

高等院校“十一五”规划教材

Visual FoxPro

数据库程序设计教程

主 编 王凤领
副主编 王万学 王国锋 于海霞
主 审 戴宗荫



中国水利水电出版社
www.waterpub.com.cn

高等院校“十一五”规划教材

Visual FoxPro 数据库程序设计教程

主 编 王凤领

副主编 王万学 王国锋 于海霞

主 审 戴宗荫



中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

本书以 Visual FoxPro 6.0 中文版为平台,结合普通高校非计算机专业数据库程序设计课程的具体要求,深入浅出地介绍 Visual FoxPro 数据库程序设计的相关知识、方法和具体的实例。本书共 12 章。分别介绍数据库系统及 Visual FoxPro 概述、项目管理器及其操作、数据表的基本操作、数据库的设计与操作、程序设计基础、面向对象程序设计、表单的建立与使用、结构化查询语言——SQL、查询与视图、报表设计、菜单设计、应用程序的生成和发布等。

本书突出案例教学并有配套《Visual FoxPro 数据库程序设计习题解答与实验指导》教材。在理论讲解过程中,配有大量实例,通过对一个个实例的分析和操作,使读者在理解所学知识的基础上,掌握数据库应用系统的开发方法。每章均附有丰富的习题与上机操作题供读者练习,并在配套的实验教材中提供了各章习题的参考答案与上机指导,以及全国计算机等级考试笔试题和上机操作题。

本书既可作为普通高等院校各专业(含高职、高专)计算机公共课数据库应用课程的教材,也可作为计算机等级考试培训教材,或供从事数据库开发的人员学习、参考。

本书配有免费电子教案可以从中国水利水电出版社网站下载,网址为:
<http://www.waterpub.com.cn/softdown/>。

图书在版编目(CIP)数据

Visual FoxPro 数据库程序设计教程/王凤领主编.

北京:中国水利水电出版社,2008

高等院校“十一五”规划教材

ISBN 978-7-5084-5403-0

I. V… II. 王… 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(2008)第 075287 号

书 名	Visual FoxPro 数据库程序设计教程
作 者	主 编 王凤领 副主编 王万学 王国锋 于海霞 主 审 戴宗荫
出版 发行	中国水利水电出版社(北京市三里河路 6 号 100044) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn 电话: (010) 63202266 (总机)、68331835 (营销中心)、82562819 (万水)
经 售	全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京蓝空印刷厂
规 格	787mm×1092mm 16 开本 18.75 印张 512 千字
版 次	2008 年 6 月第 1 版 2008 年 6 月第 1 次印刷
印 数	0001—4000 册
定 价	29.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换

版权所有·侵权必究

前 言

本书以 Visual FoxPro 6.0 中文版为平台,结合普通高校非计算机专业数据库程序设计课程的具体要求,深入浅出地介绍 Visual FoxPro 数据库程序设计的有关知识、方法和具体的实例。本教材共 12 章。分别介绍数据库系统及 Visual FoxPro 概述、项目管理器及其操作、数据表的基本操作、数据库的设计与操作、程序设计基础、面向对象程序设计、表单的建立与使用、结构化查询语言——SQL、查询与视图、报表设计、菜单设计、应用程序的生成和发布等。

本书的作者多年来一直从事计算机基础教学,根据多年的教学实践编写了这本教材。针对初学者和自学读者的特点,本书力求通俗易懂,并用大量具体的操作、各种不同的实例让读者进入 Visual FoxPro 的可视化编程环境。所有步骤都按实际操作界面一步一步地讲解,读者可一边学习,一边上机操作。通过一段时间的练习,在不知不觉之中就可逐渐掌握 Visual FoxPro 6.0 程序设计的基础知识、设计思想和方法,以及可视化编程的方法和步骤,并有助于读者提高利用 Visual FoxPro 6.0 解决实际问题的能力。

本书突出案例教学并配有《Visual FoxPro 数据库程序设计习题解答与实验指导》实验教材。本书在理论讲解过程中,配有大量实例,通过一个个实例的分析和操作,使读者在理解所学知识的基础上,掌握数据库应用系统的开发方法。各章后均附有本章小结以及丰富的习题与上机操作题供读者练习,并在配套的实验教材中对各章习题提供了参考答案与上机指导以及全国计算机等级考试笔试题和上机操作题。本书可作为非计算机专业本科生、高职和高专学生的数据库公共课教材,还可作为计算机等级考试培训辅导教材,以及数据库应用系统开发人员参考用书。

本书由王凤领担任主编,王万学、王国锋、于海霞担任副主编,由戴宗荫教授主审,参编人员有邢婷、孙志强、王凤国、唐友、单晓光、丁康健、李钰、文雪巍、刘胜达等。最后由王凤领统稿并定稿完成。

在本书的编写过程中,得到了哈尔滨商业大学德强商务学院院长领导、教务处副处长庾莉及计算机科学系办公室主任刘玉等的大力支持与帮助,在此一并表示感谢!

由于时间紧迫,编者水平所限,书中错误和疏漏之处在所难免,敬请各位读者朋友在使用中给予批评指正,并请各位老师和专家不吝赐教。

编者的电子信箱为: wfl232983@163.com

作者

2008 年 4 月

目 录

前言

第 1 章 数据库系统及 Visual FoxPro 6.0 概述..... 1

本章要点..... 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..... 2

1.2 计算机数据管理的发展..... 2

1.2.1 人工管理阶段..... 2

1.2.2 文件系统阶段..... 3

1.2.3 数据库系统阶段..... 3

1.3 数据库系统基本概念..... 4

1.3.1 数据库..... 4

1.3.2 数据库管理系统..... 4

1.3.3 数据库系统..... 5

1.3.4 数据库应用系统..... 6

1.3.5 数据库系统的数据模式..... 6

1.4 数据模型..... 7

1.4.1 数据模型概述..... 7

1.4.2 E-R 数据模型..... 7

1.4.3 几种主要数据模型..... 8

1.5 关系数据库概述..... 10

1.5.1 关系术语..... 10

1.5.2 关系的规范化..... 11

1.5.3 关系运算..... 12

1.5.4 关系数据库..... 12

1.5.5 关系的完整性..... 12

1.5.6 数据库新技术概述..... 13

1.6 Visual FoxPro 6.0 及其界面..... 14

1.6.1 Visual FoxPro 6.0 概述..... 14

1.6.2 Visual FoxPro 6.0 界面..... 15

1.6.3 Visual FoxPro 向导、设计器、

生成器简介..... 15

1.7 Visual FoxPro 6.0 语法基础..... 16

1.7.1 数据类型..... 17

1.7.2 常量..... 17

1.7.3 变量..... 19

1.7.4 数组..... 20

1.7.5 常用函数..... 22

1.7.6 表达式..... 38

1.7.7 命令..... 43

本章小结..... 44

习题 1..... 44

第 2 章 项目管理器及其操作..... 52

本章要点..... 52

2.1 项目管理器的基本概念..... 52

2.1.1 创建项目..... 52

2.1.2 项目管理器的使用..... 53

2.2 项目管理器的操作..... 55

2.2.1 查看项目中的内容..... 55

2.2.2 添加或移去文件..... 55

2.2.3 创建和修改文件..... 56

2.2.4 定制项目管理器..... 56

本章小结..... 56

习题 2..... 57

第 3 章 数据表的基本操作..... 58

本章要点..... 58

3.1 数据表概述..... 58

3.1.1 表的相关概念..... 58

3.1.2 表中的数据类型..... 59

3.2 创建新表..... 59

3.2.1 设计表结构..... 60

3.2.2 通过表设计器创建新表..... 60

3.2.3 通过表向导创建新表..... 62

3.2.4 输入记录与浏览表中的信息..... 65

3.3 浏览和编辑表..... 66

3.3.1 修改已有表的结构..... 66

3.3.2 添加新记录..... 67

3.3.3 删除记录	67	4.1 数据库的基本操作	86
3.3.4 在表中移动记录指针	67	4.1.1 数据库设计的一般步骤	86
3.3.5 定制浏览窗口	68	4.1.2 创建新数据库	87
3.4 表的索引与排序	69	4.1.3 在数据库中加入表和移去表	88
3.4.1 索引的类型	69	4.1.4 多表间关联	90
3.4.2 各种类型索引的使用	70	4.1.5 表的连接	93
3.4.3 用多个字段进行索引	70	4.2 设置数据库	93
3.4.4 排序	71	4.2.1 设置字段显示属性	93
3.5 表的数值计算	71	4.2.2 设置字段输入默认值	95
3.5.1 纵向求和 SUM	71	4.2.3 定义字段有效性规则	95
3.5.2 纵向求平均值 AVERAGE	72	4.2.4 设置永久关系与参照完整性	95
3.5.3 统计记录数 COUNT	72	4.3 数据库的操作	98
3.6 多表的操作	73	4.3.1 打开/关闭数据库	98
3.6.1 工作区的概念	73	4.3.2 在项目中添加/移去数据库	99
3.6.2 选择工作区的命令	73	4.3.3 建立表间关系	100
3.6.3 使用 USE 命令指定工作区 打开表	74	4.3.4 使用多个数据库	101
3.6.4 建立表的关联	74	本章小结	101
3.6.5 解除关联	75	习题 4	102
3.7 用命令对表进行操作	75	第 5 章 面向对象的程序设计	107
3.7.1 打开表命令	75	本章要点	107
3.7.2 关闭表命令	76	5.1 面向对象程序设计	107
3.7.3 显示表记录的命令	76	5.1.1 对象	107
3.7.4 利用已有的表建立新表	77	5.1.2 对象的属性、事件与方法	108
3.7.5 修改表结构的命令	77	5.1.3 面向对象编程	109
3.7.6 记录定位命令	77	5.1.4 对象编程实例	109
3.7.7 记录的删除命令	78	5.2 Visual FoxPro 中的类	111
3.7.8 替换 REPLACE 命令	79	5.2.1 Visual FoxPro 的基类	111
3.7.9 条件查询 LOCATE 命令	79	5.2.2 类的特性	113
3.7.10 建立单索引文件的命令	79	5.3 使用对象	113
3.7.11 建立复合索引文件的命令	80	5.3.1 对象的包容层次	113
3.7.12 追加记录 APPEND 命令	81	5.3.2 对象的引用	114
3.7.13 插入记录 INSERT 命令	81	5.3.3 设置属性	115
3.7.14 从其表文件中追加 数据 APPEND FROM 命令	81	5.4 简单的输入输出程序设计	116
本章小结	81	5.4.1 命令按钮的使用	116
习题 3	82	5.4.2 用标签输出信息	116
第 4 章 数据库的设计与操作	86	5.4.3 文本框输入输出	117
本章要点	86	5.5 Visual FoxPro 中的事件与方法程序	119
		5.6 用户自定义类	120
		5.6.1 使用类设计器创建类	121

5.6.2 编程方式使用类	123	7.1.3 程序中常用的命令	165
本章小结	124	7.2 顺序结构程序设计	166
习题 5	124	7.3 选择结构程序设计	168
第 6 章 表单的建立与使用	127	7.3.1 简单分支结构	168
本章要点	127	7.3.2 选择分支结构	169
6.1 创建表单	127	7.3.3 多分支结构	170
6.2 表单设计器	129	7.4 循环结构程序设计	172
6.2.1 “表单设计器”工具栏	129	7.4.1 条件循环语句	
6.2.2 “表单控件”工具栏	129	DO WHILE.....ENDDO	172
6.2.3 “属性”窗口	130	7.4.2 计数循环语句 FOR ENDFOR	174
6.2.4 “代码”窗口	132	7.4.3 数据表扫描循环语句	
6.3 表单的数据环境	132	SCAN ENDSCAN	175
6.3.1 数据环境设计器	132	7.5 多重循环	177
6.3.2 添加、移去表或视图	133	7.6 多模块程序设计	179
6.4 表单的保存、运行和修改	134	7.6.1 过程及其过程调用	179
6.4.1 保存表单	134	7.6.2 用户自定义函数	184
6.4.2 运行表单的多种方法	134	7.6.3 自定义方法	185
6.4.3 修改表单	134	7.6.4 变量作用域	187
6.5 表单常用控件	134	7.7 程序的调试	188
6.5.1 标签、文本框和命令按钮	134	7.7.1 调试器环境	188
6.5.2 线条、形状和图像控件	138	7.7.2 设置断点	189
6.5.3 列表框	140	7.7.3 调试菜单	190
6.5.4 容器控件	141	本章小结	190
6.5.5 选项按钮组	142	习题 7	191
6.5.6 复选框	143	第 8 章 结构化查询语言——SQL	197
6.5.7 命令按钮组	144	本章要点	197
6.5.8 计时器控件	145	8.1 SQL 语言概述	197
6.5.9 微调按钮	146	8.2 SQL 的数据查询功能	198
6.5.10 组合框	147	8.2.1 SELECT 语句格式与功能	198
6.5.11 表格控件	148	8.2.2 应用举例	204
6.5.12 页框	149	8.3 SQL 的数据操作功能	206
6.5.13 编辑框	150	8.3.1 插入记录	206
本章小结	151	8.3.2 更新记录	207
习题 6	151	8.3.3 删除记录命令	207
第 7 章 程序设计基础	162	8.4 SQL 的数据定义功能	208
本章要点	162	8.4.1 定义表	208
7.1 程序概述	162	8.4.2 修改表的结构	209
7.1.1 程序的建立、编辑	163	8.4.3 删除表	209
7.1.2 程序的运行	164	8.4.4 建立视图	210

本章小结.....	210	第 11 章 报表设计.....	259
习题 8.....	210	本章要点.....	259
第 9 章 查询与视图	217	11.1 计划报表布局.....	259
本章要点.....	217	11.1.1 报表的常规布局	259
9.1 查询与视图概述.....	217	11.1.2 报表布局文件	260
9.1.1 查询的概念.....	217	11.2 创建报表布局.....	260
9.1.2 视图的概念.....	217	11.2.1 通过“报表向导”创建报表	260
9.1.3 视图与查询比较	217	11.2.2 快速报表	263
9.2 创建查询.....	218	11.2.3 通过“报表设计器”创建报表 ..	264
9.2.1 通过查询向导建立查询	218	11.3 修改报表布局.....	265
9.2.2 通过查询设计器创建查询	221	11.3.1 规划数据的位置	266
9.2.3 查询去向	225	11.3.2 调整报表带区的大小和布局.....	266
9.3 运行查询.....	228	11.3.3 设置报表数据源	267
9.4 创建视图.....	229	11.3.4 增添报表控件	267
9.4.1 通过视图向导建立视图	229	11.4 预览和打印报表.....	270
9.4.2 通过视图设计器建立视图	232	11.4.1 预览结果	271
9.5 利用视图更新表.....	237	11.4.2 打印报表	271
本章小结.....	237	本章小结.....	272
习题 9.....	238	习题 11.....	272
第 10 章 菜单设计	242	第 12 章 应用程序的生成和发布	275
本章要点.....	242	本章要点.....	275
10.1 菜单系统概述.....	242	12.1 建立帮助文件.....	275
10.1.1 菜单系统的基本结构	242	12.1.1 建立 HTML 帮助.....	275
10.1.2 菜单系统的设计步骤	243	12.1.2 建立图形方式的帮助	276
10.2 创建快速菜单.....	244	12.1.3 设计 .DBF 帮助	276
10.3 使用菜单设计器创建菜单	246	12.2 编译应用程序.....	277
10.3.1 创建主菜单	246	12.2.1 建立项目	277
10.3.2 创建子菜单项	247	12.2.2 建立主控文件	277
10.3.3 定义菜单项功能	247	12.2.3 在项目中运行应用程序	278
10.3.4 定义快捷键	248	12.2.4 项目的连编	278
10.3.5 添加系统菜单项	249	12.3 安装向导.....	279
10.3.6 菜单项的相关设计	250	12.3.1 发布树	279
10.3.7 显示菜单中选项设置	251	12.3.2 运行安装向导	281
10.4 在顶层表单中设计菜单	253	12.4 系统开发实例.....	281
10.5 创建快捷菜单.....	254	本章小结.....	289
本章小结.....	256	习题 12.....	290
习题 10.....	256	参考文献.....	292

第 1 章 数据库系统及 Visual FoxPro 6.0 概述

本章要点

介绍与数据库有关的基本概念和知识,包括数据、信息和数据处理、数据管理技术的发展、数据库系统、数据模型以及关系数据库等。Visual FoxPro 的语法基础包括常量、字段变量、内存变量、函数、表达式的概念及其使用规则等。Visual FoxPro 作为一门数据库编程语言,和其他编程语言一样,编写的程序都是由常量、变量、函数、表达式等基本语法组成的。而作为一门数据库语言,Visual FoxPro 突出了数据库管理的特点,其中一些数据元素不仅能在程序中使用,也被包含在数据库文件中。这些基本概念和知识是学习和使用 Visual FoxPro 的基础。

近年来,随着数据库技术和网络技术相互结合和渗透,已成为计算机技术发展最迅速和应用最广泛的两大领域。对于在日常生活、生产经营、金融证券、事务管理等活动中产生的大量数据,数据库管理系统以数据库的方式进行组织和存储,并编写数据库应用程序以实现数据的共享和高效处理,从而满足人们对数据管理的各种需要。目前流行的开发数据库系统的编程语言有 Visual FoxPro、SQL、Delphi、Oracle、Java 等,它们各有其功能和特点。本书介绍的 Visual FoxPro (缩写为 VFP) 数据库语言由于具有简单易学、使用方便、开发成本低等特点,在我国有着广泛的应用基础,适合初学者用来掌握数据库语言的基本结构和特点,并很容易通过它来开发一些实用的中小型数据库系统。本章介绍数据与数据处理的基本概念,数据库的概念、相关术语、基本运算及 Visual FoxPro 的语法基础等,为进一步学习和应用 Visual FoxPro 数据库语言打好基础。

1.1 数据、信息与数据处理

1.1.1 数据与信息

1. 数据

数据是客观事物属性的取值,是信息的具体描述和表现形式,是信息的载体。在计算机系统中,凡是能为计算机所接受和处理的各种字符、数字、图形、图像及声音等都可称为数据。因此,数据泛指一切可被计算机接受和处理的符号。数据可分为数值型数据(如工资、成绩等)和非数值型数据(如姓名、日期、声音、图形、图像等)。数据可以被收集、存储、处理(加工、分类、计算等)、传播和使用。

2. 信息

信息是事物状态及运动方式的反映(表现形式),需经过加工、处理后才能进行交流和使用的。人们往往用数据去记载、描述和传播信息,因此数据是描述或表达信息的具体表现形式,是信息的载体。

信息与数据既有联系又有区别,它们之间的关系可描述为:信息是对客观现实世界的反

映,数据是信息的具体表现形式。注意,可以用不同的数据形式表示同样的信息,信息不随它的数据形式的不同而改变。例如,某个部门要召开会议,可以把“开会”这样一个信息通过广播(声音形式的数据)、文件(文字形式的数据)等方式通知给有关单位,在这里,声音或文字是不同的反映方式(表现形式),可以表示同一个信息。

1.1.2 数据处理

数据处理也称为信息处理。所谓数据处理,是指利用计算机将各种类型的数据转换成信息的过程。它包括对数据的采集、整理、存储、分类、排序、加工、检索、维护、统计和传输等一系列处理过程。数据处理的目的是从大量的、原始的数据中获得人们所需要的资料并提取有用的数据成分,从而为人们的工作和决策提供必要的数据基础和决策依据。

在叙述了数据、信息和数据处理的概念之后,这里简单介绍一下它们之间的联系。首先,信息和数据是有区别的。数据是一种符号象征,它本身是没有意义的,而信息是有意义的知识。但数据经过加工处理就能成为有意义的信息,也就是说,数据处理把数据和信息联系在了一起。下式可以简单明确地表明三者的关系:

$$\text{信息} = \text{数据} + \text{数据处理}$$

再举例说明,如计算机中日期数据的一种符号表示为“04/12/82”,不加以解释就不知道它明确的意义,究竟是人的出生日期还是商品的销售日期。通过以后的学习可以知道,在数据库中可以给它一个标识,解释这是一个人的出生日期,再用当前日期减去这个出生日期来进行数据处理,就可以获得这个人年龄的信息。

1.2 计算机数据管理的发展

数据处理的内容首先是数据的管理。计算机发明以后,人们一直在努力寻求如何用计算机更有效地管理数据。随着计算机硬件和软件技术的发展,计算机数据管理技术也经历了从低级阶段发展到高级阶段的过程,技术上也越来越成熟。按照一般文献划分,计算机数据管理的发展有如下几个阶段。

1.2.1 人工管理阶段

20世纪50年代是第一代计算机应用阶段。当时,计算机没有磁盘这样的能长期保存数据的存储设备,这个时期的数据管理是用人工方式把数据保存在卡片、纸带这类的介质上,所以称为人工管理阶段。这个阶段数据管理的最大特征是数据由计算数据的程序携带,二者混合在一起,因此具有以下特点。

1. 数据不能独立

由于数据和程序混合在一起,这样就不能处理大量的数据,更谈不上数据的独立与共享,一组数据只能被一个程序专用。此外,当程序中的数据类型、格式发生变化时,相应程序也必须进行修改。

2. 数据不能长期保存

这个阶段计算机的主要任务是科学计算。计算机运行时,程序和数据在计算机中,程序运行结束后,数据即从计算机中释放出来。

3. 数据没有专门的管理软件

由于计算机系统没有数据管理软件管理数据,也就没有数据的统一存取规则。数据的存

取、输入输出方式由编写程序的程序员自己确定，这就增加了程序编写的负担。

1.2.2 文件系统阶段

随着计算机对数据处理需求的不断增加，人们对数据处理的重要性越来越重视。20 世纪 50 年代末至 60 年代，计算机操作系统中专门设置了文件系统来管理数据，计算机的数据管理进入了文件系统阶段。这个阶段的主要特征是数据文件和处理数据的程序文件分离，数据文件由文件系统管理，它确立数据文件和程序文件的接口，保证文件能被正确地调用。与人工阶段相比，文件系统阶段是有所进步，但还是存在以下缺点。

1. 数据独立性差，不能共享数据

虽然从程序文件中分离了出来，但文件系统管理的数据文件只能简单地存放数据，且一个数据文件一般只能被相应的程序文件专用，相同的数据要被另外的程序使用，必须再产生数据文件，这样就出现了数据的重复存储问题，即数据冗余。

2. 数据文件不能集中管理

由于这阶段的数据文件没有合理和规范的结构，数据文件之间不能建立联系，使得数据文件不能集中管理，数据使用的安全性和完整性都得不到保证。

1.2.3 数据库系统阶段

20 世纪 60 年代末，计算机的数据管理进入数据库系统阶段。这时，由于计算机的数据处理量迅速增长，其数据管理得到了人们的高度重视，随后在美国产生了技术成熟、具有商业价值的数据库管理系统。数据库系统不仅有效地实现了程序和数据的分离，而且把大量的数据组织在一种特定结构的数据库文件中，多个不同程序都可以调用数据库中相同的数据，从而实现了数据的统一管理及数据共享。与文件系统相比，数据库系统具有以下特点。

1. 实现数据共享，减少数据冗余度

由于数据库文件不仅与程序文件相互独立，而且具有合理规范的结构，使得不同程序可以同时使用数据库中相同的数据，这样就大大节省了存储资源，减少了数据的冗余度。

2. 实现数据独立

数据独立包括物理数据独立和逻辑数据独立。物理数据是指数据在硬件上的存储形式，其独立性是指当数据的存储结构发生变化时，不会影响数据的逻辑结构，也就不会影响程序的运行。逻辑数据是指数据在用户面前的表现形式，当逻辑数据结构发生变化时也不影响应用程序，这就是逻辑数据的独立性。这两种数据的独立性有效地保证了数据库运行的稳定性。

3. 采用合理的数据结构加强了数据的联系

数据库采用合理的结构来安排其中的数据，不仅同一数据文件中的数据之间存在特定的联系，各数据文件之间也可以建立关系，这是文件系统不能做到的。

4. 加强数据保护

与文件系统相比，数据库系统增加了数据的多种控制功能，如并发控制能保证多个用户同时使用数据时不产生冲突；安全性控制能保证数据的安全，不被非法用户使用和破坏；数据的完整性控制保证了数据使用过程中的正确性和有效性。

值得一提的是，有的文献又把数据库系统阶段分为集中式数据库系统阶段和分布式数据库系统阶段。早期的数据库系统是集中式的，其特点是把所有的数据，无论在物理上还是在逻辑上，都集中摆放在一起。这样虽然设计简单，但影响数据的流通速度。

随着计算机网络技术的高速发展，现在更多的数据库系统采用了分布式数据库系统。通

过网络技术把分布在各处的计算机连接起来,数据库中的数据在物理上分布于网络中不同计算机结点上。但对用户使用来说,他不知道也不用关心数据存放在哪个地方,逻辑上看起来好像是在集中使用。分布式数据库系统提高了数据的使用效率,加快了数据的流通速度,更加符合今天人们对数据处理的需要。

关于分布式数据库系统的网络工作模式,现在使用较多的是客户机/服务器模式。在这种模式中,数据及数据处理程序放在数据服务器上,业务处理程序 and 用户界面放在客户机上。客户机/服务器模式数据库系统的结构图如图 1-1 所示。Visual FoxPro 数据库管理系统支持这种模式,并为开发功能强大的客户机/服务器模式的应用程序提供了专门的工具。

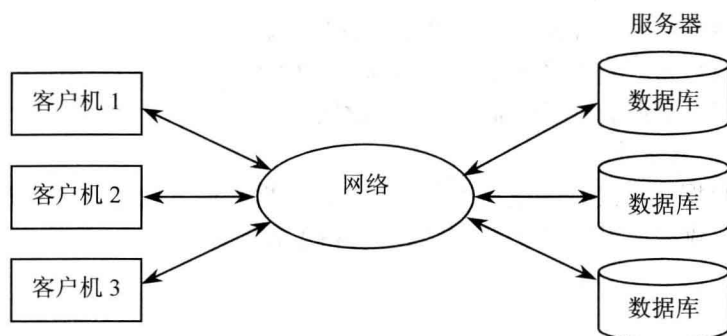


图 1-1 客户机/服务器数据库系统结构图

1.3 数据库系统基本概念

在数据库技术中,人们常常接触到数据库、数据库管理系统、数据库系统、数据库应用系统这些名词,它们之间有着一定的联系和区别。

1.3.1 数据库

数据库(DataBase, DB)就是按一定的组织形式存储在一起的相互关联的数据的集合。实际上,数据库就是一个存放大量业务数据的场所,其中的数据具有特定的组织结构。所谓“组织结构”,是指数据库中的数据不是分散的、孤立的,而是按照某种数据模型组织起来的,不仅数据记录内的数据之间是彼此相关的,而且数据记录之间在结构上也是有机联系在一起的。数据库具有数据的结构化、独立性、共享性、冗余量小、安全性、完整性和并发控制等基本特点。在数据库系统中,数据库已成为各类管理系统的核心基础,为用户和应用程序提供了共享的资源。

1.3.2 数据库管理系统

数据库管理系统(DataBase Management System, DBMS)是一种负责数据库的定义、建立、操纵、管理和维护的计算机软件,是数据库系统的核心部分。数据库管理系统是在特定操作系统的支持下工作的,它提供了对数据库资源进行统一管理和控制的功能,使数据结构和数据存储具有一定的规范性,提高了数据库应用的简明性和方便性。DBMS 是一种系统软件,也就是数据库语言本身,常用的有 Visual FoxPro(简称 VFP)、SQL、Oracle 等数据库语言。

DBMS 为用户管理数据提供了一整套命令, 利用这些命令可以实现对数据库的各种操作, 如数据结构的定义, 数据的输入、输出、编辑、删除、更新、统计和浏览等。具体可归纳为以下四大功能。

1. 数据定义功能

数据库管理系统定义和描述数据库的结构使用数据库定义语言 (Data Description Language, DDL), 这就需要用相应的解释和编译程序来实现该功能, 如 Visual FoxPro 数据库管理系统中的 CREATE STRUCTURE, 该命令可创建一个数据库并设计数据库中数据的结构。

2. 数据操作功能

DBMS 提供的数据操纵语言 (Data Manipulation Language, DML) 用于实现数据的追加、插入、修改、输出、检索等功能。不同的数据库语言提供的功能命令的格式不同, 但这些功能对数据库管理来说是最基本的, 是构成应用程序必不可少的元素。

3. 数据控制功能

为保障数据库中数据使用的安全性和可靠性, DBMS 要提供一定的手段保护数据, 这就是数据控制的概念, 包括: 数据完整性控制、并发控制、安全性控制和数据恢复控制等。

4. 数据字典 (Data Dictionary)

数据字典是以数据文件的方式存放关于数据库的结构描述和说明信息, 是一种特殊的数据库。软件开发人员可以通过查阅数据字典来方便地使用和操作数据库, 这对数据量大的应用程序是很有帮助的。大型数据库管理系统有专门创建数据字典的功能, 而 Visual FoxPro 则需较多的人工操作才能创建数据字典。

1.3.3 数据库系统

数据库系统 (DataBase System, DBS) 是在计算机系统中引入数据库后的系统构成, 它是一个具有数据库管理功能的计算机软硬件综合系统。具体地说, 它主要包括计算机硬件、操作系统、数据库 (DB)、数据库管理系统 (DBMS) 和相关软件、数据库管理员及用户等组成部分。数据库系统具有数据的结构化、共享性、独立性、可控冗余度, 以及数据的安全性、完整性和并发控制等特点。

- 硬件系统: 是数据库系统的物理支持, 包括主机、外部存储器、输入/输出设备等。
- 软件系统: 包括系统软件和应用软件。系统软件包括支持数据库管理系统运行的操作系统 (如 Windows 2000)、数据库管理系统 (如 Visual FoxPro 6.0)、开发应用系统的高级语言及其编译系统等; 应用软件是指在数据库管理系统基础上, 用户根据实际问题自行开发的应用程序。
- 数据库: 是数据库系统的管理对象, 为用户提供数据的信息源。
- 数据库管理员 (DBA): 是负责管理和控制数据库系统的主要维护管理人员。
- 用户: 是数据库的使用者, 利用数据库管理系统软件提供的命令访问数据库并进行各种操作。用户包括专业用户和最终用户。专业用户即程序员, 是负责开发应用程序的设计人员。最终用户是对数据库进行查询或通过数据库应用系统提供的界面使用数据库的人员。

数据库、数据库管理系统和用户应用程序是构成数据库系统的三要素。三者之间的关系是: 用户为了有效地处理和使用数据而建立数据库, 数据库管理系统是数据库的管理者, 它是一个软件, 其职能是维护数据库中的数据, 响应和完成用户应用程序或命令提出的访问数据的各种请求。数据库系统三要素之间的关系如图 1-2 所示。

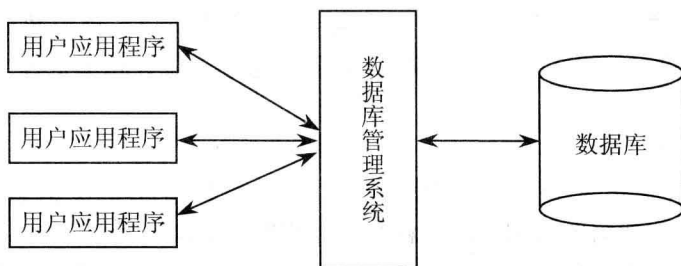


图 1-2 数据库系统三要素之间的关系

1.3.4 数据库应用系统

数据库应用系统 (DataBase Application System, DBAS) 是在 DBMS 支持下根据实际问题开发出来的数据库应用软件。一个 DBAS 通常由数据库和应用程序两部分组成, 它们都需要在 DBMS 支持下开发。

由于数据库的数据要供不同的应用程序共享, 因此在设计应用程序之前首先要对数据库进行设计。数据库的设计是以“关系规范化”理论为指导, 按照实际应用的报表数据, 首先定义数据的结构, 包括逻辑结构和物理结构, 然后输入数据形成数据库。开发应用程序也可采用功能分析、总体设计、模块设计、编码调试等步骤来实现。

1.3.5 数据库系统的数据模式

从数据库管理系统的角度来看, 数据库系统可分为三级模式, 从外到内依次为外模式、模式和内模式。

1. 外模式

外模式又称子模式或用户模式, 它是数据库用户和数据库系统的接口, 是数据库用户看到的数据视图, 是对数据库中局部数据的逻辑结构和特征的描述, 是与某一应用有关的数据的逻辑表示。外模式通常是模式的子集。一个数据库可以有多个外模式。同一个外模式可以被某一个用户的多个应用程序所使用, 但一个应用程序只有一个外模式。

2. 模式

模式也称逻辑模式或概念模式, 它是对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。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。

模式是在数据库模式结构的中间层中, 既不涉及数据的物理存储细节和硬件环境, 也与具体的应用程序、应用开发工具以及高级程序设计语言无关。DBMS 提供数据定义语言 (DDL) 来描述模式。定义模式时要定义数据的逻辑结构, 包括记录由哪些数据项构成; 数据项的名称、类型、取值范围; 数据之间的联系; 与数据有关的安全性、完整性要求等。

3. 内模式

内模式又称为存储模式, 它是对数据库物理结构和存储方式的描述, 是数据在数据库内部的表示方式。它规定了数据在存储介质上的物理组织方式, 记录了寻址技术、物理存储块的大小、溢出处理方法等。一个数据库只有一个内模式。

为了实现三级模式的联系和转换, 数据库管理系统在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。映像是一种对应规则, 指出映像双方应如何进行转换。数据库的三级模式通过这两层映像连接起来, 从而为各类用户提供操纵数据库的手段。

- 外模式/模式映像：定义外模式与模式之间的对应关系。当数据库的全局逻辑结构改变时，只需要修改外模式与模式之间的对应关系，而不必修改局部逻辑结构，即保证外模式不变，相应的应用程序也不必修改，从而实现数据和程序的逻辑独立性。
- 模式/内模式映像：定义数据全局逻辑结构与存储结构之间的对应关系。当数据库的物理存储结构改变时，只需要修改模式与内模式之间的对应关系，即可保持模式不变，从而实现数据和程序的物理独立性。

1.4 数据模型

数据模型是对现实世界数据特征的抽象，是用来描述数据的一组概念和定义。数据模型按不同的应用层次可划分为概念数据模型和逻辑数据模型两类。概念数据模型又称为概念模型，是一种面向客观世界、面向用户的模型，主要用于数据库的设计。而逻辑数据模型常称为数据模型，是一种面向计算机系统的模型，主要用于实现数据库管理系统。

1.4.1 数据模型概述

数据模型是对现实世界数据特征的抽象，是用来描述数据的结构和联系的一组概念和定义，是数据库的核心内容。

由于计算机不能直接处理现实世界中的具体事物，所以必须把具体事物转换成计算机能够处理的数据。在数据库系统中，实现转换的过程通常是先把现实世界中的客观事物抽象为概念数据模型（简称概念模型），然后再把概念数据模型转换为某一数据库管理系统所支持的逻辑数据模型（简称数据模型）。

概念数据模型和逻辑数据模型是数据模型的不同应用层次。概念数据模型是从现实世界到数据世界的一个中间层次，是一种面向客观世界、面向用户的模型，是数据库设计人员进行数据库设计的重要工具，也是数据库设计人员和用户之间进行交流的语言，E-R 模型、扩充的 E-R 模型等是常用的概念模型。逻辑数据模型是一种面向数据库系统的模型，即依赖于某种具体的数据库管理系统（DBMS），主要用于 DBMS 的实现，常见的逻辑数据模型包括层次模型、网状模型和关系模型等。

1.4.2 E-R 数据模型

E-R 数据模型（Entity-Relationship Data Model，实体—联系数据模型）用来描述现实世界，具有直观、自然、语义丰富及便于向逻辑数据模型转换等优点。

设计 E-R 模型的目标是有效和自然地模拟现实世界，而不是关心它在计算机中如何实现，因此 E-R 模型中只应包含那些对描述现实世界具有普遍意义的抽象概念。E-R 模型中的基本概念有实体、联系、属性等。

1. 实体（entity）

客观存在并可相互区分的事物称为实体。它是信息世界的基本单位。实体既可以是人，也可以是物；既可以是实际对象，也可以是抽象对象；既可以是事物本身，也可以是事物与事物之间的联系。例如，一个学生、一个教师、一门课程、一支铅笔、一部电影、一个部门等都是实体。

同类型的实体的集合称为实体集（entity set）。例如，一个学校的全体学生是一个实体集，而其中的每个学生都是实体集的成员。

2. 联系 (relationship)

联系是实体集之间关系的抽象表示, 是对现实世界中事物之间关系的描述。例如, 公司实体集与职工实体集之间存在“聘任”联系。实体集之间的联系可分为以下 3 类。

(1) 一对一联系 (1:1): 如果对于实体集 A 中的每一个实体, 实体集 B 中至多有一个实体与之联系, 反之亦然, 则称实体集 A 与实体集 B 具有一对一联系。例如, 在一个学校中, 一个班级只有一个正班长, 而一个班长只在一个班中任职, 则班级与班长之间具有一对一联系。又如职工和工号的联系是一一对一的, 每一个职工只对应于一个工号, 不可能出现一个职工对应于多个工号或一个工号对应于多名职工的情况。

(2) 一对多联系 (1:n): 如果对于实体集 A 中的每一个实体, 实体集 B 中有 n 个实体 ($n > 0$) 与之联系, 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之联系, 则称实体集 A 与实体 B 有一对多联系。

例如系和学生两个实体集, 一个学生只能在一个系里注册, 而一个系可以有很多学生, 所以系和学生是一对多联系。又如单位的部门和职工的联系是一对多的, 一个部门对应于多名职工, 多名职工对应于同一个部门。

(3) 多对多联系 (m:n): 如果对于实体集 A 中的每一个实体, 实体集 B 中有 n 个实体 ($n > 0$) 与之联系, 反之, 对于实体集 B 中的每一个实体, 实体集 A 中也有 m 个实体 ($m > 0$) 与之联系, 则称实体集 A 与实体 B 具有多对多联系。例如, 一门课程同时有若干个学生选修, 而一个学生可以同时选修多门课程, 则课程与学生之间具有多对多联系。又如在单位中, 一个职工可以参加若干个项目的工作, 而一个项目又可有多个职工参加, 则职工与项目之间具有多对多联系。

3. 属性 (attribute)

描述实体的特性称为属性。一个实体可由若干个属性来刻画。属性的组合表征了实体。例如, 商品有商品代码、商品名称、单价、生产日期、进口否、商品外形等属性; 铅笔有商标、软硬度、颜色、价格、生产厂家等属性。

唯一标识实体的一个属性集称为码, 例如, 学号是学生实体的码。属性的取值范围称为域, 例如, 学生实体中, 性别属性的域为 (男, 女), 年龄的域可定为 18~60, 这里要注意区分属性的型与属性的值, 例如, 学生实体中的学号、姓名等属性名是属性的型, 而某个学生的“0001”、“张三”等具体数据则称为属性值。

相应地, 实体也有型和值之分, 实体的型用实体名及其属性名的集合来表示。例如, 学生以及学生的属性名集合构成学生实体的型, 可以简记为: 学生(学号, 姓名, 性别, 出生日期, 籍贯, 专业, 是否团员), 而("0001", "张三", "女", {^1982/03/12}, "成都", "信息", T.) 是一个实体值。实体集实际上就是同类型实体的集合, 例如, 全体学生就是一个实体集。

1.4.3 几种主要数据模型

数据库系统能减少数据冗余度, 实现数据共享和集中管理, 都是因为数据库中的数据有特定的组织结构, 这就是数据模型的概念。不同的数据库系统采用不同的数据模型, 具体可以分为如下 4 种。

1. 层次型数据库系统 (Hierarchical Model Database System)

层次型数据库是按着数据的从属关系来组织数据的, 类似于磁盘上的树型目录结构, 适合于存放和处理一个单位的组织结构、职工的隶属关系等数据。层次型数据库管理和检索类似的数据是非常方便的。层次模型如图 1-3 所示。

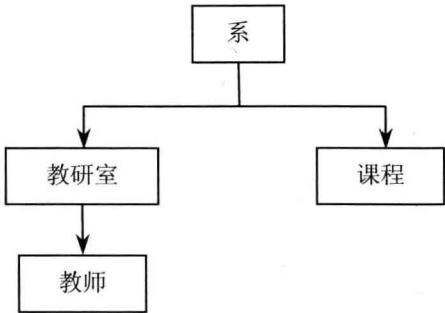


图 1-3 层次模型示意图

2. 网状型数据库系统 (Network Model Database System)

在网状模型的数据库中，一个数据实体与两个以上的其他数据实体存在联系。在这样的数据库中，除了要存储数据本身之外，还要存储指向所要联系数据的指针。这样，当检索到一个数据之后，就很容易检索到与之有联系的数据了。网状模型如图 1-4 所示。

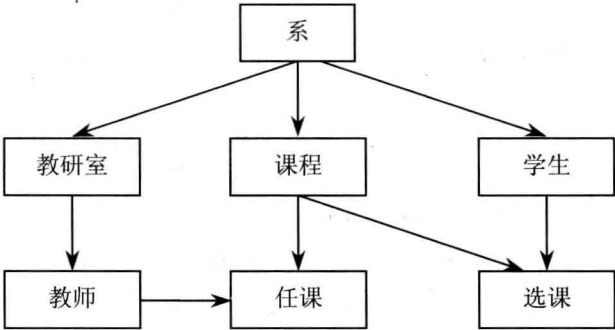


图 1-4 网状模型示意图

层次型和网状型属于非关系型数据库。它们均属于第一代数据库。

3. 关系型数据库系统 (Relational Database System, RDBS)

关系模型是一种以关系（二维表）的形式表示实体与实体之间联系的数据模型。关系模型不像层次模型和网状模型那样使用大量的链接指针把有关数据集合到一起，而是用一张二维表来描述一个关系。关系模型的主要特点如下：

- 关系中的每一分量不可再分，是最基本的数据单位。
- 关系中每一列的分量是同属性的，列数根据需要而设，且各列的顺序是任意的。
- 关系中每一行由一个具体实体或者联系的一个或多个属性构成，且各行的顺序是任意的。
- 一个关系是一张二维表，不允许有相同的列（属性），也不允许有相同的行（元组）。

表 1-1 所示的是一张学生登录表。在二维表中，每一行称为一个记录，用于表示一组数据项；表中的每一列称为一个字段或属性，用于表示每列中的数据项。表中的第一行称为字段名，用于表示每个字段的名称。

现在的数据库管理系统大多采用关系模型。关系模型中数据结点之间的联系是一对一的，每个实体只能各自和其前面一个及后面一个实体发生联系。关系模型以关系数学理论为基础，在数据库的设计和操作上就比前两种模型更为可靠和实用。本书讨论的 Visual FoxPro 数据库