

21世纪高等学校教材

Visual FoxPro 程序设计

主编 冷金麟 周治钰



上海交通大学出版社
SHANGHAI JIAO TONG UNIVERSITY PRESS

21 世纪高等学校教材

Visual FoxPro 程序设计

主编 冷金麟 周治钰

上海交通大学出版社

内 容 提 要

本书是根据教育部高等学校非计算机专业计算机基础课程教学指导委员会提出的有关“数据库技术及应用”要求,并配合全国计算机等级考试“Visual FoxPro 二级”考试大纲的知识要点,结合多年来的教学与科研开发经验而精心编写的。

本书融数据库基本理论、数据库管理系统和数据库应用程序开发为一体,既注重基础理论,又强调实际应用,在突出面向对象概念的同时又兼顾传统的面向过程的程序设计方法,以数据库应用系统开发为主线,讲解 Visual FoxPro 的基本操作方法和操作命令,介绍面向对象的程序设计方法、数据库设计以及数据库应用系统的开发方法和步骤。全书以一个完整的系统开发案例为导引,将一个实用系统分解到若干章节中进行讲解,并在最后章节中再整合成一个完整的应用系统发布,深入浅出,整体性强,以使读者学会 Visual FoxPro 的基本操作,掌握面向对象的程序设计及数据库应用系统的开发。

本书体系完整、结构清晰、图文并茂、示例翔实、精编细讲,简读易懂,全书案例具有创新性和普遍适用性,可作为高等院校非计算机专业本科生、专科生的教材,也可作为学习数据库应用技术的读者自学用书。

与书配套的实验指导书同时出版。同时,本书配有多媒体课件(PPT 格式)、源程序、教学网站(网上答疑、发布与提交作业及作业自动判分)和自动考试系统(理论知识和实际操作技能的全部自动化考核)。联系邮箱为: baiwen_sjtu@126.com

图书在版编目(CIP)数据

Visual FoxPro 程序设计/冷金麟,周治钰主编. —上海:
上海交通大学出版社,2012
21 世纪高等学校教程
ISBN 978-7-313-08009-7

I. V... II. ①冷... ②周... III. 关系数据库—数据库管理系统, Visual FoxPro—程序设计—高等学校—教材 IV. TP311.138

中国版本图书馆 CIP 数据核字(2011)第 256174 号

Visual FoxPro 程序设计

冷金麟 周治钰 主编

上海交通大学出版社出版发行

(上海市番禺路 951 号 邮政编码 200030)

电话:64071208 出版人:韩建民

常熟市文化印刷有限公司 印刷 全国新华书店经销

开本:787mm×1092mm 1/16 印张:19 字数:464 千字

2012 年 1 月第 1 版 2012 年 1 月第 1 次印刷

ISBN 978-7-313-08009-7/TP 定价:32.00 元

版权所有 侵权必究

告读者:如发现本书有印装质量问题请与印刷厂质量科联系
联系电话:0512-52219025

21 世纪高等学校教材

编审委员会

执行主任: 百 文

副 主 任:	冷金麟	李 明	靳全勤	张华隆	蒋凤璞
	冯 颖	普杰信	程全洲	潘群娜	杨裕根
	徐祖茂	张红梅	宓一鸣	姜献峰	李 敏
	李湘梅	闫洪亮	陈树平	包奇金宝	刘克成
	白丽媛	戴 兵	张占山	曹天守	

前 言

随着计算机及其应用的推广普及,各行各业纷纷建立了以数据库为核心的信息系统,数据库技术在当今信息社会的地位和作用显得尤为重要。Visual FoxPro(简称 VFP)是 Microsoft 公司推出一种基于 Windows 环境下面向对象的关系型数据库管理系统,它本身自带的语言系统涵盖了从面向过程到面向问题,再到面向对象,直至计算机自动编程等程序设计语言的发展过程。VFP 简单易学、操作方便、具有强大的数据处理能力,能满足非计算机专业计算机应用的需求。相对于算法要求较强的其他程序设计语言,它更适合非计算机专业学生修读。

为了适应数据库技术的新发展,我们根据教育部高等学校非计算机专业计算机基础课程教学指导委员会提出的有关“数据库技术及应用”要求,配合全国计算机等级考试“Visual FoxPro 二级”考试大纲的知识要点,结合多年来的教学与科研开发经验,编写了这套《Visual FoxPro 程序设计》及《Visual FoxPro 程序设计实验指导》教材。

本书融数据库基本理论、数据库管理系统和数据库应用程序开发为一体,既注重基础理论,又强调实际应用。全书共分为 10 章,从关系数据库管理系统基础理论出发,以数据库应用系统开发知识为主线,讲解了 VFP 的基本操作方法和操作命令,介绍面向对象程序设计方法以及数据库设计、数据库应用系统开发的方法与步骤,并通过丰富的实例,分析讲解数据库的基本概念和应用程序的开发。全书以一个完整的学生信息管理系统开发案例为导引,将一个实际的数据库应用系统分解到相关章节中进行讲解,并在最后章节里将其整合成一个完整的应用系统发布。本书力图通过深入浅出的分析与讲解,使读者学会 Visual FoxPro 基本操作,掌握面向对象程序设计理念及数据库应用系统开发的方法。

全书由冷金麟、周治钰任主编,参与编写的人员有:黎杰、吴国凤、李明、冷金麟和周治钰,全书由冷金麟统稿。

书中疏漏和不妥之处,恳请各位读者和专家批评指正。

编 者

2011 月 10 月 13 日

目 录

第1章 数据库系统基础知识	1	2.1.3 输入表数据	43
1.1 数据管理技术的发展	1	2.2 常量与变量	45
1.1.1 数据与数据处理	1	2.2.1 常量	45
1.1.2 数据管理技术的发展	2	2.2.2 变量	47
1.2 数据库系统	3	2.3 函数与表达式	51
1.2.1 数据库系统的组成	3	2.3.1 函数	51
1.2.2 数据库的三级模式结构	4	2.3.2 表达式	61
1.2.3 数据库系统的特点	6	2.4 表及其数据的维护命令	66
1.3 数据模型	7	2.4.1 Visual FoxPro 命令的常用子句	66
1.3.1 概念模型	7	2.4.2 表的维护	68
1.3.2 结构数据模型	9	2.4.3 记录的维护	71
1.4 关系数据库	10	思考题	77
1.4.1 关系模型	10	第3章 表的查询与统计	78
1.4.2 关系运算	12	3.1 排序与索引	78
1.4.3 关系的规范化理论	13	3.1.1 表的排序	78
1.4.4 关系的完整性约束	15	3.1.2 表的索引	80
1.5 数据库系统的应用模式与开发工具	16	3.1.3 索引文件的建立	82
1.5.1 数据库系统的应用模式	16	3.1.4 索引文件的使用	86
1.5.2 常见的数据库管理系统及其 开发工具	17	3.2 表的记录查询	90
1.6 Visual FoxPro 系统简介	18	3.2.1 顺序查询	90
1.6.1 Visual FoxPro 的运行	18	3.2.2 索引查询	91
1.6.2 Visual FoxPro 的命令与操作方式	22	3.3 表的统计与计算	94
1.6.3 Visual FoxPro 的设计与管理工具	23	3.3.1 统计记录个数	94
1.6.4 项目管理器	25	3.3.2 求和与求平均值	95
1.6.5 VFP 系统环境的设置	27	3.3.3 求平均值	95
1.6.6 VFP 的系统帮助	29	3.3.4 综合计算	96
思考题	30	3.3.5 分类汇总	96
第2章 数据库与表的基本操作	31	3.4 多个表操作	97
2.1 数据库与表的建立与修改	31	3.4.1 工作区和数据工作期	97
2.1.1 数据库的建立与操作	33	3.4.2 表的关系	99
2.1.2 表结构的建立与修改	35	3.4.3 表的连结	104
		3.4.4 表的数据完整性	105
		思考题	107

第4章 SQL语言的应用	109	5.6 综合应用	161
4.1 SQL语言概述	109	思考题	166
4.1.1 概述	109	第6章 面向对象的程序设计	167
4.1.2 SQL语言的功能	109	6.1 面向对象的基础	167
4.1.3 SQL语言的主要特点	109	6.1.1 面向对象的概述	167
4.1.4 SQL语言的组成	110	6.1.2 对象与类	169
4.2 数据定义	111	6.1.3 Visual FoxPro 中的类与对象	170
4.2.1 基本表结构的定义与维护	111	6.2 表单设计	178
4.2.2 视图的定义和维护	115	6.2.1 用表单向导建立表单	179
4.3 SQL的数据查询功能	116	6.2.2 用表单设计器创建表单	183
4.3.1 查询的概念	116	6.2.3 创建快速表单	190
4.3.2 应用实例	117	6.2.4 创建表单集	191
4.4 数据操纵	121	6.2.5 添加属性和方法程序	191
4.4.1 插入记录	122	6.2.6 单文档界面和多文档界面	193
4.4.2 删除记录	122	思考题	194
4.4.3 更新记录	122	第7章 表单控件的设计与应用	195
思考题	123	7.1 输出类控件	195
第5章 结构化程序设计	125	7.1.1 标签	195
5.1 程序设计基础	125	7.1.2 线条和形状	196
5.1.1 程序设计的概述	125	7.1.3 图像	197
5.1.2 结构化程序设计	127	7.2 输入类控件	199
5.1.3 程序的建立与执行	128	7.2.1 文本框	199
5.2 程序中常用命令	130	7.2.2 编辑框	201
5.2.1 最基本的输入输出命令	130	7.2.3 列表框和组合框	203
5.2.2 信息对话框函数	132	7.2.4 微调按钮	206
5.2.3 其他常用命令	134	7.3 控制类控件	207
5.3 程序的基本结构	135	7.3.1 命令按钮与命令按钮组	207
5.3.1 顺序结构程序	136	7.3.2 复选框	210
5.3.2 选择结构	136	7.3.3 单选按钮组	210
5.3.3 循环结构	140	7.3.4 计时器	213
5.4 程序的模块化	146	7.4 容器类控件	215
5.4.1 过程与自定义函数	146	7.4.1 表格	215
5.4.2 内存变量的作用域	153	7.4.2 页框	217
5.4.3 程序模块间的数据传递	156	7.4.3 容器	218
5.5 程序调试	158	7.5 其他控件	219
5.5.1 程序中常见错误	158	7.5.1 ActiveX 控件	219
5.5.2 查错技术	159	7.5.2 ActiveX 绑定控件	222

7.5.3 超级链接	222	9.2.2 标签设计器	260
7.6 表单控件设计的应用实例	222	思考题	261
思考题	230	第 10 章 数据库应用系统开发	262
第 8 章 菜单和工具栏设计	231	10.1 数据库应用系统的开发步骤	262
8.1 菜单设计	231	10.1.1 系统需求分析	262
8.1.1 菜单设计概述	231	10.1.2 系统设计	263
8.1.2 下拉式菜单设计	232	10.2 学生信息管理系统的开发	266
8.1.3 快捷菜单设计	237	10.2.1 需求分析	266
8.2 为顶层表单添加菜单	238	10.2.2 数据库设计	267
8.3 工具栏设计	240	10.2.3 应用程序设计	269
8.3.1 定制 VFP 工具栏	240	10.3 应用程序的管理与发布	272
8.3.2 创建自定义工具栏	241	10.3.1 系统调试与连编	272
思考题	243	10.3.2 应用程序的发布	275
第 9 章 报表和标签设计	244	思考题	278
9.1 报表设计	244	附录	279
9.1.1 利用报表向导设计报表	245	附录 1 Visual FoxPro 常用文件类型	279
9.1.2 利用报表设计器创建报表	247	附录 2 Visual FoxPro 6.0 常用命令	279
9.1.3 利用报表设计器创建快速报表	258	附录 3 Visual FoxPro 6.0 常用函数	286
9.1.4 报表的输出	258	参考文献	293
9.2 标签设计	259		
9.2.1 标签向导	259		

第 1 章 数据库系统基础知识

数据库(Database)是数据处理的基础,也是管理信息系统(MIS)、办公自动化(OA)系统和决策支持系统(DSS)等应用系统的核心部分,它能够有效、合理地存储各种数据,为有关应用准确、快速地提供有用的信息。作为数据管理的最新技术,数据库技术是计算机领域中最重要技术之一,其应用已渗透到人类社会各个领域,正在改变着人们的生活、工作方式。

要想开发出高效的数据库应用程序,除了会充分利用开发工具外,还应当对数据库的结构与概念有充分的认识,这样才能达到事半功倍的效果,并发挥数据库的强大功能。本章将从数据库的有关基本概念出发,逐一讲解信息、数据、数据处理、数据模型、数据库系统、关系数据库的关系及其运行运算等基本概念和基础知识,同时简要介绍 Visual FoxPro 数据库管理系统的相关基础知识,为后续章节的学习打下基础。

1.1 数据管理技术的发展

计算机技术的发展,使得计算机的应用已从科学计算、过程控制、辅助设计进入了数据处理阶段,数据处理方法和技术的研究已成为计算机科学的重要领域,目前的计算机系统 70% 的工作量都是用于数据处理。数据处理的核心问题就是数据管理,就是利用计算机对各种类型的数据进行处理,包括对数据的采集、分类、组织、编码、存储、排序、检索、维护、加工、统计和传输等一系列操作过程。数据管理并不是一开始就采用数据库技术,数据管理技术的发展也经历了由低级到高级、由简单到逐步完善的发展历程。

1.1.1 数据与数据处理

1. 数据和信息

数据(Data)是人们用于记录事物情况的物理符号。为了描述客观事物而用到的数字、字符以及所有能输入到计算机中并能被计算机处理的符号都可以看做是数据。在实际应用中,数据可分为 3 种基本形式:数值型数据,即对客观事物进行定量记录的、可以进行数值运算的符号,如表示成绩、工资等;字符型数据,即对客观事物进行定性记录、不能进行数值运算符号,如表示姓名、地址、职称的数据;特殊型数据,如图形、图像、声音等多媒体数据。

信息(Information)是经过处理、加工提炼并对人的行为产生影响的数据,是现实世界中的各种事物、事物的特征及其联系等在人脑中的反映。

数据与信息既有区别,又有联系。一方面,任何事物的情况特征都是通过数据来描述的,但数据只有经过加工处理后,使其具有知识性并对人类的客观行为产生影响,才能形成信息。数据是信息的载体,信息是数据的内涵。另一方面,信息是反映客观现实世界的知识,同一信息可以用多种不同的数据形式表现,而数据则具有任意性。例如亚运会某场比赛的报道是一条信息,而描述该信息的数据形式可以是文字、图像或声音等。

2. 数据处理

数据包括 2 个方面的概念，即数据的内容和数据的形式。数据内容是指所描述的客观事物的具体特性，即通常所说数据的“值”；数据形式则是指数据内容存储在媒体上的具体形式，即通常所说的数据的“类型”。数据主要有数字、文字、声音、图形和图像等多种形式。

数据处理是指将数据转换成信息的过程，其基本目的是从大量的、杂乱无章的、难以理解的数据中整理出对人们有价值、有意义的数据(即信息)，作为人们活动决策的依据。例如，学校每学期结束时都会以本学期为学生开设课程的考试成绩作为原始数据，按课程、年级、专业进行统计分析，整理出课程学习不合格的学生信息，用于下学期开学初的课程补考工作安排。

1.1.2 数据管理技术的发展

数据管理技术是指对数据进行分类、组织、编码、存储、检索和维护的技术。在计算机环境下，数据管理技术经历了从低级到高级的 3 个发展阶段：

1. 人工管理阶段

20 世纪 50 年代中期以前，当时的计算机主要用于科学计算，计算处理的数据量很小。当时的计算机没有专门的管理数据的软件，也没有像磁盘这样可随机存取的外部存储设备，对数据的管理也没有一定的格式。数据依附于处理它的应用程序，应用程序不仅要规定数据的逻辑结构，而且还要设计其物理结构，包括存储结构、存取方法及输入输出方式等。数据与程序一一对应，相互依赖。应用程序中的数据无法被其他程序利用，即使 2 个应用程序涉及某些相同的数据，也必须各自定义，不能相互利用，互相参照，使得程序与程序之间有大量的冗余数据。由于数据是对应于某一应用程序，使得数据的独立性很差，如果数据的类型、结构、存取方式或输入输出方式发生变化，处理它的程序也必须进行相应的改变，应用程序的设计与维护的负担繁重。

2. 文件系统阶段

20 世纪 50 年代后期至 60 年代后期，随着计算机软硬件技术的飞速发展，计算机应用领域不断拓宽，不仅用于科学计算，而且也用于一些非数值数据的处理。硬件方面，出现了直接存取的大容量外存储器，如磁盘、磁鼓等，这为计算机管理系统管理数据提供了物质基础。软件方面，出现了操作系统，其中包含了文件系统；处理方式上不仅有了文件批处理，而且能够实现联机实时处理。这些都为数据管理提供了技术支持。

在这一阶段，数据可以以操作系统的文件形式长期保存在计算机的外存储器上，文件的组织方式由顺序文件逐步发展到随机文件。操作系统的文件管理系统提供了对数据的输入和输出操作接口，进而提供数据存取方法。但文件系统只是简单地存放数据，数据的存取在很大程度上仍依赖于应用程序，不同程序仍难于共享同一数据文件，数据的独立性较差。数据仍是面向应用的，文件之间彼此孤立，不能反映数据之间的联系，因而仍存在数据的大量冗余和不一致性。

3. 数据库系统阶段

20 世纪 60 年代后期，针对数据的文件管理方式存在的问题，计算机软件工作人员经

过长期不懈地努力,提出了数据库的概念,开发了一类新的数据管理软件——数据库管理系统(DataBase Management System, DBMS)。数据库管理系统运用数据库技术进行数据管理,将数据管理技术推向了数据库管理阶段。

数据库技术为数据管理提供了一种较完善的高级管理方式,它克服了文件系统方式下分散管理的弱点,对所有的数据实行统一、集中的管理,使数据的存储独立于使用它的程序,从而实现数据共享,并对数据的冗余度实现有效控制,保证数据的完整性和安全性,提高了数据的管理效率。数据库也是以文件方式存储数据的,但它是数据的一种高级组织形式。在应用程序和数据库之间,由数据库管理系统软件把所有应用程序中使用的相关数据汇集起来,按统一的数据模型,以记录为单位存储在数据库中,为各个应用程序提供方便、快捷的查询和调用功能。

数据库技术的发展先后经历了层次数据库、网状数据库和关系数据库几个阶段。层次数据库和网状数据库可以看做是第一代数据库系统,关系数据库可以看做是第二代数据库系统。自20世纪70年代提出关系数据模型和关系数据库后,数据库技术得到了蓬勃发展,应用也越来越广泛。而随着应用的不断深入,占主导地位的关系数据库系统已不能满足新的应用领域的需求。例如,在实际应用中,除了需要处理数字、字符数据的简单应用之外,还需要存储并检索复杂的复合数据(如集合、数组、结构)、多媒体数据、计算机辅助设计绘制的工程图纸和地理信息系统提供的空间数据等。对于这些复杂数据,关系数据库都无法实现对它们的管理。需求推动发展,正是实际中出现的许多问题,使得数据库技术不断向前发展,数据库技术与网络通信技术、人工智能技术、面向对象程序设计技术、并行计算技术等相互渗透、相互结合,成为当前数据库技术的发展方向,涌现出许多不同类型的新型数据库系统,如分布式数据库和并行数据库、面向对象数据库、多媒体数据库、知识库和数据挖掘、空间数据库等等。

3个阶段数据管理技术的特点如表1-1所示。

表 1-1 3个阶段数据管理技术的特点

	手工管理	文件系统管理	数据库系统管理
数据的管理者	用户(程序员)	文件系统	数据系统
数据的针对者	特定应用程序	面向某一应用	面向整体应用
数据的共享性	无共享	共享性差,冗余大	共享性好,冗余小
数据的独立性	无独立性	独立性差	独立性好
数据的结构化	无结构	记录有结构,整体无结构	整体结构化

1.2 数据库系统

数据库系统其实就是以数据库应用为基础的计算机系统,和其他的应用系统相比,数据库系统有其自身的特点。

1.2.1 数据库系统的组成

数据库系统是把有关计算机硬件、软件、数据和人员组合起来,为用户提供信息服务的整个计算机系统,因此,数据库系统除了计算机软、硬件环境外,还包括以下几个组织部分。

1. 数据库

数据库(DataBase, DB)是数据库系统中按一定规则存储在外存储器中, 能为多个用户共享且独立于应用程序的相关数据的集合, 它不仅包括描述事物的数据本身, 也包括相关事物之间的联系。

2. 数据库管理系统

数据库管理系统(DataBase Management System, DBMS)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系统软件。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; 它提供数据定义、数据操作、数据库管理和控制等功能, 用户发出的命令或应用程序中的各种操作数据库中数据的命令, 都是通过数据库管理系统来实现的; 数据库管理系统还承担着数据库的维护工作, 能够根据数据库管理员设置的规则, 保证数据库的安全性和完整性。目前, 较流行的数据库管理系统有 Oracle、Sybase、SQL Server、Access、Visual FoxPro 等。

3. 数据库应用系统

数据库应用系统是系统开发人员利用数据库系统资源开发出来的, 面向某一类实际应用的应用软件系统。通常可分为 2 类: 一类是面向单位内部业务和管理的数据库应用系统, 即管理信息系统, 例如, 仓储物资管理系统、企业工资管理系统等。另一类是面向外部、提供动态信息查询功能, 以满足不同信息需求的数据库应用系统, 又称为开放式信息服务系统。例如, 天气预报查询系统, 股市实时行情信息系统及各种信息查询系统等。无论哪一类信息系统, 都是以数据库技术为基础的计算机应用系统。

4. 人员

数据库系统的有关人员主要是指开发、管理和使用数据库系统的所有人员, 通常包括数

据库管理员、系统开发人员和终端用户。数据库管理员(DataBase Administrator, DBA)负责管理、监督、维护数据库系统的正常运行。系统开发人员包括系统分析员、系统设计员和程序员。系统分析员负责应用程序的分析, 他们和用户、数据库管理员相配合, 参与系统分析; 系统设计员负责应用系统设计和数据库设计; 程序员则根据设计要求进行编码。终端用户是指在 DBMS 与应用程序的支持下, 操作使用数据库系统的普通使用者。不同规模的数据库系统, 系统相关人员的配置可以根据实际情况有所不同, 大多数人员都属于终端用户, 在小型数据库系统中, 特别是在微机上运行的数据库系统中, 通常 DBA 就由终端用户担任。数据库系统结构图如图 1-1 所示。

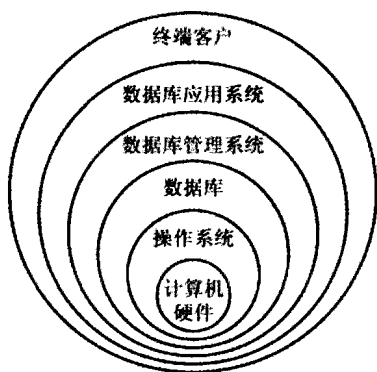


图 1-1 数据库系统层次结构图

1.2.2 数据库的三级模式结构

设计数据库时, 强调的是数据库的结构, 应用数据库时, 则关心的是数据库中的数据。数据库中的数据是经常变化的, 而其结构则具有相对的稳定性。为了有效地组织、管理数据,

提高数据库的逻辑独立性和物理独立性，人们为数据库设计了一个严谨的体系结构，数据库领域公认的标准结构是三级模式结构，它包括外模式、模式和内模式，如图 1-2 所示。

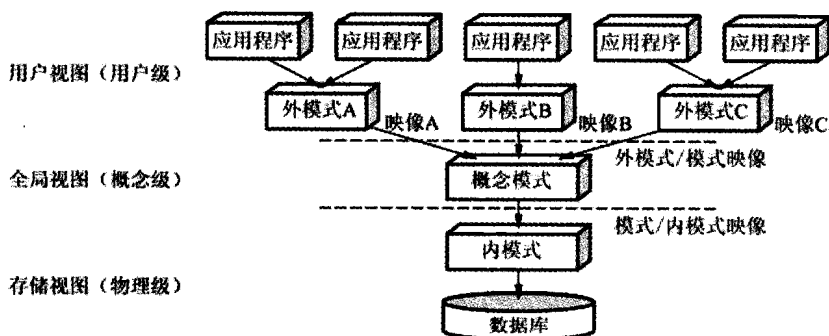


图 1-2 数据库系统的体系结构

1. 数据库系统模式的概念

数据模型中涉及“型”(Type)和“值”(Value)的概念。型是对某一类数据的结构和属性的说明，值是型的一个具体赋值。例如：学生记录定义为(学号，姓名，性别，出生日期)这样的记录型，而(07011001，李珊，女，1988.10.10)则是该记录型的一个记录值。

模式是数据库中全体数据的逻辑结构和特征的描述，它仅仅涉及型的描述，不涉及具体的值。模式的一个具体值则称为模式和一个实例。同一个模式可以有很多实例。模式是相对稳定的，而实例是相对变化的，因为数据库中的数据总是在不断地更新。模式反映的是数据的结构和联系，而实例反映的是数据库在某一时刻的状态。

2. 模式

模式又称概念模式或逻辑模式，对应于概念级。是数据库中全体数据的逻辑结构和特征的描述，是所有用户的公共数据视图。它是数据库系统模式结构的中间层，既不涉及数据的物理存储细节和硬件环境，也不与具体的应用程序、所使用的应用开发工具及高级程序设计语言有关。

模式实际上是数据库数据在概念级上的视图，一个数据库只有一个模式。数据库模式以某一种数据模型为基础，统一综合地考虑所有用户的需求，并将这些需求有机地结合成一个逻辑整体。定义模式时不仅要定义数据的逻辑结构，例如数据记录由哪些数据项构成，数据项的名字、类型、取值范围等，而且要定义数据之间的联系，定义与数据有关的安全性、完整性要求。

DBMS 提供模式数据定义语言 DDL 来描述概念模式，即严格地定义数据的名称、特征、相互关系、约束等。概念模式的基础是数据模型。

3. 外模式

外模式又称子模式或用户模式，对应于用户级。它是数据库用户(包括应用程序员和终端用户)能够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。

通常，外模式是模式的子集，一个数据库中可以包含多个外模式。由于外模式是各个不

同用户的数据视图,因此,当不同的用户在应用需求、看待数据的方式、对数据保密的要求等方面有所差异时,则外模式的描述也是不同的。即使对模式中同一数据,在外模式中的结构、类型、长度、保密级别等都可以不同。此外,同一外模式也可以为某一用户的多个应用系统所使用,但一个应用程序只能使用一个外模式。

外模式是保证数据安全性的一个有力保障,每个用户只能看见和访问所对应的外模式中的数据,数据库中的其余数据是不可见的。

DBMS 提供子模式的描述语言(子模式 DDL)来严格地定义子模式。

4. 内模式

内模式又称存储模式,对应于物理级,一个数据库只有一个内模式。它是数据物理结构和存储方式的描述,是数据库内部的表示方法。例如,记录的存储方式是顺序存储、按照 B 树结构存储还是按 hash 方法存储;索引以什么方式组织;数据是否压缩存储,是否加密;数据存储记录结构有什么规定等。

5. 三级模式间的映像

数据库的三级模式是数据在 3 个层次上的抽象。使用户能够逻辑地、抽象地处理数据而不必关心数据在计算机中的物理表示和存储方式。为了能够在数据库内部实现这 3 个抽象层次的联系和转换,数据库管理系统在这三级模式之间提供了 2 层映像:外模式/模式映像,模式/内模式映像。

(1) 外模式/模式映像。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/模式的映像,它定义该外模式和模式之间的对应关系。当模式发生改变时(例如增加新的关系、新的属性、改变属性的数据类型等),则数据库管理对各个外模式/模式的映像作相应改变,就可以使外模式保持不变,对应的应用程序也可保持不变。应用程序是依据数据的外模式编写的,从而应用程序不必修改,保证了数据与程序的逻辑独立性。

(2) 模式/内模式映像。数据库中只有一个模式,因此,也只有一个内模式,所以模式/内模式映像是唯一的。它定义了数据库全局逻辑结构与存储结构之间的对应关系,例如,说明逻辑记录和字段在内部是如何表示的。该映像定义通常包含在模式描述中,当数据库的存储结构发生改变时(例如,选用了另一种存储结构),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,从而应用程序也不必改变,这样就保证了数据与程序的物理独立性。

数据与程序之间的独立性,使得数据的定义和描述可以从应用程序中分离出去。另外由于数据的存取由 DBMS 管理,用户可以不必考虑存取路径等细节,从而简化了应用程序的编制,大大减少了应用程序的维护和修改。

1.2.3 数据库系统的特点

与其他系统相比,数据库系统具有下列特点。

1. 数据结构化

在文件系统中,各个文件不存在相互联系。从单个文件来看,数据一般是有结构的;但

从整个系统来看,数据在整体上又是没有结构的。数据库系统则不同,在同一数据库中的数据文件是有联系的,且在整体上服从一定的结构形式。

2. 数据共享

数据共享是指多个用户可以同时存取数据而不相互影响,数据共享包括以下3个方面:

- (1) 所有用户可以同时实施数据存取。
- (2) 在满足现有用户的服务需求基础上,也可以为将来的新用户提供服务。
- (3) 可以满足多种语言通过接口访问数据库。

3. 可控数据冗余

在非数据库系统中,由于每个应用程序都有自己的数据文件,难免有许多数据相互重复,这就是冗余。实现共享后,不必要的重复可以全部消除,但为了提高查询效率,有时也保留少量的重复数据,其冗余度可由设计人员控制。

4. 较高的数据独立性

数据独立是指数据与应用程序之间彼此独立,不存在相互依赖的关系,即数据的逻辑结构、存储结构和存取方式等,不因应用程序的修改而修改,反之亦然。在数据库系统中,数据库管理系统通过映像实现了应用程序对数据的逻辑结构与物理存储结构之间的较高独立性。数据库的数据独立包括2个方面:

(1) 物理数据独立。即数据的物理结构(或存储结构)的改变,如物理存储设备的更换、物理存储位置的变更、存取方式的改变等,不影响数据库的逻辑结构,从而不引起应用程序的修改。

(2) 逻辑数据独立。数据库总体逻辑结构的改变,如修改数据的定义、新加新的数据类型、改变数据间的联系等等,无需修改原来的应用程序,这就是数据的逻辑独立性。只是到目前为止,数据的逻辑独立性还没有完全彻底的实现。

数据独立提高了数据处理系统的稳定性,从而提高了程序维护的效益。增强了数据安全性和完整性保护。

此外,由于数据库是多用户的共享资源,而计算机的共享一般是并发的,即多用户同时使用数据库,因此,数据的安全性和可靠性是一个数据库系统是否实用的关键问题。在数据库中加入了安全保密机制,可以防止对数据的非法存取。通过实行集中控制,有利于保障数据的完整性。数据库系统采取了并发访问控制,保证了数据的正确性。另外,数据库系统还采取了一系列措施,实现了对数据库破坏的恢复。

1.3 数据模型

数据模型是对数据库结构和语意的一种抽象,是现实世界事物及事物之间联系的数据表示。从数据组织形式来看,数据模型就是数据库的数据结构形式。

1.3.1 概念模型

人们把客观存在的事物以数据的形式存储到计算机中,经历了对现实世界事物特征及事

物间的内在联系在人们头脑中的反映认识,再到计算机数据库里的具体表示的逐级抽象过程。客观现实世界在人们头脑中的反映,称为信息世界。现实世界中的客观事物在信息世界中称为实体。在信息世界中用来描述客观事物的特征及事物间内在联系的模型即为实体模型或概念模型。

1. 基本概念

(1) 实体(Entity)。客观存在并可相互区分的事物。实体可以指实际存在的对象,也可以指某些概念。例如一名学生、一辆汽车、一个公司、一次会议等。

(2) 属性(Attribute)。实体所具有的某些特性称为属性,属性由属性名和属性值 2 个部分组成。一个实体可以由若干个属性来描述。例如某学生可以由学号、姓名、性别、出生日期、班号等属性来描述。(07021010, 费敏, 女, 1988/01/28, JS0702)这些属性值组合起来表征了一个学生。

(3) 关键字(Key)。唯一标识实体的属性或属性的组合称为关键字,又称码、键。例如,学号是学生实体的关键字。

(4) 域(Domain)。实体某个(些)属性的取值范围。例如:学号的域为 8 位数字;姓名的域为字符串集合;性别的域为 2 个汉字(男,女)。

(5) 实体型(Entity Type)与实体集(Entity Set)。具有相同属性的实体具有共同的特征和性质,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、出生日期、班号)是一个实体型。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。

(6) 联系(Relationship)。现实世界中的客观事物之间是有联系的,这种联系在信息世界中表现为实体之间相互关联的关系,称之为联系。如老师和学生之间的联系是老师给学生授课。

2. 联系的类型

根据联系的特点,可将联系分成如下 3 种基本类型:

(1) 一对一联系(1:1)。若对于实体集 A 中的每一个实体,实体集 B 中至多有一个实体与其联系,反之亦然,则称实体集 A 与实体集 B 的联系为一对一联系,记为 1:1。例如在一个学校中,存在有“班级”和“班长”2 个实体集,从“班级”实体集中任取一个班级实体,在“班长”实体集中至多仅能找到一个班长实体与其联系;反之,从“班长”实体集中任取一个班长实体,在“班级”实体集中至多仅有一个班级实体与其联系,所以“班级”和“班长”2 个实体集属于一对一联系。

(2) 一对多联系(1:n)。若对于实体集 A 中的每一个实体,实体集 B 中有 $n(n \geq 0)$ 个实体与其联系;反之,对于实体集 B 中的每一个实体,实体集 A 中至多只有一个实体与其联系,则称实体集 A 与实体集 B 的联系为一对多联系。记为 1:n。例如在一个学校中,存在有“班级”和“学生”2 个实体集,从“班级”实体集中任取一个班级实体,在“学生”实体集中能找到多名学生实体与其联系;反之,从“学生”实体集中任取一名学生实体,在“班级”实体集中至多仅有一个班级实体与其联系,所以“班级”和“学生”2 个实体集属于一对多联系。

(3) 多对多联系(m:n)。若对于实体集 A 中的每一个实体,实体集 B 中有 $n(n \geq 0)$ 个实体与其联系,反之,对于实体集 B 中的每一个实体,实体集 A 中也有 $m(m \geq 0)$ 个实体与其联系,则称实体集 A 与实体集 B 的联系为多对多联系。记为 m:n。例如在一个学校中,存在有“课

程”和“学生”2个实体集,从“课程”实体集中任取一门课程实体,在“学生”实体集中一定能找到多名学生实体与其联系(多名学生选修过该门课程);反之,从“学生”实体集中任取一名学生实体,在“课程”实体集中也一定能找到多门课程实体与其联系(该名学生选修过多门课程),所以“课程”和“学生”2个实体集属于多对多联系。

3. 实体的联系模型

表示概念模型最常用的方法是实体——联系方法(Entity-Relationship Approach),该方法是用 E-R 图来描述概念模型中实体的相互联系,因此这种模型也称为实体联系模型(E-R 模型)。

在 E-R 图中,实体用“矩形框”表示,属性用“椭圆框”表示,联系为“菱形框”表示,再用线段将各部分连接起来并注明联系的类型,若实体之间联系也具有属性,则把属性和菱形也用线段连接上。如图 1-3 所示为用 E-R 图表示的学生选课实体联系 E-R 模型。

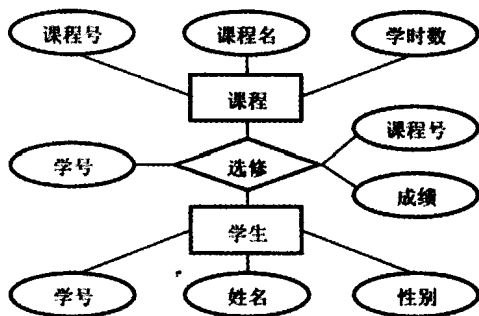


图 1-3 学生选课实体联系 E-R 模型

E-R 模型是独立于计算机系统的模型,简单、清晰,易于用户理解,是面向用户,而不是面向计算机的。现有的数据库系统没有一个能直接接受 E-R 模型,因为 E-R 模型只能说明实体以及实体间语义的联系,还不能进一步说明详细的数据结构。但在解决实际问题时,通常总是先设计一个 E-R 模型,然后再把 E-R 模型转换成计算机能实现的结构数据模型。

1.3.2 结构数据模型

结构数据模型是直接面向数据库的逻辑结构,它是现实世界的第二层抽象。这类模型有严格的形式化定义,以便于在计算机系统中实现。常用的结构数据模型主要有层次、网状和关系 3 种类型。

1. 层次模型

层次模型采用树状结构表示实体及其联系,适合于补救实体之间 1:n 联系。其结构自顶向下层次分明,如一个学校行政机构可以成为一个层次模型。对层次模型的数据搜索仅允许自上向下的单向搜索,故使得层次模型中数据操纵受到很大限制。1969 年,IBM 公司推出的 IMS 系统就是最典型的层次模型系统。

2. 网状模型

网状模型采用结点间的连通图(网状结构)表示实体及其联系,能表示实体之间各种复杂联系情况。网状模型是美国 DASYL 委员会数据库任务组(DBTG)于 1969 年提出的一种模型,在网状模型中,对数据的搜索可以用 2 种方式:

- (1) 可以从网络中任一点开始搜索。
- (2) 可沿着网中的路径按任意方向搜索,但在计算机中实现较为困难,使用也不太方便。

3. 关系模型

关系模型采用“二维表”表示实体及其联系,能直接表示实体之间各种复杂联系情况。