)
5.3	程序流程的控制命令	(111)
5.3.1	分支选择命令	(112)
5.3.2	多分支选择命令	(115)
5.3.3	DO WHILE...ENDDO 循环命令	(117)
5.3.4	FOR...ENDFOR 循环命令	(120)
5.3.5	SCAN...ENDSCAN 循环(扫描循环)	(122)
5.4	菜单驱动	(123)
5.5	数组及应用	(125)
5.5.1	数组的定义与赋值	(126)
5.5.2	数组与数据库文件之间单记录数据交换	(127)
5.5.3	数据库文件与数组之间的多记录数据交换	(129)
5.6	子程序、过程和自定义函数	(131)
5.6.1	子程序	(132)
5.6.2	过程	(133)
5.6.3	自定义函数	(135)
5.6.4	全局变量与局部变量	(136)
5.7	综合举例	(138)
5.7.1	数据分析	(138)
5.7.2	功能分析	(139)
5.7.3	程序示例	(139)

第六章 创建 FoxPro 应用程序

6.1	窗口设计	(146)
6.1.1	程序方式制作窗口	(146)
6.1.2	用屏幕生成器设计窗口	(158)
6.2	菜单设计	(164)
6.2.1	概述	(164)
6.2.2	程序方式制作用户菜单	(165)
6.2.3	用菜单生成器设计用户菜单	(171)
6.3	报表设计	(175)
6.3.1	程序方式制作报表	(175)
6.3.2	用报表生成器设计报表	(177)
6.4	项目管理器	(182)
6.4.1	项目管理器简介	(182)
6.4.2	项目管理器的使用	(182)

第七章 结构化查询语言(SQL)与关系范例查询(RQBE)

7.1 结构化查询语言(SQL)的命令	(185)
7.1.1 CREATE CURSOR 命令	(185)
7.1.2 CREATE TABLE 命令	(186)
7.1.3 INSERT 命令	(187)
7.1.4 SELECT 命令	(187)
7.1.5 合并查询结果	(191)
7.2 关系范例查询(RQBE)	(191)
7.2.1 打开 RQBE 窗口	(192)
7.2.2 RQBE 窗口的组成	(193)

第八章 FoxPro 系统参数配置及配置文件

8.1 系统参数配置命令	(194)
8.1.1 系统参数配置命令的两种基本格式	(194)
8.1.2 常用系统参数配置命令	(194)
8.2 通过 View 窗口进行系统参数配置	(200)
8.3 系统配置文件	(201)

第九章 数据库技术在工程中的应用

9.1 工程数据库	(203)
9.1.1 概述	(203)
9.1.2 历史回顾及发展动态	(203)
9.1.3 国内外工程数据库技术应用简介	(205)
9.2 数据库技术在 CAD 中的应用	(206)
9.2.1 数据库技术在 CAD 中的应用方式	(206)
9.2.2 工程数表的处理	(207)
9.3 FoxPro For Windows 应用系统的开发方法与技巧	(209)
9.3.1 凯斯达 CAPP 系统分析	(209)
9.3.2 凯斯达 CAPP 系统界面的设计	(216)
9.3.3 凯斯达 CAPP 系统编制过程	(220)

第一章 概 论

1.1 信息、数据及数据处理

随着人类社会的不断进步,科学技术的飞速发展,信息量急剧增加,对载荷信息的各种数据的处理量也日趋增加,成为人们十分关注的课题。在现代社会中,信息与数据已成为一种重要的资源,如何利用、开发这种资源,已成为一种推动社会进步、促进科技发展的必不可少的工作。在现实生活中信息与数据这两个概念往往很难区分,使用时也常易混淆不清。为此,我们可以给它们下一个简单的定义:信息伴随物质而存在并随物质的变化而变化,是反映客观事物的可通讯的知识。简而言之,物质是信息的基础,而信息是物质的反映。数据则是用以载荷信息的数字、字母或符号,是信息的具体表示形式。由此可见,信息与数据这两个概念是有区别的。

可以根据使用的要求把信息分为不同类别,如按信息的反映形式分,可分为数字信息、图像信息和声音信息等。另外也可以按应用领域分为社会信息、科技信息、经济信息等。计算机技术的飞速发展,尤其是其高速处理数据的能力和存储容量巨大的特点,使得人们有可能对大量的信息进行各种有效处理。所谓信息处理,实质上就是数据处理,包括对各种数据的收集、记载、分类、整理、存储、排序、计算、加工、传输等。而数据库系统则是到目前为止利用计算机技术进行数据处理最重要的工具与手段。数据库技术已成为计算机应用领域中最重要的一个方面。

1.2 数据处理技术的发展

1.2.1 数据处理技术的发展阶段

数据处理技术与计算机技术是密不可分的,且随计算机技术的发展而不断发展,大体上讲已经历了三个发展阶段。

一、人工管理阶段

这是数据处理的初期阶段,这一时期的主要特点在于数据与程序相互依存,即在解决某一问题时,把数据处理纳入程序设计的过程。在程序中除设计求解方式以外还要设计数据的逻辑结构、物理结构、存取方式、输入输出方式等,也就是说数据与程序不具有相互的独立性,一组数据只对应一个特定的程序。由于数据是面向应用的,因此无需软件对数据进行管理。一般也不长期保存数据,基本上没有文件的概念。

二、文件系统管理阶段

这一阶段计算机的操作系统已提供了文件系统来实现对数据的管理。文件系统由文件和文件管理系统组成,其中文件是把数据按一定的规则组织起来的数据组合体,它以记录为基本单位,而记录又由若干相关数据项组成。因此也可以说文件是若干具有相同性质的记录的集合。而文件管理系统则负责存取、管理文件。文件系统利用磁盘、磁带等大容量存储器来存储文件。由于存储介质的性质不同,文件又可分为顺序文件、索引文件、链接文件、倒排文件等。

用文件管理系统对数据进行管理是数据处理的一大进步,我们可以通过文件对数据进行反复处理。如进行查询和各种编辑操作,例如删除、插入、修改等,同时文件管理系统作为应用程序与数据文件之间的接口,使得应用程序与数据文件之间保持了一定的独立性,因此在程序设计时一般较少考虑数据的物理结构,包括存取结构和存取方法等。但是利用文件管理数据仍然有很多缺点,即数据文件基本上还是对应于一个或几个应用程序,仍然是一个不具有弹性的、无结构的信息集合。数据的冗余度大、不易扩充,同时也难反映客观事物之间存在的内在联系。

三、数据库系统管理阶段

数据库系统代表的是一种新型的数据管理技术,它是在文件系统管理数据的基础上发展起来的。数据库是以一定的组织方式存储在一起的相关数据的集合。它能以最佳方式、最小的数据冗余为多种应用程序服务,数据独立于应用程序而存在,可为各类用户共享使用。数据库技术产生于 20 世纪 60 年代中期,随计算机技术的发展而发展,目前已在社会生活的各个领域得到广泛使用,成为当今最重要也是最先进的数据处理的工具。

1.2.2 数据库系统处理数据的特点

如前所述,数据库系统是当今最重要也是最先进的数据处理工具,它克服了文件系统管理数据的弊端,具有鲜明的特点,具体来说有:

一、数据的独立性

即数据独立于应用程序。当数据的存储方式和数据的逻辑结构改变时,不需要用户改变用户程序。数据的独立性包括逻辑独立性和物理独立性。独立性的实现是数据库系统通过改变映射关系而完成的。

二、数据的安全性

即数据的保护,可防止不合法地使用数据,也就是防止无关人员获取他不应知道的数据。通常采用密码、口令或其他手段鉴别用户身份或存取权限,限制各类用户对文件的自由存取,从而提高数据的安全性。

三、数据的完整性

即数据的正确性、真实性、客观性。数据库系统应有检验措施保证数据的完整性,也就是要保证数据之间的内在联系。硬件故障、软件故障或人为的误操作常常导致数据的失真,从而破坏了数据的完整性。我们应尽量避免这种破坏,同时一旦发生破坏数据完整性的情况,应有尽快恢复正常状态的能力。

四、数据的共享性

即同一数据文件可以为多个应用程序提供服务,同时不同的使用者可以用不同的方式调用数据。要实现数据的共享,首先要实现和保证数据的独立性。

五、控制数据冗余

即存储在数据库中的数据应尽可能地减少重复,若数据冗余度较大则不仅多占用存储空间,而且增加搜寻、查询数据的时间,同时容易导致数据的不一致性,如修改数据时可能忽略对重复数据的修改等。

六、并发控制

在实现数据共享时,有可能出现多个用户同时存取同一数据的情况,如果其中包含修改操作就可能破坏数据的完整性或一致性。我们把这些同时发生的操作称为并发操作,所谓并发控制就是对并发操作进行控制的策略,避免相互之间的干扰。

七、数据的一致性

即描述同一客观事物的数据不论在何时、何处应保持一致。而数据的不一致性指数据的矛盾性或不相容性。数据的不一致性一般是由于数据的冗余、并发控制不当、各种故障、误操作而造成的。数据库系统采取了各种措施以维护数据的一致性。

八、数据的集中管理

数据库的基本思想就是对所有数据进行集中管理。统一、全面地控制数据的结构、存储方式、操作方式、输入输出方式等。正是由于施行了集中管理才可能达到保持数据独立性、数据共享及最小冗余度的目标。

1.3 信息结构及数据描述

1.3.1 信息结构的三个世界及其相互关系

在信息结构中,从客观事物出发,通过去伪存真、去粗取精的过程将数据存入数据库,并通过控制决策机构又反作用于客观事物。信息的这一循环过程经历了与数据处理有关的三个领域,即现实世界、信息世界和数据世界。其中现实世界是存在于人们头脑之外的客观世界,由客观事物及其相互联系组成;而信息世界是现实世界在人们头脑中的反映,人们从观察客观事物及其相互联系中得到大量的数据,对这些数据进行记录、整理、分类,即去伪存真、去粗取精(也称为数据规范),获得大批信息,就此进入信息世界;对这些信息的进一步加工处理就进入了数据世界,数据世界实质上就是信息世界中的信息数据化,对现实世界中的客观事物及其相互关系用数据加以描述。这三个世界之间的相互关系如图 1.1 所示。

1.3.2 信息世界与数据世界中的基本概念

在信息世界与数据世界中对数据的描述是不同的,下面将分别对信息世界与数据世界中的一些基本概念进行介绍:

一、信息世界中的一些基本概念

1. 实体(Entity)

客观事物在信息世界中称为实体,可以是具体的事物,如一本书、一个人,也可以是抽象的

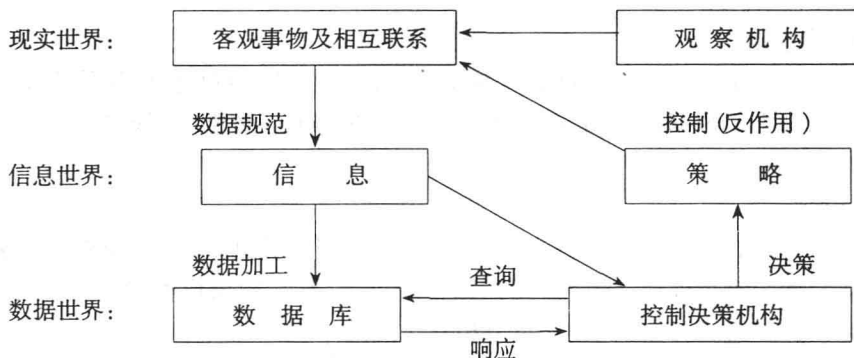


图 1.1 信息结构的三个世界之间的相互关系

事物,如事物之间的联系。

2. 属性 (Attribute)

属性用以描述实体在某一方面的特征。由于每一实体一般具有多种特征,因此实体有各种不同的属性。如学生这一实体的属性有学号、姓名、性别等。

3. 实体型 (Entity Type)

由某些属性组成的集合,它表征了某一类实体的类型,故称为实体型。如教科书这一实体型可用书名、书号、作者、出版社、定价等属性组合加以描述。

4. 实体集 (Entity Set)

相同类型的实体的集合称为实体集。如一批教科书、一班学生等,它们都具有相同的实体型。

5. 实体模型 (Entity Model)

反映实体之间联系的模型称为实体模型。如教学情况这一实体模型就反映了学生、教师、课程等实体之间的相互联系。实体模型为一种静态模型,即它只反映当前实体之间的状态,不能反映实体状态发生变化的过程。建立实体模型最关键的一点是必须正确反映客观事物,即实体之间的联系。

二、数据世界中的一些基本概念

1. 记录 (Record)

对应于每一个实体的一组数据。如反映教科书这一实体的一组数据为,书名:计算机辅助设计原理与应用技术,书号:ISBN 7-81053-103-4/TP·11,作者:刘子建等,出版社:湖南大学出版社,定价:30 元,这就是一条记录。

2. 数据项 (Field)

对应于信息世界中的属性,也称为字段。实体的每一属性均对应于一个数据项,如描述教科书的书名、书号、作者、出版社、定价等都是描述教科书的数据项。

3. 记录型 (Record Type)

对应于信息世界中的实体型,由一组数据项表征记录的类型。

4. 文件 (File)

对应于信息世界中的一个实体集,为相同类型的记录的集合。

5. 数据模型 (Data Model)

对应于信息世界中的实体模型,为实体模型的数据化,是数据库设计的核心问题。

6. 关键字 (Key Word)

能惟一标识记录的一个或多个数据项称为关键字。对一条记录而言,可以有不止一个关键字。通常我们从中选一个主关键字,其余的则称为候选关键字。如前述教科书这一实体集中可以用书号这一数据项作为关键字来惟一标识各种不同的教科书。

1.3.3 实体之间的相互联系

现实世界中的客观事物都不是孤立存在的,它们之间存在各种联系。相应地在信息世界中实体间也存在两种联系:一种是实体内部的联系,即记录内部字段之间的联系;另一种是实体与实体之间的联系,记录与记录之间的联系。数据库系统除了考虑记录内部的联系之外,还要考虑记录之间的联系,也就是文件之间的联系。这种联系是比较复杂的,但经过分析、抽象化以后,可以归结为三类。

一、1-1 关系(One to One)

如果实体集 E1 中的每一个实体至多对应于实体集 E2 中的一个实体,反之实体集 E2 中每一实体至多对应于实体集 E1 中的一个实体,则称实体集 E1、E2 的关系为 1-1 关系。例如独生子女与父母的联系就是一种 1-1 关系,这是最简单的实体联系。应注意的是 1-1 关系不一定都是一一对应关系。

二、1-m 关系(One to Many)

如果实体集 E1 中至少有一个实体对应于实体集 E2 中一个以上的实体,反之实体集 E2 中任一实体至多对应于实体集 E1 中的一个实体,则称实体集 E1 对 E2 的关系是 1-m 关系。例如公司与职员的关系就是 1-m 关系,因为一个公司有多个职员,而一个职员只能服务于一个公司。1-m 关系为一种比较普通的关系。

三、m-m 关系(Many to Many)

如果实体集 E1 中至少有一个实体对应于实体集 E2 中一个以上的实体,反之实体集 E2 中也至少有一个实体对应于实体集 E1 中一个以上的实体,则称实体集 E1 对 E2 为 m-m 的关系。例如学生与老师之间的关系就是 m-m 关系,因为一个学生选多个老师的课,而一个老师教多个学生。要说明的是 1-1 关系实际上是 1-m 关系的特例,1-m 关系又是 m-m 关系的特例。它们之间为一种包含关系。图 1.2 为用图形表示实体之间的三种联系。

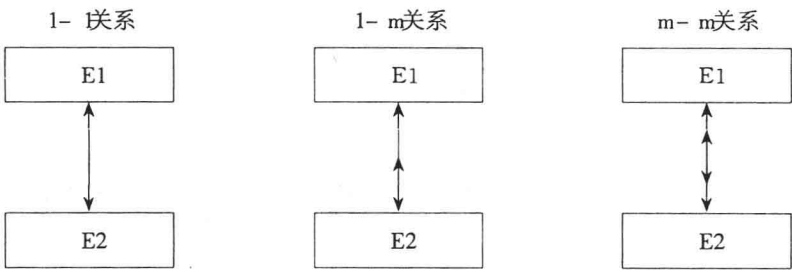


图 1.2 实体关系的图形表示

1.4 数据模型

如前所叙,数据模型为实体模型的数据化,也就是对实体之间相互联系的一种数据描述。是用于表示现实世界的一种方法与工具,数据模型的建立是数据库设计的核心问题。数据模型的不同将决定相应不同的数据库管理系统 DBMS。目前最著名的数据模型有层次模型、网状模型及关系模型,即三大经典数据模型,其中关系模型应用最为广泛。同时,随着计算机多媒体技术的飞速发展,存储和处理图像、声音等多媒体对象的要求日益强烈。自 20 世纪 80 年代中期以来,对新一代数据库系统的研究也日益活跃,并推出了对象-关系型数据库系统和面向对象型数据库系统。这些新一代的数据库系统目前虽然尚不成熟,但已显示了光明的发展前景。本书只介绍三大经典数据模型。

1.4.1 层次模型

层次模型可以用一个倒立的树形结构表示,树中的结点表示实体集,连线表示两实体间的联系,这种联系只可能是 1-1 关系或 1-m 关系。另外层次模型满足下列两个限制:

1. 树形结构有且仅有一个结点无父结点。此结点就是树的最高层的结点,也叫根结点。
2. 除了根结点以外的其余结点都有且只有一个父结点。

在现实世界中,有许多实体之间的联系就是这种层次关系,如家族关系、学校关系、企事业单位管理机构等。图 1.3 给出了一个层次模型的示例,其中 S 表示根结点,其他各层 S_i 与 S_{ij} 结点为父子结点关系,均为 1-1 关系或 1-m 关系。要说明的是 m-m 关系不能直接用层次模型表示,一般应先设法将该 m-m 关系分解为两个 1-m 关系,然后再用层次模型表示。层次模型的树形结构限定了层次模型具有层次性、定向性及有序性。从根结点开始自顶向下、从左到右依次给链接的各结点一个类型码,以表示各结点在层次结构中的位置,如图 1.3 所示。层次模型的数据库管理系统 DBMS 是最早出现的,其典型代表作为 IMS (Information Management System),它于 1968 年问世,是世界上第一个 DBMS。

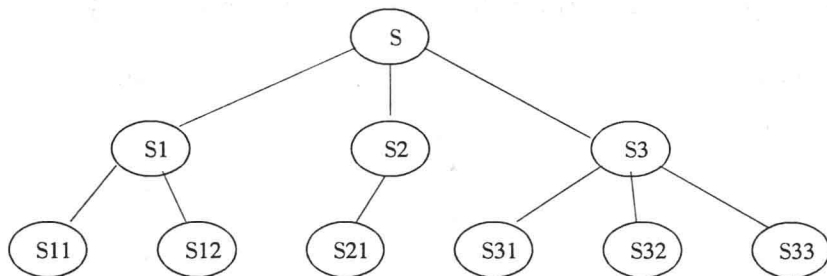


图 1.3 层次模型的示例

1.4.2 网状模型

如果取消层次模型中的两个限制条件,即允许有零个或多个结点无父结点,同时允许每一

个结点有多个父结点,且两个结点之间可以有二种或二种以上联系的话,便形成了网,又叫丛结构。用丛结构表示实体及其联系的模型叫网状模型。在理论上它可以直接处理 $m - m$ 关系,但很多情况下对这一点作了限制,只允许处理 $1 - m$ 关系。图 1.4 为网状模型示例。显然层次模型只是网状模型的特例,两者在本质上是一样的,均用结点表示实体集,用连线表示实体间的联系。差别在于网状模型中的连线更加纵横交错,总体结构更为复杂。美国 CODASYL (Conference on System Language) 下属机构于 1969 年所开发的 DBTG 系统就是网状模型的数据库系统的典型代表。

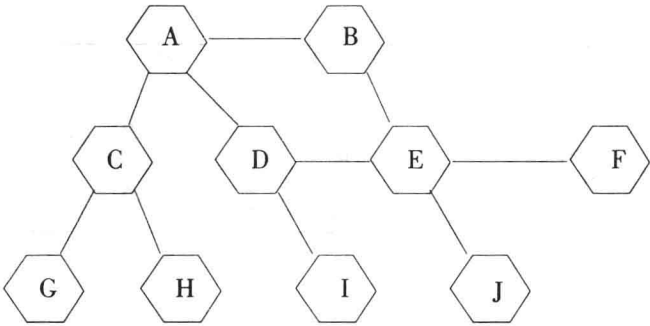


图 1.4 网状模型示例

1.4.3 关系模型

关系模型把实体及其联系归结为满足一定条件的二维表格加以描述,每一关系为一个二维表,相当于一个文件。这种用表格数据来表示实体及其联系的模型叫关系模型。将一组数据列成一个二维表,二维表中每一列对应于实体的一个属性,相当于一个字段,其中给出相应各实体的属性值,每一行形成一个由多种属性组成的多元组与某一特定实体对应,相当于一个记录值。如一个二维表有 m 行、 n 列。则称此关系为 n 元组,且有 m 个 n 元组。在关系模型中要求关系(二维表)满足以下条件:

- 1. 关系中每一列属性必须是不可再分的数据项,并没有重复组。
- 2. 关系中各列指定相异的名字,但列的次序可以任意排列。
- 3. 关系中各行相异,不允许重复,行的顺序也无关紧要。

现以高校教学实体之间的联系为例,构造其关系模型。如图 1.5 所示:

学生关系记录型:

学号	姓名	年龄	性别	专业
----	----	----	----	----

教师关系记录型:

姓名	年龄	职称	系别
----	----	----	----

课程关系记录型:

课程号	课程名	学时数
-----	-----	-----

学习关系记录型:

学号	课程号	分数
----	-----	----

任课关系记录型:

课程号	任课教师姓名
-----	--------

图 1.5 高校教学实体之间的联系关系模型示例

以下为根据上面所构造的关系模型而建立的二维表:

学生关系表:

学 号	姓 名	年 龄	性 别	专 业
98001	张 明	19	男	机电
98002	刘 冬	21	男	计算机
98003	李江山	18	男	土木
98004	陈彬彬	20	女	建筑
98005	周 亮	19	女	环保

教师关系表:

姓 名	年 龄	职 称	系 别
左 建	35	讲师	机械系
刘端文	42	副教授	土木系
李乐慧	28	讲师	计算机系
赵 同	45	教 授	环境工程系

课程关系表:

课程号	课程名	学 时
K001	机械原理	60
K002	数据结构	60
K003	工程力学	50
K004	供暖通风	50

任课关系表:

课程号	任课教师姓名
K001	左 建
K002	李乐慧
K003	刘端文
K004	赵 同

学习关系表:

学 号	课程号	分 数
98001	K001	87
98002	K002	91
98003	K003	94
98004	K004	88

从以上实例可以看出,在关系模型中,其文件中存放了两类数据:实体本身的数据、实体之间的联系。其中实体之间的联系是通过存放两实体的关键字来实现的。如要知道每一个学生所选课程,可通过学号这一关键字在学生关系表与学习关系表之间建立联系。同样,要知道哪一个教师讲授哪一门课,则可通过教师关系表中的关键字姓名与任课关系表中的关键字任课教师姓名来建立联系。

由此可见关系模型与层次模型和网状模型在本质上是不同的,关系模型是通过表格数据

而不是通过指针链接来表示和实现两实体之间的联系的。

关系模型是三种经典数据模型中最晚发展的一种,但它发展最快、应用最为广泛,这主要是由于关系模型最为简单,一个关系就是一个二维数据表格,同时也在于关系模型具有一种富有特色的数据语言,这种语言灵活方便、表达能力和功能均很强,其数学基础为集合运算和谓词运算,并创造了投影、选择和联接运算,统称为关系代数。通过关系代数的各种运算可以对二维表进行任意的分割和组装,随机地构造出各种用户所需要的表格即新的关系。

1.4.4 关系代数

关系代数的运算可分为两大类:一类是通常的集合运算,包括并、差、交和笛卡尔积等;另一类是特殊的关系运算,包括投影、选择、联接等。

一、并、差、交运算

设 R 和 S 为同类关系,即 R 和 S 包含相同个数的属性且属性名相同。R 和 S 如表 1.1 所示。

表 1.1 关系 R 和 S

关系 R:			关系 S:		
A	B	C	A	B	C
a1	b1	c1	a1	b2	c2
a1	b2	c2	a2	b2	c2
a2	b2	c1	a2	b2	c1

1. 并(Union)

R 和 S 的并是由属于 R 或 S 或同时属于 R 和 S 的元组所组成的集合,记为 $R \cup S$,如表 1.2 所示。

2. 差(Difference)

R 和 S 的差是由属于 R 而不属于 S 的所有元组所组成的集合,记为 $R - S$ 。如表 1.3 所示。

3. 交(Intersection)

R 和 S 的交是由同时属于 R 和 S 的所有元组所组成的集合,记为 $R \cap S$,如表 1.4 所示。

交运算也可以用差运算表示即 $R \cap S = R - (R - S)$,要指出的是并、差、交这三种运算的结果与 R 和 S 是同类关系。

表 1.2 关系 $R \cup S$

A	B	C
a1	b1	c1
a1	b2	c2
a2	b2	c1
a2	b2	c2

表 1.3 关系 $R - S$

A	B	C
a1	b1	c1

表 1.4 关系 $R \cap S$

A	B	C
a1	b2	c2
a2	b2	c1

二、笛卡尔积(Cartesian Production)

设 R 为 K_1 元关系,S 为 K_2 元关系,则 R 和 S 的笛卡尔积是一个 $(K_1 + K_2)$ 元组的集合,其中元组的前 K_1 个分量是 R 的一个元组,而后 K_2 个分量是 S 的一个元组,记为 $R \times S$ 。 $R \times S$ 是所有具备这种条件的元组的集合。若 R 有 N 个元组,S 有 M 个元组,则 $R \times S$ 共有: $M \times N$ 个元组。