

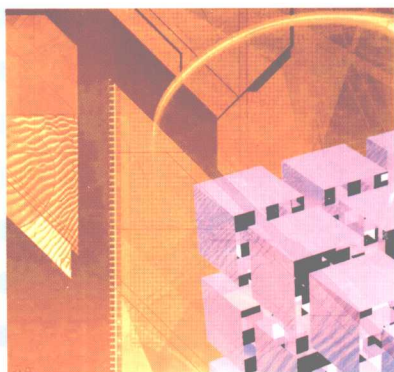


21 世纪高等学校应用型教材

微机数据库 程序设计及应用

——Visual FoxPro (第二版)

□ 沈祥玖 徐 红 李作伟 崔曼宁 黄丽民 编



高等教育出版社
Higher Education Press

TP311.138F0

84

2004

21 世纪高等学校应用型教材

微机数据库程序设计及应用 ——Visual FoxPro

(第二版)

沈祥玖 徐 红 李作伟 崔曼宁 黄丽民 编

88242040-010
8020-810-008
<http://www.hep.edu.cn>
<http://www.hep.com.cn>

2004年11月第1次
2004年11月第2次
2004年11月第1次印刷
25.00元

787×1092 1/16
380 000
380 000

高等教育出版社

18119-00

内 容 提 要

本书针对高等院校教学特点以及全国计算机等级考试二级(Visual FoxPro 数据库语言程序设计部分)考试大纲的要求,以 Visual Foxpro 6.0 中文版数据库系统作为教学数据库,总结作者多年从事教学和实际应用开发的经验,以实际应用案例作为任务驱动,由浅入深,理论结合实际,全面讲述了 Visual Foxpro 关系数据库系统的特点及应用开发技术。

全书共 10 章,主要内容包括:数据库概述、Visual FoxPro 基础知识、创建数据库和数据表、创建查询与视图、表单设计、报表设计、使用命令操作数据库和数据表、面向过程的程序设计、面向对象程序设计以及应用实例——高校学生综合管理信息系统。

本书深入浅出,通俗易懂,可作为高等学校应用型本科、成人高校和高职高专学校相关专业的数据库课程教材,也可作为全国计算机等级考试二级 Visual FoxPro 的考试参考书和各种技术培训教材。本书配套电子教案和实例源代码可从高等教育出版社网站下载,网址为 <http://www.hep-st.com.cn>。

图书在版编目(CIP)数据

微机数据库程序设计及应用:Visual FoxPro/沈祥
玖等编. —2 版. —北京:高等教育出版社,2004.11
ISBN 7-04-015119-7

I. 微... II. 沈... III. 关系数据库—数据库管理
系统, Visual FoxPro IV. TP311.138

中国版本图书馆 CIP 数据核字(2004)第 113753 号

策划编辑 雷顺加 责任编辑 萧 潇 市场策划 韩 飞
封面设计 王凌波 责任印制 陈伟光

出版发行 高等教育出版社
社 址 北京市西城区德外大街 4 号
邮政编码 100011
总 机 010-58581000

购书热线 010-64054588
免费咨询 800-810-0598
网 址 <http://www.hep.edu.cn>
<http://www.hep.com.cn>

经 销 新华书店北京发行所
印 刷 北京民族印刷厂

开 本 787×1092 1/16
印 张 16
字 数 380 000

版 次 1997 年 11 月第 1 版
2004 年 11 月第 2 版
印 次 2004 年 11 月第 1 次印刷
定 价 22.00 元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究
物料号:15119-00

前 言

本书第一版自 1997 年出版以来,多次重印,受到广大读者的肯定和好评。为了适应数据库技术的发展,作者针对高等院校学生的特点以及全国计算机等级考试二级(Visual Foxpro 数据库语言程序设计部分)考试大纲的要求,在继承第一版优点的基础上,对大部分内容进行了修订。

Visual Foxpro 6.0 中文版数据库系统是最流行的和功能强大的桌面数据库管理系统之一,通过直观的可视化的操作即可完成大部分数据的管理工作。Visual Foxpro 是完全面向对象、采用事件驱动机制的关系型数据库系统,内嵌简单易学的程序设计语言和类,使数据库的应用和开发更加便捷、灵活。本书以 Visual Foxpro 6.0 中文版数据库系统作为教学数据库,总结作者多年来从事教学和实际应用开发的经验,以实际应用案例作为任务驱动,理论结合实际,由浅入深,全面讲述了 Visual Foxpro 关系数据库系统的特点及应用开发技术。

本书结构编排以适用于初学者为目的,知识难度有所控制,对于哪些内容可以了解、哪些必须掌握、哪些是较深入的应用等都给出了明确的说明。讲解简明扼要、条理清楚,应用例子贯穿始终、简单易学。

全书共 10 章。第 1、2 章讲述了数据库的基本概念和基本知识。第 3~6 章通过一个典型的进销存数据库应用实例,讲述了在项目管理器中创建表、查询、表单、报表等可视化工具和向导如表向导、查询向导、表单向导、报表向导等,数据库的最终用户利用这些工具和向导不用编程即可构造简单实用的信息系统。这部分充分展示了 Visual Foxpro 数据库易学易用的特点,是数据库应用与开发的入门基础。第 7~10 章讲述数据库程序设计,介绍了使用命令操作数据库和数据表、面向过程和面向对象的程序设计。最后以高校学生综合管理信息系统作为应用实例,全面讲述了菜单和数据库应用系统的开发设计步骤和方法。

本书由沈祥玖、徐红、李作纬、崔曼宁、黄丽民等编写。其中第 1、10 章由沈祥玖编写,第 2、8、9 章由李作纬编写,第 3 章由崔曼宁编写,第 5、6 章由徐红编写,第 4、7 章由黄丽民编写,附录由解永刚编写。最后由沈祥玖统稿。岳国英、刘於勋、尹涛、刘志军、周建玲、宋义发、杨栋老师参与了本书的编写并给予大力帮助,在此一并表示感谢。

由于作者水平有限,书中难免存在不足之处,恳请读者提出批评指正。作者 E-mail: jnjtsxj@163.com。

沈祥玖

2004 年 8 月 10 日于济南

目 录

第1章 数据库概述	(1)
1.1 引言	(1)
1.1.1 数据、数据库、数据库系统和数据库管理系统	(1)
1.1.2 数据管理技术的发展	(2)
1.1.3 数据库技术的研究领域	(2)
1.2 数据模型	(3)
1.2.1 数据模型的三要素	(3)
1.2.2 概念模型	(3)
1.2.3 数据模型	(5)
1.3 数据库系统结构	(8)
1.3.1 数据库系统的模式结构	(8)
1.3.2 数据库系统的体系结构	(8)
1.3.3 数据库管理系统	(9)
1.4 Visual FoxPro 6.0 概述	(11)
1.4.1 Visual FoxPro 6.0 的特性	(11)
1.4.2 Visual FoxPro 6.0 的安装、启动和退出	(11)
1.4.3 Visual FoxPro 6.0 的主窗口和项目管理器	(12)
1.4.4 Visual FoxPro 6.0 的工作方式	(13)
1.4.5 Visual FoxPro 6.0 的可视化交互式设计工具	(14)
习题一	(17)
第2章 Visual FoxPro 基础知识	(19)
2.1 数据类型	(19)
2.2 常量与变量	(21)
2.2.1 常量	(21)
2.2.2 变量	(21)
2.3 表达式	(23)
2.3.1 运算符	(23)
2.3.2 表达式	(25)
2.4 函数	(27)
2.4.1 数值型函数	(27)
2.4.2 字符型函数	(28)
2.4.3 日期和时间函数	(32)

2.4.4 测试函数	(33)
2.4.5 宏替换函数	(35)
习题二	(35)
第3章 创建数据库和数据表	(37)
3.1 项目 managers 的基本操作	(37)
3.1.1 创建项目	(37)
3.1.2 打开项目	(38)
3.1.3 项目 managers 的窗口操作	(38)
3.2 项目 managers 的组成	(38)
3.2.1 项目 managers 的选项卡	(38)
3.2.2 项目 managers 的命令按钮及功能	(40)
3.3 数据库和数据表的建立	(40)
3.3.1 数据库的建立	(40)
3.3.2 数据表的创建	(41)
3.4 数据表设计步骤及举例	(42)
3.4.1 分析问题设计数据表	(42)
3.4.2 数据表结构的操作	(43)
3.4.3 数据表结构的字段属性	(44)
3.5 数据表中记录的操作	(51)
3.5.1 输入和追加记录	(51)
3.5.2 浏览记录	(52)
3.5.3 指针定位	(54)
3.5.4 修改记录	(55)
3.5.5 删除记录	(56)
3.6 数据的索引	(56)
3.6.1 索引的分类	(57)
3.6.2 索引的建立	(57)
3.6.3 索引的使用	(59)
习题三	(60)
第4章 创建查询与视图	(61)
4.1 查询	(61)
4.2 创建查询	(62)
4.2.1 利用查询向导创建查询	(62)
4.2.2 交叉表向导	(65)
4.2.3 利用查询设计器创建查询	(66)
4.2.4 修改查询	(69)

4.3 使用查询	(72)	6.2.2 报表的带区调整	(134)
4.3.1 定向输出查询结果	(72)	6.2.3 报表的网格调整	(135)
4.3.2 查询文件的操作	(73)	6.2.4 报表的控件对象	(135)
4.4 视图	(74)	6.3 输出报表	(140)
4.4.1 视图文件的创建	(75)	6.3.1 用菜单实现输出报表	(140)
4.4.2 定制视图	(79)	6.3.2 用命令实现输出报表	(141)
4.5 结构化查询语言 SQL	(82)	习题六	(142)
4.5.1 SQL 的数据定义	(82)	第7章 使用命令操作数据库和数据表	(143)
4.5.2 SQL 的数据修改	(82)	7.1 数据库、数据表的基本操作	(143)
4.5.3 SQL 的数据查询	(83)	7.1.1 数据库、数据表结构的建立	(143)
4.5.4 简单查询技术	(84)	7.1.2 数据库、数据表的打开和关闭	(144)
4.5.5 高级查询技术	(86)	7.2 数据表结构的浏览与修改	(145)
4.6 多数据表的操作	(89)	7.2.1 浏览表结构	(145)
4.6.1 工作区的编号与别名	(89)	7.2.2 修改表结构	(145)
4.6.2 设置表间的永久关系	(90)	7.3 对数据表记录的操作	(146)
4.6.3 创建、编辑临时关系	(92)	7.3.1 输入和追加数据	(146)
4.6.4 设置参照完整性	(95)	7.3.2 指针定位	(146)
习题四	(96)	7.3.3 浏览、显示和插入数据	(147)
第5章 表单设计	(98)	7.3.4 修改和批量替换数据	(149)
5.1 表单概述	(98)	7.3.5 删除数据	(150)
5.2 用向导生成表单	(99)	7.4 表的索引排序	(152)
5.2.1 用表单向导创建表单	(99)	7.4.1 表的排序	(152)
5.2.2 用一对多表单向导创建表单	(101)	7.4.2 索引	(153)
5.3 用表单设计器加工处理表单	(103)	7.4.3 数据的查找	(156)
5.3.1 打开“表单设计器”窗口	(103)	7.5 数据表的统计	(158)
5.3.2 使用“属性”窗口修改表单	(103)	7.5.1 记录统计命令	(158)
5.3.3 使用表单控件工具在表单上	(105)	7.5.2 累加求和及求平均值	(158)
5.3.3.1 创建控件	(105)	7.6 多表操作	(160)
5.3.4 表单的数据环境	(108)	7.6.1 工作区的选择与使用	(160)
5.3.5 控件的方法	(109)	7.6.2 建立表间的临时关系	(161)
5.3.6 控件的布局	(110)	习题七	(162)
5.3.7 常用控件	(111)	第8章 面向过程的程序设计	(165)
5.4 创建表单应用举例	(120)	8.1 面向过程的程序设计基础	(166)
5.4.1 非数据表表单	(120)	8.1.1 程序的建立、保存和运行	(166)
5.4.2 数据表表单	(121)	8.1.2 常用命令	(169)
习题五	(122)	8.1.3 程序运行控制命令	(170)
第6章 报表设计	(124)	8.1.4 数组	(172)
6.1 创建报表	(124)	8.1.5 输入和输出命令	(175)
6.1.1 利用报表向导创建报表	(125)	8.2 结构化程序设计	(176)
6.1.2 利用快速报表创建报表	(128)	8.2.1 顺序结构程序设计	(177)
6.2 利用报表设计器编辑报表	(131)	8.2.2 分支结构程序设计	(177)
6.2.1 报表设计器的组成和数据环境	(131)	8.2.3 循环结构程序设计	(182)

8.3 子程序及其调用	(189)	9.3.4 面向对象程序设计应用举例	(205)
8.3.1 子程序调用命令	(190)	习题九	(214)
8.3.2 返回主程序语句	(190)	第10章 应用实例——高校学生综合管理	
8.3.3 变量的作用域	(192)	信息系统	(215)
8.4 过程和过程文件	(194)	10.1 系统设计流程	(215)
习题八	(195)	10.2 需求分析及主要功能模块	(216)
第9章 面向对象程序设计	(197)	10.3 数据库规划与设计	(216)
9.1 面向对象程序设计概述	(197)	10.4 输入/输出设计	(218)
9.1.1 面向对象程序设计的基本概念 ..	(197)	10.5 系统定制	(223)
9.1.2 面向对象程序设计的特点	(199)	10.5.1 系统详细设计	(223)
9.2 类的设计	(200)	10.5.2 使用菜单设计器	(224)
9.2.1 类的建立	(200)	10.5.3 使用菜单设计器应用示例	(225)
9.2.2 类的设置	(201)	10.5.4 主程序设计	(226)
9.2.3 类的修改	(202)	10.6 系统生成和发布	(227)
9.2.4 类的使用	(203)	10.6.1 生成应用程序系统	(227)
9.3 面向对象程序设计应用举例	(204)	10.6.2 系统发布	(227)
9.3.1 对象的引用	(204)	习题十	(232)
9.3.2 调用对象的方法程序	(205)	附录 Visual FoxPro 常用命令及函数	(233)
9.3.3 调用对象的事件程序	(205)	参考文献	(246)

第1章

数据库概述

1.1.1 数据库系统 (Database System, DBS)

学习目标

本章主要讲述数据库的基本概念和基本知识。通过本章学习,读者应该掌握以下内容:

- 数据库的基本概念
- 关系数据模型
- Visual FoxPro 6.0 启动与退出
- Visual FoxPro 6.0 的基本操作方法

1.1 引言

数据库技术产生于20世纪60年代中期,是数据管理的最新技术,是计算机科学的重要分支,它的出现极大地促进了计算机应用向各行各业的渗透。本章将介绍数据库的有关概念、数据库的系统结构、数据库管理系统的功能和组成等知识,作为后面各章学习的基础。

1.1.1 数据、数据库、数据库系统和数据库管理系统

数据、数据库、数据库系统和数据库管理系统是4个密切相关的基本概念。

1. 数据(Data)——数据是描述事物的符号记录。文字、图形、图像、声音、学生的档案记录、货物的运输情况等

都是数据。

2. 数据库(DataBase, DB)

所谓数据库是长期储存在计算机内、有组织的、可共享的数据集合。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, DBMS)

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。数据库管理系统使用户能方便地定义数据和操纵数据,并能够保证数据的安全性、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. 数据库系统(DataBase System, DBS)

数据库系统是指在计算机系统中引入数据库后构成的系统,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。

1.1.2 数据管理技术的发展

数据管理是指如何对数据分类、组织、编码、储存、检索和维护,是数据处理的中心问题。数据管理经历了人工管理、文件系统和数据库系统三个阶段。

1. 人工管理阶段

在20世纪50年代中期以前,计算机主要用于科学计算。当时的硬件状况是,外存只有纸带、卡片和磁带,没有磁盘等直接存取的存储设备;软件状况是,没有操作系统,没有管理数据的软件;数据处理方式是批处理。

2. 文件系统阶段

20世纪50年代后期到60年代中期,计算机的应用范围逐渐扩大,计算机不仅用于科学计算,而且还大量用于管理。这时硬件方面已有了磁盘、磁鼓等直接存取存储设备;软件方面,操作系统中已经有了专门的数据管理软件,一般称为文件系统;处理方式上不仅有了文件批处理,而且能够联机实时处理。

3. 数据库系统阶段

20世纪60年代后期以来,计算机用于管理的规模越来越大,应用越来越广泛,数据量急剧增长,同时多种应用、多种语言互相覆盖地共享数据集合的要求越来越强烈。这时硬件已有大容量磁盘,硬件价格下降,软件价格上升,为编制和维护系统软件及应用程序所需的成本相对增加;在处理方式上,联机实时处理要求更多,并开始提出和考虑分布处理。在这种背景下,以文件系统作为数据管理手段已经不能满足应用的需求,于是为解决多用户、多应用共享数据的需求,使数据为尽可能多的应用服务,出现了数据库技术,出现了统一管理数据的专门软件系统——数据库管理系统。

数据库技术从20世纪60年代后期产生到今天仅仅40年的历史,但其发展速度之快,使用范围之广是其他技术所不及的。20世纪60年代末出现了第一代数据库——网状数据库和层次数据库,20世纪70年代出现了第二代数据库——关系数据库。目前关系数据库已成为当今最为流行的商用数据库系统。本书介绍的 Visual FoxPro 就是一个关系数据库管理系统(RDBMS)。

1.1.3 数据库技术的研究领域

当前,数据库研究的范围有以下三个领域。

1. 数据库管理系统软件的研制

数据库管理系统 DBMS 是数据库系统的基础。DBMS 的研制包括研制 DBMS 本身以及以 DBMS 为核心的一组相互联系的软件系统。研制的目标是扩大功能、提高性能和提高用户的生

产率。念翻的科支 2MB2 一某式不又越系机算计的科具于解不题解一式象能时事些些附人

2. 数据库设计

数据库设计的主要任务是在 DBMS 的支持下,按照应用的要求,为某一部门或组织设计一个结构合理、使用方便、效率较高的数据库及其应用系统。其中主要的研究方向包括数据库设计方法、设计工具和设计理论的研究,数据模型和数据建模的研究,计算机辅助数据库设计方法及其软件系统的研究,数据库设计规范和标准的研究等。

3. 数据库理论

数据库理论的研究主要集中于关系的规范化理论、关系数据理论等。近年来,随着人工智能与数据库理论的结合以及并行计算技术的发展,数据库逻辑演绎和知识推理、并行算法等理论研究,以及演绎数据库系统、知识库系统和数据仓库的研制都已成为新的研究方向。

1.2 数据模型

数据模型就是现实世界特征的模拟和抽象。由于计算机不可能直接处理现实世界中的具体事物,所以人们必须事先把具体事物转换成计算机能够处理的数据。在数据库中用数据模型这个工具抽象、表示和处理现实世界中的数据和信息。

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,它是按用户的观点对数据和信息建模。另一类模型是数据模型,主要包括网状模型、层次模型和关系模型等,它是按计算机系统的观点对数据建模。

1.2.1 数据模型的三要素

数据模型由三个要素组成:数据结构、数据操作和完整性约束等。

1. 数据结构

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。数据模型按其数据结构分为层次模型、网状模型和关系模型。

2. 数据操作

数据操作用于描述系统的动态特性,是指对数据库中各种对象的实例允许执行的操作的集合,包括操作及有关的操作集合。

3. 数据的完整性约束条件

数据的完整性约束条件是一组完整性规则的集合。完整性规则是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,它反映数据数据库状态合理性的约束以及状态变化时应满足的约束条件,以保证数据的正确、有效和相容。

1.2.2 概念模型

概念模型是现实世界到机器世界的一个中间层次。现实世界的事物反映到人的脑子中来,

人们把这些事物抽象为一种既不依赖于具体的计算机系统又不为某一 DBMS 支持的概念模型，然后再把概念模型转换为计算机上某一 DBMS 支持的数据模型。

1. 概念模型的主要概念

实体:客观存在并可相互区别的事物及其事物之间的联系。例如,一个学生、一门课程、学生的一次选课等都是实体。

属性:实体所具有的某一特性。例如,学生的学号、姓名、性别、出生年份、系、入学时间等。

码:惟一标识实体的属性集。例如,学号是学生实体的码。

域:属性的取值范围。例如,年龄的域为大于 15 小于 35 的整数,性别的域为(男,女)。

实体型:用实体名及其属性名集合来抽象和刻画同类实体,称为实体型。例如,学生(学号,姓名,性别,出生年份,系,入学时间)就是一个实体型。

实体集:同型实体的集合称为实体集。例如,全体学生就是一个实体集。

联系:实体与实体之间以及实体与组成它的各属性间的关系。联系有三种情况:一对一联系、一对多联系和多对多联系。

2. 概念模型的表示方法

概念模型的表示方法很多,最常用的是实体-联系方法。该方法用 E-R 图来描述现实世界的概念模型。

E-R 图提供了表示实体型、属性和联系的方法。

实体型:用矩形表示,矩形框内写明实体名。

属性:用椭圆形表示,并用无向边将其与相应的实体连接起来。

联系:用菱形表示,菱形框内写明联系名,并用无向边分别与有关实体连接起来,同时在无向边旁标上联系的类型(1:1,1:n 或 m:n)。

例如图 1-1 就是一个班级的概念模型的 E-R 图。

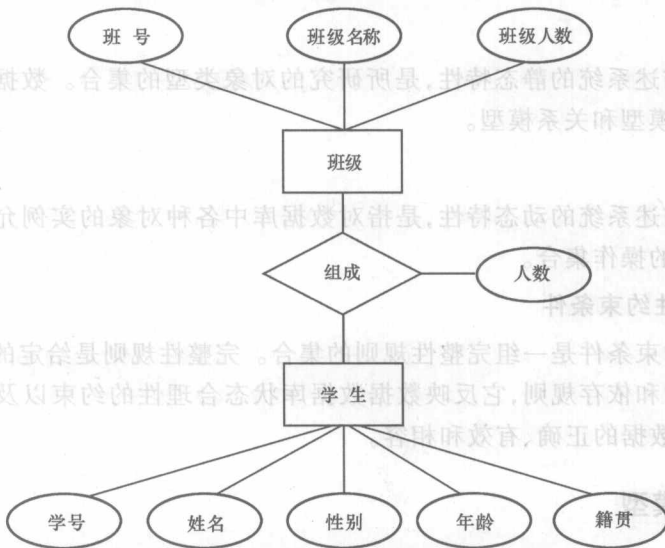


图 1-1 是一个班级的概念模型 E-R 图

1.2.3 数据模型

目前最常用的数据模型有层次模型、网状模型和关系模型。其中层次模型和网状模型统称为非关系模型。

1. 层次数据模型

层次模型是数据库系统中最早出现的数据模型,它用树形结构表示各类实体以及实体间的联系。层次模型数据库系统的典型代表是 IBM 公司的 IMS (Information Management Systems) 数据库管理系统,这是一个曾经广泛使用的数据库管理系统。

在数据库中,对满足以下两个条件的数据模型称为层次模型:

(1) 有且仅有一个结点无双亲,这个结点称为“根结点”;

(2) 其他结点有且仅有一个双亲。

若用图来表示,层次模型是一棵倒立的树。结点层次(Level)从根开始定义,根为第一层,根的孩子称为第二层,根称为其孩子的双亲,同一双亲的孩子称为兄弟。

图 1-2 给出了一个系的层次模型。

层次模型对具有一对多的层次关系的描述非常自然、直观、容易理解,这是层次数据库的突出优点。

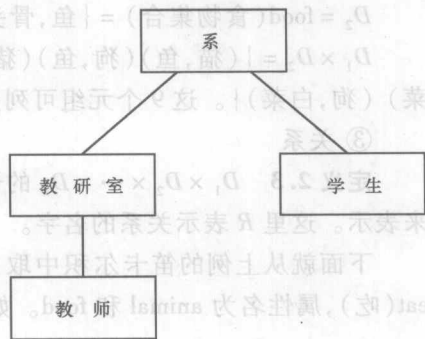


图 1-2 简单的层次模型

2. 网状数据模型

在数据库中,对满足以下两个条件的数据模型称为网状模型:

(1) 允许一个以上的结点无双亲;

(2) 一个结点可以有多于一个的双亲。

网状数据模型的典型代表是 DBTG (Data Base Task Group) 系统,也称 CODASYL (Conference On Data Systems Language) 系统。这是 20 世纪 70 年代数据系统语言研究会 CODASYL 下属的数据库任务组提出的一个系统方案。若用图表示,网状模型是一个网络。

图 1-3 给出了一个抽象的简单的网状模型。自然界中实体型间的联系更多的是非层次关系,用层次模型表示非树形结构是很不直接的,网状模型则可以克服这一弊病。

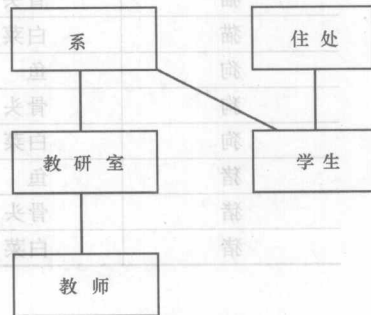


图 1-3 简单的网状模型

3. 关系数据模型

关系模型是目前最重要的一种模型。美国 IBM 公司的研究员 E. F. Codd 于 1970 年发表题为“大型共享系统的关系数据库的关系模型”的论文,文中首次提出了数据库系统的关系模型。20 世纪 80 年代以来,计算机厂商新推出的数据库管理系统几乎都支持关系模型,非关系系统的产品也大都加上了关系接口。数据库领域当前的研究工作都是以关系方法为基础。本书的重点也将放在关系数据模型上。这里只简单介绍一下关系模型。

(1) 关系模型的基本概念

① 域

定义 2.1 域是一组具有相同数据类型的值的集合。
例如,整数、实数、字符串、{男,女},大于0小于等于100的正整数等都可以是域。

② 笛卡尔积

定义 2.2 给定一组域 $D_1, D_2, \dots, D_n$, 则这组域的笛卡尔积为

$$D_1 \times D_2 \times \dots \times D_n = \{(d_1, d_2, \dots, d_n) \mid d_i \in D_i, i = 1, 2, \dots, n\}$$

其中每一个元素 $(d_1, d_2, \dots, d_n)$ 叫作一个 n 元组,元素中的每一个值 d_i 叫作一个分量。

例如,给出两个域:

$D_1 = \text{animal}(\text{动物集合}) = \{\text{猫}, \text{狗}, \text{猪}\}$

$D_2 = \text{food}(\text{食物集合}) = \{\text{鱼}, \text{骨头}, \text{白菜}\}$

$D_1 \times D_2 = \{(\text{猫}, \text{鱼})(\text{狗}, \text{鱼})(\text{猪}, \text{鱼})(\text{猫}, \text{骨头})(\text{狗}, \text{骨头})(\text{猪}, \text{骨头})(\text{猫}, \text{白菜}), (\text{猪}, \text{白菜})(\text{狗}, \text{白菜})\}$ 。这9个元组可列成一张二维表,如表1-1所示。

③ 关系

定义 2.3 $D_1 \times D_2 \times \dots \times D_n$ 的子集叫做在域 $D_1, D_2, \dots, D_n$ 上的关系,用 $R(D_1, D_2, \dots, D_n)$ 来表示。这里 R 表示关系的名字。

下面就从上例的笛卡尔积中取出一个子集来构造一个关系 $\text{eat}(\text{animal}, \text{food})$, 关系名字为 $\text{eat}(\text{吃})$, 属性名为 animal 和 food 。如表1-2所示。

表 1-1 D_1, D_2 的笛卡尔积

animal(动物)	food(食物)
猫	鱼
猫	骨头
猫	白菜
狗	鱼
狗	骨头
狗	白菜
猪	鱼
猪	骨头
猪	白菜

表 1-2 关系 eat

animal	food
猫	鱼
狗	骨头
猪	白菜

(2) 关系的性质

① 列是同质的,即每一列中的分量是同一类型的数据,来自同一个域。

② 不同的列可出自同一个域,称其中的每一列为一个属性,不同的属性要给予不同的属性名。

③ 列的顺序无所谓,即列的次序可以任意交换。

④ 任意两个元组不能完全相同。

⑤ 行的顺序无所谓,即行的次序可以任意交换。

⑥ 分量必须取原子值,即每一个分量都必须是不可分的数据项。

(3) 关系数据结构

在用户看来,一个关系模型的逻辑结构是一张二维表,它由行和列组成。例如,表 1-3 中的人事记录就是一个关系模型,它涉及下列概念。

表 1-3 人事记录表

学 号	姓 名	性 别	年 龄	所 在 系
0000101	王萧	男	17	经济系
000207	李云虎	男	18	机械系
010302	郭敏	女	18	信息系
010408	高红	女	20	土木系
⋮	⋮	⋮	⋮	⋮
020309	王睿	男	19	信息系
020506	路旭青	女	21	管理学

关系:一个关系对应一张二维表,表 1-3 中的这张学生记录表就是一个关系元组:表中的一行即为一个元组,若表 1-3 有 20 行,就有 20 个元组。

属性:表中的一列即为一个属性,表 1-3 有 5 列,对应 5 个属性(学号,姓名,性别,年龄和所在系)。

码(key):表中的某个属性(组),它可以惟一确定一个元组,则称该属性组为“候选码”。若一个关系有多个候选码,则选定其中一个为主码。如表 1-3 中的学号,是该学生关系的码。

域(domain):属性的取值范围,如表 1-3 中学生年龄的域应是(16~28),性别的域是(男,女),系别的域是一个学校所有系名的集合。

分量:元组中的一个属性值。

关系模式:对关系的描述,一般表示为:

关系名(属性 1,属性 2,⋯,属性 n)

例如,上面的关系可描述为:

学生(学号,姓名,性别,年龄,所在系)

(4) 关系数据模型的存储结构

关系数据模型中,实体及实体间的联系都用表来表示。在数据库的物理组织中,表以文件形式存储,每一个表通常对应一种文件结构。

(5) 关系数据模型的优缺点

关系模型与非关系模型不同,它是建立在严格的数学概念的基础上的。

关系模型的概念单一。无论实体还是实体之间的联系都用关系来表示。对数据的检索结果也是关系(即表)。所以结构简单、清晰,用户易懂易用。

关系模型的存取路径对用户透明,从而具有更高的数据独立性,更好的安全保密性,也简化了程序员的工作和数据库开发建立的工作。所以关系数据模型诞生以后发展迅速,深受用户的喜爱。

当然,关系数据模型也有缺点,其中最主要的缺点是,由于存取路径对用户透明,查询效率往往不如非关系数据模型。因此,为了提高性能,必须对用户的查询请求进行优化,增加了开发数

数据库管理系统的负担。

1.3 数据库系统结构

从数据库管理系统角度看待数据库结构可以发现数据库系统采用三级模式结构。

1.3.1 数据库系统的模式结构

1. 数据库系统的三级模式结构

数据库系统的三级模式结构是指数据库系统是由外模式、模式和内模式三级组成。

(1) 外模式

外模式也称子模式或用户模式,它是数据库用户(包括应用程序员和最终用户)看见和使用的局部数据的逻辑结构和特征的描述,是数据库用户的数据视图,是与某一应用有关的数据的逻辑表示。一个数据库可以有多个外模式。

(2) 模式

模式也称逻辑模式,是数据库中全体数据的逻辑结构和特征的描述,是所有用户的公用数据视图。一个数据库只有一个模式。

(3) 内模式

内模式也称存储模式,它是数据物理结构和存储结构的描述,是数据在数据库内部的表示方式。一个数据库只有一个内模式。

2. 数据库的二级映像功能与数据独立性

数据库系统在这三级模式之间提供了两层映像:外模式/模式映像和模式/内模式映像。正是这两层映像保证了数据库系统的数据能够具有较高的逻辑独立性和物理独立性。

模式描述的是数据的全局逻辑结构,外模式描述的是数据的局部逻辑结构。对应于同一个模式可以有任意多个外模式。对于每一个外模式,数据库系统都有一个外模式/模式映像,它定义了该外模式与模式之间的对应关系。当模式改变时(例如,增加新的数据类型、新的数据项、新的关系等),由数据库管理员对各个外模式/模式的映像作相应改变,可以使外模式保持不变,从而应用程序不必修改,保证了数据的逻辑独立性。

数据库中只有一个模式,也只有一个内模式,所以模式/内模式映像是惟一的,它定义了数据全局逻辑结构与存储结构之间的对应关系。当数据库的存储结构改变了(例如,采用了更先进的存储结构),由数据库管理员对模式/内模式映像作相应改变,可以使模式保持不变,从而保证了数据的物理独立性。

1.3.2 数据库系统的体系结构

从最终用户角度来看,数据库系统体系结构分为单用户结构、主从式结构、分布式结构和客户/服务器结构。

1. 单用户数结构

单用户结构是一种早期的最简单的体系结构。在这种结构中,整个数据库系统(包括应用程序、DBMS、数据)都装在一台计算机上,由一个用户独占,不同机器之间不能共享数据。

2. 主从式结构

主从式结构是指一个主机带多个终端的多用户结构。在这种结构中,数据库系统(包括应用程序、DBMS、数据)都集中存放在主机上,所有处理任务都由主机来完成,各个用户通过主机的终端并发地存取数据库,共享数据资源。

3. 分布式结构

分布式结构是指数据库中的数据在逻辑上是一个整体,但物理地分布在计算机网络的不同结点上。网络中的每个结点都可以独立处理本地数据库中的数据,执行局部应用;同时也可以同时存取和处理多个异地数据库中的数据,执行全局应用。

4. 客户/服务器结构

主从式数据库系统中的主机和分布式数据库系统中的每个结点机是一个通用计算机,既执行 DBMS 功能又执行应用程序。随着工作站功能的增强和广泛使用,人们开始把 DBMS 功能和应用分开,网络中某个(些)结点上的计算机专门用于执行 DBMS 功能,称为数据库服务器,简称服务器,其他结点上的计算机安装 DBMS 的外围应用开发工具,支持用户的应用,称为客户机,这就是客户/服务器结构的数据库系统。

在客户/服务器结构中,客户端的用户请求被传送到数据库服务器,数据库服务器进行处理后,只将结果返回给用户(而不是全部数据),从而显著减少了网络上的数据传输量,提高了系统的性能、吞吐量和负载能力。另一方面,客户/服务器结构的数据库往往更加开放。客户与服务器一般都能在多种不同的硬件和软件平台上运行,可以使用不同厂商的数据库应用开发工具,应用程序具有更强的可移植性,同时也可以减少软件维护开销。

1.3.3 数据库管理系统

数据库管理系统是数据库系统的核心,是为数据库的建立、使用和维护而配置的软件。它建立在操作系统的基础上,是位于操作系统与用户之间的一层数据管理软件,负责对数据库进行统一的管理和控制。用户发出的或应用程序中的各种操作数据库中数据的命令,都要通过数据库管理系统来执行。数据库管理系统还承担着数据库的维护工作,能够按照数据库管理员所规定的要求,保证数据库的安全性和完整性。

1. DBMS 的功能

由于不同 DBMS 要求的硬件资源、软件环境是不同的,因此其功能与性能也存在差异,但一般说来,DBMS 的功能主要包括以下 6 个方面。

(1) 数据定义

数据定义包括定义构成数据库结构外模式、模式和内模式,定义各个外模式与模式之间的映射,定义模式与内模式之间的映射,定义有关的约束条件。例如,为保证数据库中数据具有正确语义而定义的完整性规则、为保证数据库安全而定义的用户口令和存取权限等。

(2) 数据操纵

数据操纵包括对数据库数据的检索、插入、修改和删除等基本操作。

(3) 数据库运行管理

对数据库的运行进行管理是 DBMS 运行时的核心部分,包括对数据库进行并发控制、安全性检查、完整性约束条件的检查和执行、数据库的内部维护(如索引、数据字典的自动维护)等。所有访问数据库的操作都要在这些控制程序的统一管理下进行,以保证数据的安全性、完整性、一致性以及多用户对数据库的并发使用。

(4) 数据组织、存储和管理

数据库中需要存放多种数据,如数据字典、用户数据、存取路径等,DBMS 负责分门别类地组织、存储和管理这些数据,确定以何种文件结构和存取方式物理地组织这些数据,如何实现数据之间的联系,以便提高存储空间利用率以及提高随机查找、顺序查找、增、删、改等操作的时间效率。

(5) 数据库的建立和维护

建立数据库包括数据库初始数据的输入与数据转换等。维护数据库包括数据库的转储与恢复、数据库的重组织与重构造、性能的监视与分析等。

(6) 数据通信接口

DBMS 需要提供与其他软件系统进行通信的功能。例如,提供与其他 DBMS 或文件系统的接口,从而能够将数据转换为另一个 DBMS 或文件系统能够接受的格式,或者接收其他 DBMS 或文件系统的数据库。

2. DBMS 的组成

DBMS 通常由以下 4 个部分组成。

(1) 数据定义语言及其翻译处理程序

DBMS 一般都提供数据定义语言(Data Definition Language, DDL)供用户定义数据库的外模式、模式、内模式、各级模式间的映射和有关的约束条件等。用 DDL 定义的外模式、模式和内模式分别称为源外模式、源模式和源内模式,各种模式翻译程序负责将它们翻译成相应的内部表示,即生成目标外模式、目标模式和目标内模式。

(2) 数据操纵语言及其编译(或解释)程序

DBMS 提供了数据操纵语言(Data Manipulation Language, DML)实现对数据库的检索、插入、修改和删除等基本操作。DML 分为宿主型 DML 和自主型 DML 两类。宿主型 DML 本身不能独立使用,必须嵌入主语言中,例如,嵌入 C、COBOL、FORTRAN 等高级语言中。自主型 DML 又称为自含型 DML,它们是交互式命令语言,语法简单,可以独立使用。

(3) 数据库运行控制程序

DBMS 提供了一些负责数据库运行过程中的控制与管理的系统运行控制程序,包括系统初始化程序、文件读写与维护程序、存取路径管理程序、缓冲区管理程序、安全性控制程序、完整性检查程序、并发控制程序、事务管理程序和运行日志管理程序等,它们在数据库运行过程中监视着对数据库的所有操作,控制管理数据库资源,处理多用户的并发操作等。

(4) 实用程序

DBMS 通常还提供一些实用程序,包括数据初始装入程序、数据转储程序、数据库恢复程序、