

RDB MS
RDB MS
RDB MS

中西文

FoxBASE PLUS

原理及其应用

袁淑君 编著

中国科学技术出版社

RDB MS 中西文 FoxBASE PLUS 原理及其应用

袁淑君 编著

中国科学技术出版社

内 容 提 要

FoxBASE+ PLUS 是 80 年代最广泛应用的先进的关系数据库管理系统。本书描述了该语言的定义、概念和使用规则;系统介绍了数据库文件的建立方法及其使用的命令和操作;描述了数据库的各种操作及其命令;以结构化程序设计方法描述了该系统的基本控制结构以及过程的概念和使用。书中列举了大量的实例,各章均配有丰富的复习思考题和上机练习题,以利于各类非计算机专业的读者学习与理解。

本书宜作为大学、大专、中专文理工农医各科及成人教育中非计算机专业的微机应用教材,也可作为各行业、部门从事信息系统办公室自动化的广大人员的参考书。

(京)新登字 175 号

RDB MS 中西文 FoxBASE PLUS 原理及其应用

袁淑君 编著

责任编辑:谭建新 孙倩

封面设计:王序德

技术设计:范小芳

*

中国科学技术出版社出版(北京海淀区白石桥路 32 号)

新华书店北京发行所发行 各地新华书店经售

北京市燕山联营印刷厂印刷

*

开本:787×1092 毫米 1/16 印张:20.75 字数:500 千字

1992 年 11 月第 1 版 1992 年 11 月第 1 次印刷

印数:1—5 000 册 定价:14.00 元

ISBN 7-5046-0778-9/TP·28

前 言

关系数据库管理系统,特别是在微机上使用的关系数据库管理系统,近年来在我国得到广泛应用,而流行最广、应用面最宽的要算 dBASE III 了。但是 80 年代的后起之秀 FoxBASE+ 以它的速度快(比 dBASE III 快 6~7 倍)、可移植性(可在不同的操作系统环境中运行)、兼容性(与 dBASE III 和 dBASE III PLUS 完全兼容)超越了其他众多的竞争者,引起了国内用户的极大关注。

数据库技术是在 60 年代末期作为数据管理的最新技术登上历史舞台的。自那以后,随着电子计算机应用的扩展和数据处理在计算机应用中所占比重的上升,数据库技术得到了迅速发展,数据库技术和计算机网络通讯已成为当前计算机应用中两个重要的基础领域。电子计算机的一些重要应用,如管理信息系统、办公室自动化技术、计算机辅助设计、智能专家系统等等,大都离不开这两个基本手段。

数据库技术是在传统文件技术的基础上发展起来的,它与传统文件技术的最大区别是具有数据共享性。在数据库中,数据不再属于特定的应用,而是面向整个组织。数据仍然组织成文件形式,但是这些文件之间已不再是松散无关,通常都采用某种数据模型将全部数据文件组织成一个结构化的整体,系统提供对不同数据文件交叉访问的手段,这使消除数据的冗余存储成为可能。数据库系统的其他优点包括:系统提供较强的数据独立性,使数据结构对应用程序的影响大大减少;系统还提供统一的数据操作手段和安全性及完整性控制,使数据的可靠性得到了保证,从而使数据的应用更加有效。

当前流行的数据库管理系统中,采用的数据模型主要有层次模型、网状模型和关系模型三种。80 年代中期以来,关系数据模型逐渐成为占主导地位的数据模型。与其他模型相比,关系模型的优点是模型结构简单(全部数据都组织成关系的形式),逻辑物理界面清晰(从而具有较强的数据独立性),具有较强的集合处理功能。这使数据库应用系统开发的生产率大大提高。此外,由于关系模型具有坚实的理论基础,从而使数据库技术的研究从经验和技巧上升到理论的高度。

本书以 Fox 软件公司最新推出的关系数据库管理系统 FoxBASE+ (V2.10) 为基础,从应用的角度,系统地介绍了 FoxBASE+ 语言及其使用方法。

本书的特点是:它不只是描述 FoxBASE+ 语言的定义、概念和使用规则,而是在简述 FoxBASE+ 的数据结构之后,通过数据库的建立、整理和使用(对数据的各种操作),描述了与之有关的命令及其用法。FoxBASE+ 有很强的编程能力,在描述这些编程用的语句时,我们突出了结构化程序设计方法。学习程序设计语言的目的在于应用,在于解决各种实际问题。为此,我们在第八章专门介绍了应用系统的开发,通过大量例题分门别类地描述了菜单类、输入类、操作类、输出类及其他程序模块的设计方法和技巧。

第四、五、六章是本书的基本部分。第四章系统地介绍了数据库文件的建立方法及其使用的命令和操作。第五章对数据库的各种操作及其命令进行了描述。第六章以结构化程序设计方法描述了 FoxBASE+ 的基本控制结构,以及过程的概念和使用。第七章介绍如何使用系统 FoxBASE+ 的表格输出功能。

本书列有大量例题和习题,适于作大专院校计算机应用课程教材,亦可供工矿企业、商

业、机关、学校等部门从事信息系统、办公室自动化的广大有关人员参考。

本书是作者在近年来讲授 dBASE III 和 FoxBASE+ 及其应用的过程中逐渐编成的,是以 FoxBASE+ 及其应用为基础的。国家教委管理信息中心的李伟同志负责整理了第一、第七和第九章。航空航天部十三所的林詹同志和复旦大学的崔源同志参加了本书的整理工作。北京师范大学计算中心的同志们给予了大力的支持和帮助,在此深表谢意。

作者水平有限,书中难免有许多错误和不当之处,望同行和广大读者批评指正。

编者

1991.8

目 录

第一章 数据库管理系统概述	(1)
1. 1 数据、信息与数据处理	(1)
1. 1. 1 数据与信息	(1)
1. 1. 2 数据处理	(1)
1. 2 数据库	(2)
1. 2. 1 什么是数据库	(2)
1. 2. 2 数据库结构	(3)
1. 3 数据库系统构成	(7)
1. 3. 1 数据库管理系统和数据库系统	(7)
1. 3. 2 数据描述语言和数据操纵语言	(9)
1. 3. 3 数据库管理子程序	(9)
1. 3. 4 数据库系统的典型结构	(10)
习 题	(10)
第二章 FoxBASE+ 概述	(12)
2. 1 关于 FoxBASE+	(12)
2. 2 2. 10 版的特点和益处	(12)
2. 2. 1 FoxCentral—非编程用户接口	(13)
2. 2. 2 FoxView	(13)
2. 2. 3 FoxCode	(13)
2. 2. 4 FoxDoc	(13)
2. 2. 5 FoxGraph	(14)
2. 3 与 dBASE 相比的新功能	(14)
2. 3. 1 内存管理和系统性能的优化	(14)
2. 3. 2 允许同时打开16个文件以上	(14)
2. 3. 3 内存变量个数多达3600个	(14)
2. 3. 4 每个过程文件中可以包括更多个过程	(14)
2. 3. 5 过程文件的装配	(15)
2. 3. 6 命令文件的编译	(15)
2. 4 运行 FoxBASE+	(15)
2. 4. 1 FoxBASE+ 的运行环境	(15)
2. 4. 2 FoxBASE+ 运行过程	(17)
2. 5 FoxBASE+ 的数据结构和文件类型	(19)
2. 5. 1 数据结构	(20)
2. 5. 2 文件类型	(21)
2. 6 FoxBASE+ 的用户界面	(23)
2. 6. 1 本书中使用印刷符的约定	(24)
2. 6. 2 命令结构及其分类	(24)

2. 6. 3 命令的书写格式	(26)
2. 6. 4 函数	(27)
2. 6. 5 全屏幕操作	(27)
2. 6. 6 命令行的输入和编辑	(29)
2. 7 FoxBASE+ 系统容量	(29)
习 题	(31)
第三章 常量、变量、表达式和函数	(32)
3. 1 显示表达式的值——? 和 ??命令	(32)
3. 2 常量	(33)
3. 2. 1 数值常量	(33)
3. 2. 2 字符常量	(33)
3. 2. 3 逻辑常量	(33)
3. 2. 4 日期型常量	(33)
3. 3 变量	(34)
3. 3. 1 变量名	(34)
3. 3. 2 内存变量	(34)
3. 3. 3 定义内存变量数组	(35)
3. 3. 4 内存变量的赋值	(36)
3. 4 表达式	(37)
3. 4. 1 算术运算表达式	(37)
3. 4. 2 关系表达式	(38)
3. 4. 3 逻辑运算表达式	(39)
3. 4. 4 字符串操作	(40)
3. 4. 5 运算符的优先级	(41)
3. 4. 6 检测表达式的类型—TYPE ()	(42)
3. 5 函数	(42)
3. 5. 1 数值运算函数	(42)
3. 5. 2 字符操作函数	(45)
3. 4. 3 日期时间函数	(53)
3. 5. 4 类型转换函数	(55)
3. 5. 5 数据库函数	(58)
3. 5. 6 测试函数	(63)
3. 5. 7 用户自定义函数 (UFD)	(65)
习 题	(65)
第四章 数据库的建立和修改	(67)
4. 1 概述	(67)
4. 2 数据库结构	(67)
4. 2. 1 建立数据库结构 — CREATE 命令	(67)
4. 2. 2 修改数据库结构 — MODIFY STRUCTURE 命令	(76)
4. 2. 3 观察数据库结构 — DISPLAY、LIST 命令	(77)

4. 2. 4 复制数据库文件结构 — COPY STRUCTURE 命令	(79)
4. 3 数据库文件的调用和关闭	(80)
4. 3. 1 调用数据库文件	(80)
4. 3. 2 数据库文件的操作	(81)
4. 3. 3 关闭数据库文件	(81)
4. 4 数据库数据的输入	(82)
4. 4. 1 从键盘向数据库文件输入数据 —CREATE、APPEND、INSERT、BROWSE 命令	(83)
4. 4. 2 从磁盘文件向当前使用的数据库文件添加数据 —APPEND FROM 命令	(88)
4. 5 数据库的显示	(90)
4. 6 数据库文件的编辑和修改	(91)
4. 6. 1 记录的定位—GO 和 SKIP 命令	(91)
4. 6. 2 数据库文件记录的删除与恢复 —DELETE、PACK、ERASE、RECALL、ZAP	(95)
4. 6. 3 数据库文件的删除—ERASE 和 DEIETE FILE	(98)
4. 7 数据库文件记录的编辑和修改 —EDIT、CHANGE、BROWSE 和 REPLACE 命令	(98)
4. 7. 1 EDIT 和 CHANGE 命令	(99)
4. 7. 2 BROWSE 命令	(101)
4. 7. 3 成批修改记录—REPLACE 命令	(104)
4. 8 数据库文件的复制—COPY 命令	(105)
习 题	(107)
第五章 数据库文件的使用	(109)
5. 1 排序与索引	(109)
5. 1. 1 排序—SORT 命令	(109)
5. 1. 2 索引—INDEX 命令	(112)
5. 1. 3 索引文件的打开与关闭—INDEX TO 命令	(115)
5. 1. 4 索引文件的修改—REINDEX 命令	(117)
5. 2 数据查询	(119)
5. 2. 1 直接查找—LOCATE 命令和 CONTINUE 命令	(119)
5. 2. 2 索引查找—FIND 和 SEEK 命令	(121)
5. 2. 3 选择和投影—LIST, DISPLAY 和 COPY 命令	(123)
5. 3 多重数据库文件的操作	(124)
5. 3. 1 选择工作区—SELECT 命令	(125)
5. 3. 2 建立两个数据库之间的联系—SET RELATION 命令	(127)
5. 3. 3 两个数据库的联结—JOIN 命令	(129)
5. 3. 4 数据库记录字段的更新—UPDATE 命令	(132)
5. 4 数据统计	(133)
5. 4. 1 统计数据库文件中记录个数—COUNT 命令	(133)

5. 4. 2 求数据项和—SUM 命令	(133)
5. 4. 3 求平均值—AVERAGE 命令	(134)
5. 4. 4 分类统计—TOTAL 命令	(134)
习 题	(136)
第六章 命令文件	(139)
6. 1 FoxBASE+ 程序	(139)
6. 1. 1 FoxBASE+ 程序结构	(139)
6. 1. 2 FoxBASE+ 程序的书写格式	(140)
6. 2 命令文件的建立与执行	(141)
6. 2. 1 命令文件的建立和修改	(141)
6. 2. 2 命令文件的执行	(144)
6. 3 命令文件中的交互式命令	(145)
6. 3. 1 输入字符串—ACCEPT 命令	(146)
6. 3. 2 输入任何类型数据—INPUT 命令	(147)
6. 3. 3 输入一个字符—WAIT 命令	(147)
6. 3. 4 格式输入输出命令—@	(148)
6. 4 FoxBASE+ 的基本控制结构	(153)
6. 4. 1 顺序结构	(153)
6. 4. 2 选择结构	(153)
6. 4. 3 循环结构	(159)
6. 4. 4 结构嵌套	(166)
6. 5 过程及其调用	(171)
6. 5. 1 过程的基本概念	(171)
6. 5. 2 过程定义—PROCEDURE 命令	(174)
6. 5. 3 过程文件调用和关闭—SET PROCEDURE、CLOSE PROCEDURE 命令	(175)
6. 5. 4 过程引用—DO 命令	(176)
6. 5. 5 变量定义与参数传递	(176)
6. 5. 6 过程的嵌套与递归	(180)
6. 5. 7 过程文件的状配—FOXBIND 程序	(181)
6. 6 编译 FoxBASE+ 程序文件	(184)
习 题	(184)
第七章 数据库数据的输出	(188)
7. 1 输出方向的设置	(188)
7. 2 输出格式的设置	(190)
7. 3 数据输出	(193)
7. 3. 1 单个记录输出	(193)
7. 3. 2 多个记录连续输出	(194)
7. 3. 3 统计输出	(194)
7. 4 报表格式文件	(194)

7. 4. 1 报表格式文件的建立和修改—CREATE REPORT 命令	(195)
7. 4. 2 报表格式文件的修改—MODIFY REPORT 命令	(205)
7. 4. 3 报表的输出—REPORT FORM 命令	(205)
7. 5 标签格式文件	(207)
7. 5. 1 标签文件的建立—CREATE LABEL 命令	(207)
7. 5. 2 标签文件的修改—MODIFY LABEL 命令	(210)
7. 5. 3 标签文件的输出—LABEL 命令	(211)
习 题	(212)
第八章 应用程序设计	(214)
8. 1 菜单类程序设计	(214)
8. 1. 1 菜单设计要求	(215)
8. 1. 2 菜单画面的设计	(215)
8. 1. 3 选择代码的接收	(221)
8. 1. 4 选择的实现	(223)
8. 1. 5 菜单中的容错处理	(225)
8. 1. 6 例	(226)
8. 1. 7 菜单上的选择对象	(229)
8. 2 输入类程序设计	(230)
8. 2. 1 数据库文件结构的建立和修改	(231)
8. 2. 2 数据库数据的装入和修改	(237)
8. 3 处理类程序设计	(243)
8. 4 输出类程序设计	(244)
8. 4. 1 查询输出	(244)
8. 4. 2 报表输出	(248)
8. 5 辅助类程序设计	(254)
习 题	(255)
第九章 数据库的辅助操作	(257)
9. 1 内存变量的观察、保存、清除与恢复	(257)
9. 1. 1 观察内存变量	(257)
9. 1. 2 内存变量的保存	(258)
9. 1. 3 内存变量的清除	(259)
9. 1. 4 内存变量的恢复	(261)
9. 2 FoxBASE+ 环境及外部接口	(262)
9. 2. 1 文件管理	(262)
9. 2. 2 文件转换—COPY TO, APPEND FROM 命令	(265)
9. 2. 3 在 FoxBASE+ 环境中执行 DOS 命令和外部程序—RUN (!) 命令 ...	(267)
9. 2. 4 在 FoxBASE+ 调用汇编语言程序	(267)
9. 2. 5 操作系统与 FoxBASE+ 环境	(268)
习 题	(269)
第十章 FoxBASE+ 系统的配置与调整	(270)

10. 1 系统基本配置文件 CONFIG. SYS	(270)
10. 2 FoxBASE+ 配置文件 CONFIG. FX/DB	(271)
10. 2. 1 FoxBASE+ 系统配置机制	(271)
10. 2. 2 系统配置文件的使用	(272)
10. 2. 3 系统配置命令说明	(273)
10. 2. 4 CONFIG. FX/DB 配置命令一览表	(276)
10. 2. 5 系统配置文件的一个实例	(277)
10. 3 参数控制命令—SET 命令.....	(278)
10. 4 功能键	(282)
习 题	(283)
附录 A 各种版本 FoxBASE+ 命令一览表.....	(284)
附录 B 各种版本 FoxBASE+ 函数一览表	(295)
附录 C 出错报告和出错信息	(301)
附录 D 问题分析图—PAD	(319)
附录 E INKEY 函数的可用键及其值	(321)
参 考 书 目	

第一章 数据库管理系统概述

数据库(Database),数据库管理系统(Data Base Management System),数据库系统(Database System),是数据库技术中常用的三个术语,三者之间有着一定的区别和联系。

1.1 数据、信息与数据处理

1.1.1 数据与信息

数据由一些符号组成,它们可以表达、描述或记录现实事物。但是,数据不同于现实事物。换句话说,一个名字能鉴别一个人,但名字并非等同于该人。数据符号决不可能完整地代表现实事物,它们不完全地描述物体、事件及它们的特性。因此,决定从现实事物提取什么?用什么符号表达它们?应当反应用户对数据的要求,即用户观点。例如,一个组织可以问及雇员过去的经历,将他们的回答编码,以几个数据项存储起来。这是因为人员管理部门把雇员过去的经历视为了解雇员能力的最重要的方面。雇员的另一用户(如雇员的医疗保险公司),可能对雇员在儿童时期患过那些疾病感兴趣,因为他们认为雇员对某种疾病的免疫性很重要。

人们把大量数据装入自己的头脑,但数据项并非孤立的记忆,人们把它们归类,并把它与已经记住的数据项联系起来。同样的概念可以施加于计算机的存储器,即要存储的数据必须被分类,并且与其他已存储的数据联系起来。因此,了解数据项的结构是非常必要的,因为它们与组织及组织内使用数据的人有关。对数据的这种解释称为数据模型(见 1.2.2 节)。

信息在许多场合被认为与数据是同义的,但经常这样区别它们:信息是经过加工(解释),并且对人类社会实践和生产及经营活动产生决策影响的数据。

数据与信息在概念上的区别告诉人们,不是所有的数据都能成为信息,只有经过提炼和浓缩之后,具有新知识的数据才能成为信息。未经加工处理的数据只是一堆死材料,对人类的生活动,对管理者的行动起不了决策作用。数据经过加工处理之后所得的信息仍以数据的形式出现,此时的数据是信息的载体,成为人们认识信息的媒介。

1.1.2 数据处理

数据处理是指对数据进行收集、存储、分类、计算、加工、检索和传输的过程。

自从由计算机实现数据管理以来,数据处理曾经历了人工处理阶段和文件系统阶段。在人工数据处理阶段,数据和程序是一一对应的,程序中包含所需处理的数据,数据不独立于程序,而且有大量的重复。为了克服这些缺陷,出现了用文件系统处理数据的技术。

文件系统是指一种专门管理数据的软件。在这一系统中,按一定的规则将数据组织为一个文件,用户(应用程序)通过文件系统对文件中的数据进行存取。一个文件管理系统示意于图 1-1。

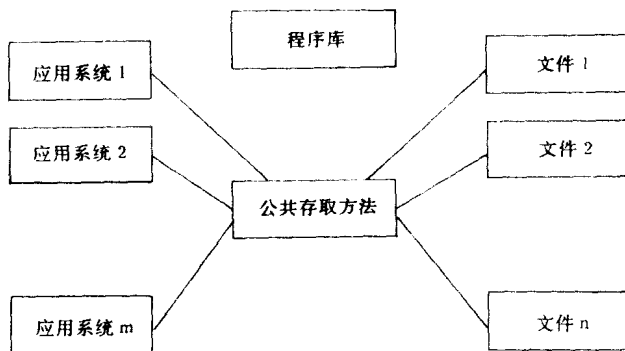


图 1-1 文件管理系统示意

利用文件系统对数据进行管理,实际上是使文件系统成为应用程序和数据之间的接口,这样使应用程序和数据都有了一定的独立性,数据的重复量也有所降低。但是,数据间仍缺乏有机的联系,不同应用程序中会出现许多相同的数据,其冗余度仍然较大,也缺乏充分的独立性,应用程序的编写也较繁琐,这些都使数据的统一控制和管理十分困难。

为了解决文件系统中存在的问题,进入 70 年代后,数据库技术应运而生。

1.2 数 据 库

1.2.1 什么是数据库

顾名思义,数据库是存放数据的“仓库”。直观上说,计算机使用的“仓库”就是磁盘(硬盘或软盘)等外存储媒介。

“数据库”一词的英文是 database 或 data base,base 有基地之意,故 data base 是指供给数据的基地。

然而给数据库下一个确切的定义是很困难的,这是因为:首先数据库是近 20 年来迅速发展起来的计算机软件中的一门新兴学科,它目前正处在从实践向理论过渡的阶段,它的概念、原理和方法正在发展变化,人们对它的认识也有一个历史的发展过程。其次,数据库是一个相当复杂的系统,涉及面很广,很难用几句话严格、简明、准确地概括它的全部特征。鉴于上述原因,现有的数据库定义众说不一。尽管如此,我们还是在下面列出几本书中有关数据库的定义,供读者参考。

《An Introduction to Database System—C. J. DATE》一书中定义:数据库是为某个特定企业的多个应用系统使用的、存储的、操作数据的集合。这里“企业”(Enterprise)只是一个方便的通用的术语,它可以是一个适当的自成一体的商业、科学、技术或其它组织。如制造业公司、银行、医院、学校、政府部门。任何企业都必须保存大量与它的日常业务有关的数据,这就是它的“操作数据”(Operational Data),上述列举的企业的操作数据多半是指产品数据、帐户数据、病人数据、学生数据及教师数据、规划数据等。

《LEARNING dATABASE III PLUS》一书中定义:数据库是相关信息或数据的有组织的集合。我们每天都能碰到几个数据库,例如:通信录、电话录、备忘录等。

《英汉计算机辞典》一书中定义:数据库是在计算机存储设备上合理存放的相互关联的数据的集合。这些数据库具有如下特点:

1. 尽可能不重复(即最小冗余)。
2. 以最优的方式服务于一个或多个应用程序(应用程序共享数据资源)。
3. 数据的存放尽可能独立于使用它的程序(数据的独立性)。
4. 用一个软件统一管理这些数据。例如维护、增加、变更和检索这些数据。

上述说明以数据库组织数据的目的之一是为了数据共享,数据共享的目的是为了减少重复数据,从而节省计算机的存储空间。如一个单位的管理信息系统中,人事部门需要用到有关他的职工的基本情况信息,劳资部门、财务部门同样要用到职工的基本信息,虽然他们各自的应用目的不同,但使用的基本内容是相似的,如都将用到姓名、出生年月、参加工作年月、参加工作时间,基本工资等等。如果每一个部门都建立一个关于职工基本情况的数据库,这将浪费可观的内存,也很难做到数据的一致性。图 1-2 是数据共享示意。

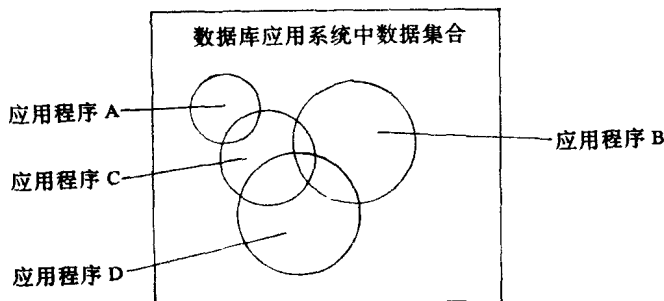


图 1-2 数据共享示意

1.2.2 数据库结构

一. 数据模型

数据库是结构化的数据的集合,数据模型(Data Model)是数据的抽象描述,它详细说明组织数据项及建立数据项之间联系的方法。模型可使用多种方式表达,如图象、数学公式和表格。数据模型的目标是描述数据的本质元素,不涉及其细节。

数据库中的数据结构具有逻辑结构和物理结构两个侧面,因此数据模型有两种主要类型:逻辑数据模型和物理数据模型。面向用户的描述及理解数据的方法称作逻辑数据模型或用户视图(User View);描述数据在外存储器上的存储组织形式称作物理数据模型或物理视图(Physic View)。这两类模型反映了这样一个事实:数据有效的物理存储及检索必须围绕介质及设备的特性进行设计,而数据的用户应能描述及使用数据,却不必关心数据的物理存储。这两种数据模型也反应了如何把使用数据与存储及访问数据理想地分开。理想情况下,存储技术的改变应当能够不影响使用数据的那些应用程序;逻辑数据的改变也应该能够不影响数据的物理存储。这种物理数据模型与逻辑数据模型之间的分离叫数据独立性。

逻辑数据模型(称为模式)帮助用户及设计者详细描述数据需求及数据项之间的联系,逻辑数据模型是概念性的,它反映用户描述现实事物的方法。针对不同的用户和不同的应用又可把模式分为若干子模式,子模式是直接面向用户的,可看作是模式的一个窗口,它们之间的关系示意于图 1-3。用户或分析家负责确定逻辑数据需求,数据库系统的开发人员负

责确定能满足逻辑需求的数据的物理存储

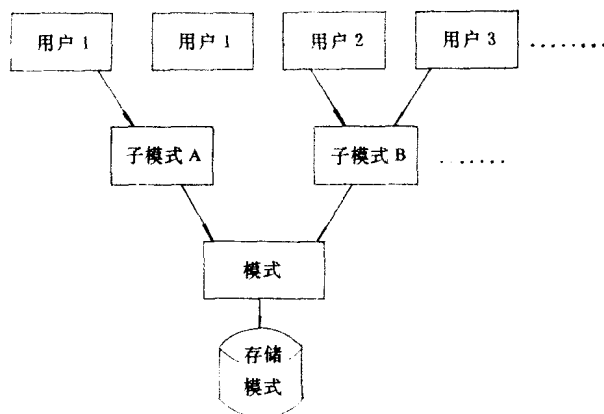


图 1-3 模式示意

从现实世界的事物到数据的逻辑视图,再到数据的物理模型的总体流程示于图 1-4。复杂的现实事物通过思维模型(取决于用户及用户的目标)被条理化、简单化。这种思维多数是不确切的,但是能以某种形式来描述数据需求,这个逻辑的面向用户的需求描述被映射成物理模型或者说与物理存储模型相关联,这个物理存储模型描述数据的存储结构。

数据模型是数据库设计的核心问题。在观念世界中,我们用实体来描述客观事物,实体可以是人、一件物体、一个抽象概念或一事情,它是采集数据的对象。对一个实体有意义的特征称之为属性,如实体“教师”可以有姓名、性别、年龄、民族、专业、职称等属性。它们是客观事物的性质的抽象描述。各属性的取值刻画一个特定的实体,如张某、男、50岁、汉、计算机软件、副教授,这些属性值的集合表征一个具体教师。在机器世界中对应于一个实体的数据称为记录,对应于属性的数据称为数据项或数据项组成的组项。实体内部的联系反映在数据上是记录内数据项之间的联系,实体之间的联系反映在数据上是记录之间的联系。记录有类型与值之分,记录类型是记录的框架,记录值是记录的内容,因而记录之间的联系包括记录类型之间的联系和记录值之间的联系。

在文件系统阶段,从整体上讲,数据是无结构的,即记录与记录间无联系,所以只研究记录内部数据项之间的联系,这种联系通常表现为树形结构。在数据库阶段,还必须研究记录之间的联系。所谓数据模型就是指具有这种联系的数据结构形式。数据模型除数据结构外,还应该包括操作符集合和完整规则集合等方面。这里我们主要讨论数据结构形式。

目前比较流行的数据模型有三种,即按图论理论建立的层次结构模型、网状结构模型以及按照关系理论建立的关系结构模型。它们被分别用于设计层次数据库,网状数据库及关系数据库。下面我们简要介绍这三种模型。

二. 层次模型

数据的层次模型用层次结构或树结构描述实体之间的联系。如一个高等学校的组织机构可以用图 1-5 表示。这种数据结构就象一棵倒置的树,它有如下特点:

1. 有且仅有一个结点无双亲,这个结点即为树的根,称为根结点;
2. 其它结点有且仅有一个双亲。

凡满足上面两个条件的“基本层次联系”集合,就称之为层次模型。在层次模型中同一个



图 1-4 现实、数据模型与数据的物理存储之间的关系

双亲的结点称为兄弟,无从属结点的结点称为叶子。

在层次模型中,一个记录可以有多个从属于它的记录,这些从属记录又有多个记录从属于它们。换句话说,具有特定类型的多个记录同时从属于高一层中另一类型的一个记录,即“双亲”记录可以有多个“子”记录,但一个子记录只能有一个“父”记录。图 1-6 给出了一个层次模型的结构示意图。在图中,每一个框代表一个实体,有向线段代表它们之间的联系。若用 $R_i (i=1,2,3,4,5,6)$ 代表记录, R_1 是祖先,它是 R_2, R_3, R_4 的“双亲”记录, R_2, R_3, R_4 是 R_1 的“子”记录; R_3 是 R_5, R_6 的“双亲”记录, R_5, R_6 是 R_3 的“子”记录。属于同一“双亲”记录的诸记录如 R_2, R_3 与 R_4, R_5 与 R_6 为兄弟。无“子”记录的结点如, R_2, R_4, R_5, R_6 称为叶子。这样 R_i 就构成了一棵倒置的树,这就是一个层次模型。在树结构中,每一个记录只有一个“双亲”记录。因此,对每个记录(根记录除外)只需指出它的“双亲”记录就可以表示出层次模型的整体结构。层次数据库结构是基于层次模型建立的数据库,其典型代表是 IMS (Information Management System)。这是 IBM 公司研制的最早的大型数据库管理系统。

三. 网状模型

在现实世界中,实体之间的联系往往是多重的。如要建立一个课程管理的数据库,多个教师讲授同一门公共课,不同系的学生可能选修同一门公共课。广义地讲,任意一个连通的基本层次联系的集合就是一个网状模型。这种广义的提法把树也包括在网状中。但为了与树区别,我们一般提及网状模型时,要加一些限制,即满足下列的基本层次联系的集合为网状模型:

1. 至少有一个以上结点无双亲;
2. 至少有一个结点有多于一个以上的双亲。

图 1-7 给出一个网状结构模型。网状模型与层次模型的主要区别在于:层次模型中从子女到双亲的联系是唯一的,而网状模型中从子女到双亲的联系不是唯一的。因此,对于网状模型就不能象层次模型那样只用双亲来描述记录间的联系,而是给每一种联系赋予名字,并利用这个名字来查找。

四. 关系模型

关系模型把数据组织成二维表形式。也就是说,它把每个实体集合看成是一张二维表,即一种关系。如图 1-8 所示,每个关系均有一个名称,叫做关系名,这里分别命名为 D, P, DP 。表的行(称为元组)代表实体或记录,每一行代表一个唯一的实体或记录;列代表属性,每一列相当于记录中同类属性的数据项,亦称为域(Field)或字段。表的第一行是各域型的集合,构成一个框架,此即为记录的“型”,其它的行则是各记录的值。关系模型有以下性质:

1. 每一列中的分量(属性或字段)是同类型的数据,取自同一个域;
2. 列的顺序不固定,即列的次序可以任意交换;

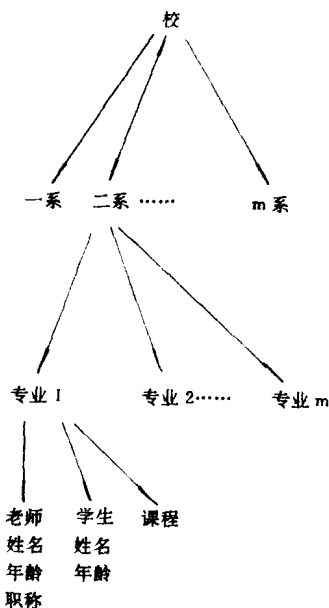


图 1-5 高等学校的组织结构示意图

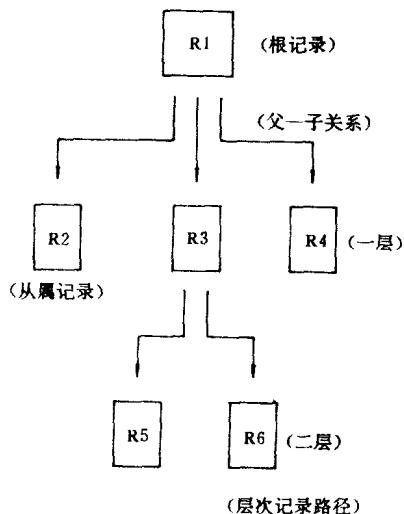


图 1-6 层次结构模型

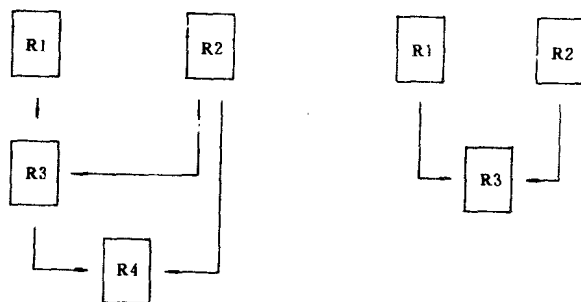


图 1-7 网状模型示意图

3. 任意两个记录不能完全相同；
4. 行的顺序不固定，即行的次序可以任意交换；
5. 每一分量必须是不可分离的数据项。

关系数据库管理系统是建立在关系模型基础上的，可以把它看成是表的集合。一张表是一种关系，相当于关系数据库管理系统中一个数据库文件。关系数据库管理系统的关键特征是实体之间的联系完全用关系(表)中公共域的数据项值来描述。

观察图 1-8 会发现：关系 D 和 DP 有一个公共域(商号)，P 和 DP 有一个公共域(零件号)，D 和 P 有一个公共域(城市)。商号 D3 和零件 P2 在同一个城市是用与它们相关的两个元组的列“城市”有相同值来表示的。这是关系模型独有的特征。事实上，关系数据库中的所有信息：实体的描述，关系(实体之间的联系)的描述，均采用单一的一致的方法，即二维表形式表示。利用近世代数可以给关系下一个确切的定义，关系数据库用数学方法处理数据组