

21世纪高校计算机教材

Visual FoxPro

面向对象 程序设计



第2版

实用教程

主 编

匡 松 刘 容

主 审

黄迪明

副主编

易 涛 何振林

邱俊山 黄 敏



西南交通大学 出版社

<http://press.swjtu.edu.cn>

Visual FoxPro

面向对象程序设计

实用教程 第2版

主 编: 匡 松 刘 容

主 审: 黄迪明

副主编: 易 涛 何振林

邱俊山 黄 敏

编 委: (以姓氏拼音字母为序)

陈 年	高仕忠	郭兴吉	蒋 斌	李明龙	梁金明
刘 毅	刘成安	刘家彬	刘玉萍	孟 丽	孟 林
宋 健	王 超	王 锦	王学平	伍治林	肖 甘
徐 鹰	杨 帆	杨海中	杨清平	张 平	张选芳
赵 亮	周 敏	周学文			

总策划: 星网科教

秘书处: 孙康江



西南交通大学出版社

<http://press.swjtu.edu.cn>

图书在版编目 (C I P) 数据

Visual FoxPro 面向对象程序设计实用教程 / 匡松,
刘容主编. — 2 版. — 成都: 西南交通大学出版社,
2006.7

ISBN 7-81104-373-4

I. V... II. ①匡...②刘... III. 关系数据库—数
据库管理系统, Visual FoxPro—程序设计—教材
IV. TP311.138

中国版本图书馆 CIP 数据核字 (2006) 第 077126 号

内 容 提 要

本教程按 Visual FoxPro 程序设计计算机等级 (二级) 考试大纲编写, 系统介绍了表的创建、排序和索引、数据库操作、SQL 语言、面向过程和面向对象的程序设计方法、表单设计、报表设计等内容; 并介绍了数据结构、软件工程、操作系统的软件基础内容。

本教程同时配套出版实验指导和等级考试习题集的《Visual FoxPro 面向对象程序设计上机和级考实训教程》(匡松 何振林主编, 西南交通大学出版社), 附有参考答案和重点解析。

本套教材既适宜作为省级计算机等级考试教材, 也适宜作为全国计算机等级考试教材。

Visual FoxPro 面向对象程序设计实用教程 (第二版)

主编 匡松 刘容

责任编辑
特约编辑

王 婷 唐 晴
孙康江

出版发行

西南交通大学出版社
(成都二环路北一段 111 号)

经 销
邮 编
发行科电话

新华书店
610031
028-87600564 87600533

网 址

<http://press.swjtu.edu.cn>

电子邮箱

cbsxx@swjtu.edu.cn

印 刷

四川煤田地质制图印刷厂

成品尺寸

205mm×285mm

印 张

18

字 数

628 千字

版 次

2006 年 7 月第 2 版

印 次

2006 年 7 月第 2 次印刷

书 号

ISBN 7-81104-373-4

定 价

28.00 元

版权所有, 盗版必究, 举报电话: (028) 87600562

出版说明

(第二版)

教材编写背景

教育部从战略的高度提出了国家精品课程建设的目标。在谈到什么是国家精品课程的问题上,教育部副部长吴启迪指出:国家精品课程就是具有一流教师队伍、一流教学内容、一流教学方法、一流教材、一流教学管理等特点的示范性课程。国家精品课程建设的六个方面之一,就是教材建设,即要建设或使用精品系列教材(包含各种媒体形式的立体化教材)。国家精品课程对教材的完整配套,如实验指导、习题等,都做了具体要求。

为配合各大专院校精品课程建设工作,并满足各大专院校计算机教学和等级考试的需要,本套教程的编委会一开始就把这套教程的编写出版工作定位在打造一流优秀精品教材的目标上。

本套教程自2004年第一版以来,与国内同类教材相比,站在了全国一流教材的前列。她以教材知识体系的科学性、结构安排的合理性、编写内容的完美性等多种优势和突出的特点,受到各大专院校的普遍肯定和赞扬,有效地帮助本套教材使用学校提高了等级考试过关率。编委会在保持第一版的优点和长处之外,又根据教学发展新需要,对该套教材的主讲教材作了全面的修订,并针对教学和等级考试的新特点,对实训教材作了重要调整,使最新推出的第二版主讲教材与实训教材更加同步,写作和出版质量更加提高和完善。同时,本套教程第二版以一种新颖独特的版面设计风格呈现在读者面前。

本套教程编委会由具有多年教学实践经验的各高校第一线老师们组成,并以具有丰富的教材编写经验,在全国计算机教材编写方面很有影响力的教授任主编。最值得称道的是,本套教程的编委们以严谨踏实的治学态度,结合教学实践,对本套教程第二版的修改提出了多方面的建设性改进意见。本套教程集编委会智慧的结晶,并在各大专院校广泛采用本套教程和反复修改的基础上,使本套教程在精品打造的千锤百炼中成熟起来,成为主编所编写过的多个版本中最成功的一个版本。

教材特色

1. 本教程博采国内类似教材之众长,力争在教材的系统性、科学性、创新性等方面走在全国同类教材的前面,把整个编写结构调整得非常到位,成为这类课程教学中的经典教材。本教程信息量大,实例丰富,结合教学实践的需要,使教材具有一定的宽容度和可选择性,以满足大专院校不同层次的教学需要。一方面照顾大专院校教学的普遍情况和学生基础,按计算机等级考试(二级)大纲覆盖教学内容,把问题讲透;另一方面,介绍有一定深度的教

学选讲内容,给教学提供进一步的扩展空间,使教师能根据教学计划和本校学生的实际情况灵活掌握教学。

2. 教材完整配套,为课程的教学和实验搭建了一个很好的多功能平台,强调对学生动手能力的训练,并为学生的自主学习和研究性学习提供了充足的资料。教师可以在教学中借助这个平台,进一步创建各种媒体形式的立体教学效果。为培养学生的实际动手能力,避免老师“空对空”的教学,减轻老师教学负担,整体提高学生 Visual FoxPro 二级考试通过率,本教程同时配套出版实验指导和附有参考答案及重点解析的习题题库的《Visual FoxPro 面向对象程序设计上机和级考实训教程(第二版)》(匡松 何振林主编,西南交通大学出版社),以满足不同学校,不同教学情况,不同学生的不同要求。本套教材既适用于省级计算机等级考试,也适用于全国计算机等级考试。这套完善的配套教材一定能使教师和学生受益匪浅。

3. 为满足各大专院校双语教学的需要,每章都提供相应的双语教学参考英语词汇。对一些计算机专业术语列出以基本含义为词根的英语名词、动词、形容词,以及一些同义词、近义词和词组等,谨供双语教学灵活掌握使用(鉴谅未注明英语词类、词性和音标)。

鸣谢

本套教程编委会秘书处以科学的系统工程概念进行整体组织出版策划,广泛听取各高校第一线老师的建议,致力于把这套教材打造成受广大师生喜爱的一流精品教材。在对本套教程稿件的组织、编审、设计和印制各环节中,编委会肩负着对教师和学生的高度责任感,付出了辛勤的劳动。编委会向参加编委会各项工作的各高校的院长、主任和教学第一线的老师们致以最诚挚的谢意!

21 世纪高校计算机教材出版策划编委会

E-mail: skj5555@126.com

skj5555@tom.com

starnet5555@tom.com

目 录

第 1 章 数据库基本理论	1	2.4.1 向 导	23
1.1 数据、信息和数据处理	2	2.4.2 设计器	23
1.1.1 数据与信息	2	2.4.3 生成器	24
1.1.2 数据处理	2	2.5 项目管理器	25
1.2 数据管理技术的发展	2	2.5.1 创建项目	25
1.2.1 人工管理阶段	3	2.5.2 项目管理器的使用	26
1.2.2 文件管理阶段	3	2.5.3 定制项目管理器	27
1.2.3 数据库系统阶段	3	第 3 章 Visual FoxPro 的数据基础 ...	29
1.3 数据库系统基本概念	4	3.1 数据类型	30
1.3.1 数据库	4	3.2 常 量	32
1.3.2 数据库管理系统	5	3.3 变 量	32
1.3.3 数据库系统	5	3.3.1 内存变量	33
1.3.4 数据库应用系统	5	3.3.2 数组变量	34
1.3.5 数据库系统的数据模式	6	3.3.3 字段变量	35
1.4 数据模型	6	3.3.4 系统变量	35
1.4.1 数据模型的三要素	6	3.4 表达式	35
1.4.2 实体的描述	7	3.5 常用函数	37
1.4.3 实体间的联系及联系的种类	7	3.5.1 数值运算函数	38
1.4.4 常用数据模型	7	3.5.2 字符处理函数	40
1.5 关系数据库基础知识	10	3.5.3 转换函数	43
1.5.1 关系术语	10	3.5.4 日期和时间函数	44
1.5.2 关系的规范化	10	3.5.5 测试函数	45
1.5.3 关系运算	12	3.6 Visual FoxPro 的文件类型	48
1.5.4 关系的完整性	13	3.7 Visual FoxPro 表的类型	49
第 2 章 Visual FoxPro 系统概述	15	第 4 章 表的建立和操作	51
2.1 Visual FoxPro 的特点及性能指标	16	4.1 表的建立	52
2.1.1 Visual FoxPro 的特点	16	4.1.1 表结构的分析和设计	52
2.1.2 Visual FoxPro 的性能指标	17	4.1.2 表结构的建立	54
2.2 Visual FoxPro 的启动与用户界面	17	4.1.3 表数据的输入	55
2.2.1 Visual FoxPro 的运行环境	17	4.2 数据表的打开与关闭	59
2.2.2 Visual FoxPro 的启动与退出	17	4.2.1 打开表	59
2.2.3 Visual FoxPro 的界面	18	4.2.2 关闭表	59
2.2.4 Visual FoxPro 的菜单系统	19	4.3 表结构的修改	60
2.2.5 工作区窗口和命令窗口	21	4.4 表的记录定位和显示	61
2.3 Visual FoxPro 的工作方式和命令结构	22	4.4.1 记录的定位	61
2.3.1 Visual FoxPro 的工作方式	22	4.4.2 数据的显示与修改	62
2.3.2 Visual FoxPro 的命令结构	22	4.5 表结构和数据的复制	66
2.4 Visual FoxPro 的辅助工具	23	4.5.1 复制任何类型的文件	66

4.5.2 复制表文件	66	6.3 视图	107
4.5.3 复制表的结构	67	6.3.1 本地视图的创建	107
4.6 记录的删除与恢复	67	6.3.2 远程视图的创建	111
4.6.1 逻辑删除表中的记录	68	第7章 结构化查询语言 SQL	115
4.6.2 恢复表中逻辑删除的记录	69	7.1 SQL 的数据定义功能	116
4.6.3 物理删除表中的记录	71	7.1.1 创建数据表	116
4.7 修改表中的数据	73	7.1.2 修改数据表	117
4.7.1 用 EDIT/CHANGE、REPLACE 和 BROWSE 命令修改	73	7.1.3 删除数据表	118
4.7.2 数据表与数组之间的数据交换	75	7.2 SQL 的数据查询功能	118
4.8 数据表的过滤	76	7.2.1 SELECT 命令的格式	118
4.8.1 记录过滤	77	7.2.2 投影查询	119
4.8.2 字段过滤	77	7.2.3 条件查询	120
第5章 排序、查询和多表操作	79	7.2.4 统计查询	122
5.1 排 序	80	7.2.5 分组查询	123
5.2 索 引	81	7.2.6 查询的排序	124
5.2.1 索引的概念及类型	81	7.2.7 连接查询	124
5.2.2 索引的建立	82	7.2.8 嵌套查询	126
5.2.3 索引的使用、更新与删除	85	7.2.9 查询结果输出	129
5.3 查 询	87	7.3 SQL 的数据更新功能	129
5.3.1 顺序查询	87	7.3.1 插入数据记录	130
5.3.2 索引查询	88	7.3.2 修改数据记录	130
5.4 统计与汇总	89	7.3.3 删除数据记录	131
5.4.1 计数命令 COUNT	89	7.4 用查询设计器建立查询	131
5.4.2 求和命令 SUM	90	7.4.1 进入查询设计器	132
5.4.3 求平均命令 AVERAGE	90	7.4.2 创建查询	134
5.4.4 计算命令 CALCULATE	90	7.4.3 指定查询去向	134
5.4.5 汇总命令 TOTAL	91	7.4.4 查看 SQL	135
5.5 多工作区的操作	91	7.4.5 运行、保存和修改查询	135
5.5.1 工作区的选择与数据工作期	91	7.4.6 查询菜单	135
5.5.2 表之间的关系	93	第8章 Visual FoxPro 程序设计基础	137
第6章 数据库操作与视图的使用	97	8.1 面向过程程序设计	138
6.1 数据库的创建与管理	98	8.1.1 程序文件的建立和执行	138
6.1.1 数据库的创建	98	8.1.2 程序中常用的命令	139
6.1.2 在项目中添加数据库	99	8.1.3 程序的基本结构	143
6.1.3 数据库的打开、修改和关闭	99	8.1.4 过程及过程调用	148
6.1.4 数据库对表的管理	101	8.1.5 变量的作用域和参数调用	149
6.1.5 数据库文件察看	102	8.1.6 数组应用	151
6.2 数据字典	102	8.2 面向对象程序设计	152
6.2.1 字段属性	103	8.2.1 基本概念	152
6.2.2 记录规则	104	8.2.2 面向对象程序设计基本方法	153
6.2.3 永久关系	105	8.2.3 Visual FoxPro 中的类	154
6.2.4 设置参照完整性	106	8.2.4 属 性	154
		8.2.5 事 件	155

8.2.6	方法程序	156
8.2.7	数据环境	156
8.2.8	对象的操作	156
8.2.9	设计类与对象	157
8.3	调试程序	160
8.3.1	调试器窗口	160
8.3.2	调试器应用	161
第9章	表单设计	163
9.1	表单设计基础	164
9.1.1	表单简介	164
9.1.2	表单向导	167
9.1.3	表单设计器	169
9.2	表单常用控件	172
9.2.1	控件的基本操作	172
9.2.2	标签控件	173
9.2.3	文本框控件	174
9.2.4	命令按钮控件	176
9.2.5	常用表单控件应用	177
9.3	表单其他控件	178
9.3.1	选项按钮组	178
9.3.2	复选框	179
9.3.3	编辑框	180
9.3.4	列表框	181
9.3.5	组合框	182
9.3.6	微调按钮	184
9.3.7	计时器	185
9.3.8	图像	185
9.3.9	表格	186
9.3.10	页框	187
9.3.11	命令按钮组	187
9.3.12	ActiveX 控件和 ActiveX 绑定控件	189
9.3.13	表单集	190
9.3.14	表单控件的综合应用	191
第10章	Visual FoxPro 程序设计	
	综合应用	195
10.1	数据库操作实例	196
10.1.1	浏览数据实例	196
10.1.2	维护数据实例	197
10.1.3	学生选课实例	201
10.1.4	成绩查询实例	204
10.1.5	教师课程查询实例	206
10.1.6	计算平均年龄实例	207
10.2	表单控件应用实例	208

10.2.1	求素数实例	208
10.2.2	打字测试实例	210
10.2.3	算术计算实例	211
10.2.4	数值排序实例	213
第11章	报表设计	215
11.1	报表设计基础	216
11.1.1	报表的常规布局	216
11.1.2	报表设计的步骤	216
11.1.3	创建报表布局文件	217
11.2	创建简单报表	217
11.2.1	创建空白报表	217
11.2.2	创建快速报表	217
11.2.3	用报表向导创建报表	218
11.3	报表设计器	220
11.3.1	报表设计器的基本环境	220
11.3.2	设置报表的数据环境	221
11.3.3	报表控件	221
11.3.4	报表控件的调整与控制	224
11.4	报表打印输出	224
11.4.1	页面设置	224
11.4.2	预览结果	225
11.4.3	打印报表	225
11.5	报表设计示例	226
第12章	应用系统的集成	229
12.1	菜单系统	230
12.1.1	菜单系统的基本结构	230
12.1.2	菜单系统的设计步骤	231
12.2	设计菜单	232
12.2.1	菜单设计器的使用	232
12.2.2	主菜单中的“显示”和“菜单”下拉 菜单中有关选项	233
12.2.3	下拉式菜单设计示例	233
12.2.4	快捷菜单	235
12.3	应用系统的集成	236
12.3.1	应用系统启动主程序的设计	236
12.3.2	项目集成	237
12.3.3	应用程序的发布	238
第13章	软件技术基础	241
13.1	数据结构	242
13.1.1	数据结构的基本概念	242
13.1.2	数据结构的意义	243
13.1.3	算法分析	244
13.1.4	线性表	245

13.1.5	树	248
13.1.6	图结构	250
13.1.7	线性表的查找	252
13.1.8	内排序	253
13.2	软件工程	256
13.2.1	概 述	256
13.2.2	软件生存周期	257
13.2.3	软件定义	258
13.2.4	软件开发	259
13.2.5	软件测试	260
13.2.6	软件维护	260

13.2.7	计算机辅助软件工程	261
13.3	操作系统	261
13.3.1	操作系统的概念和类型	261
13.3.2	处理机管理	263
13.3.3	存储器管理	266
13.3.4	设备管理	266
13.3.5	文件管理	267
附 录		269
附录一	Visual FoxPro 6.0 常用函数表	269
附录二	Visual FoxPro 6.0 命令概要	273
附录三	研究性学习参考书目	281

第1章

数据库基本理论

本章双语教学参考英语词汇

参照完整性 consult integrity

层次模型 hierarchical model

发展 development

分布式数据库 distributed database

概述 briefly, in broad outline

关键字 key word

关系 relation, relationship

关系模式 relation schema/scheme

关系模型 relation model, relational model

关系数据库 relational database

关系运算 relation(al) operation

基本知识 basic knowledge

集中式数据库 centralized database

技术 technology

阶段 stage

控制 control

连接 connection, link, join, bind, connect, conjunct

人工管理 man-made management

实体完整性 entity integrity,
entity-integrity property

术语 terms, terminology

数据 data

数据安全 data security

数据采集 data acquisition

数据操纵语言 data manipulation language

数据处理 data processing, data handing

数据处理系统 data processing system

数据处理系统 data processing system

数据处理中心 data processing center

数据处理周期 data processing cycle

数据定义语言 data definition language

数据丢失 data loss

数据格式 data format

数据管理 data management/administration

数据检索 data retrieval

数据控制 data control

数据库 database

数据库管理软件 data base management software

数据库管理系统 database management system

数据库管理员 database manager

数据库结构 database structure

数据库设计 database design

数据库引擎 database engine

数据库应用系统 data base application systems

数据模型 data model

数据冗余 data redundancy

数据视图 data view

数据完整性 data integrity

数据文件 data file

数据无关性 data independence

数据项 data item

数据银行 data bank

数据属性 data attribute

数据字典 data dictionary

投影 project, projection

完整性 integrity

网状模型 network model

文件管理 document management

系统 system

信息 information, message

选择 select, choose

应用系统 applied system

域 territory, region, area, field, domain

域完整性 territory integrity

元组 tuple

运算 operate, operation

属性 attribute, quality

本章导读

本章主要介绍信息、数据、数据处理、数据模型的基本概念；重点介绍数据库、数据库管理系统、数据库系统的功能、组成和相互关系；还介绍了关系数据库的三种关系运算和关系表中的记录、字段、关键字段、关系模型等概念。

1.1 数据、信息和数据处理

1.1.1 数据与信息

1. 数据

数据是反映客观事物属性的记录，是描述或表达信息的具体表现形式，是信息的载体。在计算机领域，凡能为计算机所接受和处理的物理形式，例如字符、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（如产量、价格、成绩等）和非数值型数据（如人名、日期、文章、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

信息是经过加工处理并对人类客观行为产生影响的数据表现形式。信息是客观事物属性的反映。信息是有用的数据，是通过数据符号来传播的。信息无时不有，无处不在，它客观存在于人类社会的各个领域，而且不断变化。人们需要不断地获取信息、加工信息和运用信息为社会的各个领域服务。从计算机应用的角度来说，人们通常将信息看成是进行各种活动所需要获取的知识。

信息与数据既有联系又有区别。数据反映了信息的内容，而信息又依靠数据来表达；用不同的数据形式可以表示同样的信息，但信息并不随它的数据形式的不同而改变。例如，某个部门召开会议这个事件形成了“开会”这样一个信息。把这个信息通知有关单位时，可以使用广播，即通过声音这种数据形式向有关单位传递；也可以通过文件，以文字数据形式向有关单位传递。有关单位就从声音和文字这两种不同的数据形式中得到“开会”这一信息。尽管声音和文字这两种数据形式不同，但“开会”这个信息的内容没变，因此可以说信息是数据的内涵，而数据是信息的具体表现形式。在许多地方，信息和数据并不是截然分开的，因为有些信息本身就是数据化的，数据本身又是一种信息。因此在多数情况下不对它们进行区分，计算机进行数据交换也可以说是信息交换，进行数据处理也指信息处理。总之，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。信息这种被加工为特定形式的数据对于使用者来说是有意义的，而且对当前和将来的决策具有明显的实际价值。

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库基础和决策依据。

1.2 数据管理技术的发展

数据管理是指对数据进行组织、存储、分类、检索和维护等数据处理的技术，是数据处理的核心。随着计算机硬件技术和软件技术的发展，计算机数据管理的水平不断提高，管理方式也发生了很大的变化。数据管理技术的发展主要经历了人工管理、文件管理和数据库系统管理三个阶段。

1.2.1 人工管理阶段

人工管理阶段始于 20 世纪 50 年代, 出现在计算机应用于数据管理的初期。这个时期的计算机主要用于科学计算。从硬件看, 由于当时没有磁盘作为计算机的存储设备, 数据只能存放于卡片、纸带、磁带上。在软件方面, 既无操作系统, 也无专门管理数据的软件, 数据由计算或处理它的程序自行携带。

在人工管理阶段数据管理存在的主要问题是:

① 数据不能独立, 编写的程序要针对程序中的数据。当数据修改时, 程序也得修改, 而程序修改后, 数据的格式、类型也得变化以适应处理它的程序。

② 数据不能长期保存。数据被包含在程序中, 程序运行结束后, 数据和程序一起从内存中释放。

③ 没有专门进行数据管理的软件。人工管理阶段不仅要设计数据的处理方法, 而且还要说明数据在存储器中的存储地址。应用程序和数据是相互结合且不可分割的, 各程序之间的数据不能相互传递, 数据不能被重复使用。因而这种管理方式既不灵活, 也不安全, 编程效率低。

④ 一组数据对应于一个程序, 一个程序中的数据不能被其他程序利用, 数据无法共享, 从而导致程序和程序之间有大量重复的数据存在。

人工管理阶段程序与数据之间的关系如图 1.1 所示。

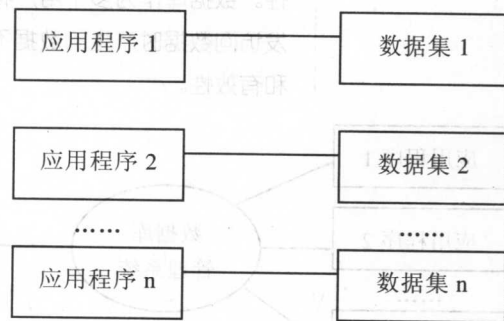


图 1.1 人工管理阶段程序与数据之间的关系

1.2.2 文件管理阶段

在 20 世纪 60 年代, 计算机软、硬件技术得到快速发展, 硬件方面有了磁盘、磁鼓等大容量且能长期保存数据的存储设备, 软件方面有了操作系统。操作系统中有专门的文件系统用于管理外部存储器上的数据文件, 数据与程序分开, 数据能长期保存。

在文件管理阶段, 把有关的数据组织成一个文件, 这种数据文件能够脱离程序而独立存储在外存储器上, 由一个专门的文件管理系统对其进行管理。在这种管理方式下, 应用程序通过文件管理系统对数据文件中的数据进行加工处理。应用程序与数据文件之间具有一定的独立性。与早期人工管理阶段相比, 使用文件系统管理数据的效率和数量都有很大提高, 但仍存在以下问题:

① 数据没有完全独立。虽然数据和程序被分开, 但所设计的数据依然是针对某一特定的程序, 所以无论是修改数据文件还是程序文件二者都要相互影响。也就是说, 数据文件仍然高度依赖于其对应的程序, 不能被多个程序所共享。

② 存在数据冗余。文件系统中的数据没有合理和规范的结构, 使得数据的共享性极差, 哪怕是不同程序使用部分相同数据, 数据结构也完全不同, 也要创建各自的数据文件。这便造成了数据的重复存储, 即数据的冗余。

③ 数据不能被集中管理。文件系统中的数据文件没有集中的管理机制, 数据的安全性和完整性都不能得到保障。各数据之间、数据文件之间缺乏联系, 给数据处理造成不便。

文件系统阶段程序与数据之间的关系如图 1.2 所示。

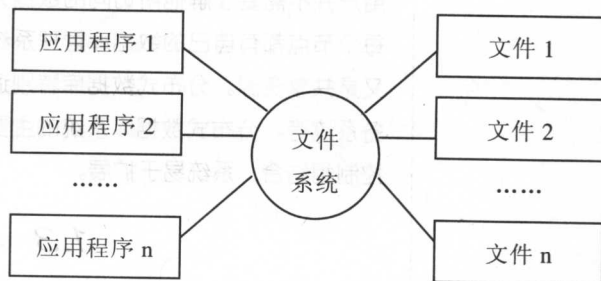


图 1.2 文件系统阶段程序与数据之间的关系

1.2.3 数据库系统阶段

由于文件系统管理数据存在缺陷, 迫切需要一种新的数据管理方式, 把数据组成合理结构, 进行集中、统一管理。数据库技术始于 20 世纪 60 年代末, 到了 20 世纪 80 年代, 随着计算机的普遍应用和数据库系统的不断完善, 数据库系统在全世界范围内得到广泛的应用。

在数据库系统管理阶段, 是将所有的数据集中到一个数据库中, 形成一个数据中心, 实行统一规划,

集中管理, 用户通过数据库管理系统 (Data Base Management System, DBMS) 来使用数据库中的数据。

1. 数据库系统的主要特点

① 实现了数据的结构化。数据库采用了特定的数据模型组织数据。数据库系统把数据存储于有一定结构的数据库文件中, 实现了数据的独立和集中管理, 克服了人工管理和文件管理的缺陷, 大大方便了用户的使用和提高了数据管理的效率。

② 实现了数据共享。数据库中的数据能为多个用户服务。

③ 实现了数据独立。用户的应用程序与数据的逻辑结构及数据的物理存储方式无关。

④ 实现了数据统一控制。数据库系统提供了各种控制功能, 保证了数据的并发控制、安全性和完整性。数据库作为多个用户和应用程序的共享资源, 允许多个用户同时访问。并发控制可以防止多用户并发访问数据时产生的数据不一致性; 安全性可以防止非法用户存取数据; 完整性可以保证数据的正确性和有效性。

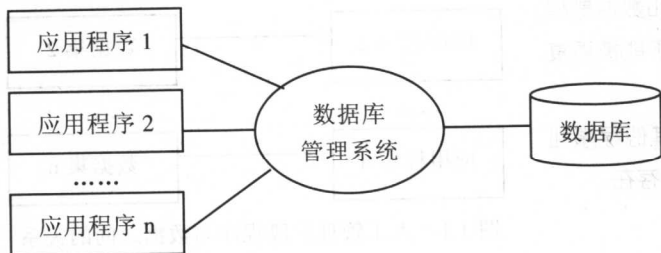


图 1.3 数据库系统阶段程序与数据之间的关系

在数据库系统阶段, 应用程序和数据完全独立, 应用程序对数据管理和访问更加灵活。一个数据库可以为多个应用程序共享, 使得程序的编制和效率大大提高, 减少了数据冗余, 实现了数据资源共享, 提高了数据的完整性、一致性以及数据的管理效率。

数据库系统阶段程序与数据之间的关系如图 1.3 所示。

2. 数据库系统的分类

数据库系统的分类有多种方式, 按照数据的存放地点的不

同, 数据库系统可分为集中式数据库系统和分布式数据库系统。

① 集中式数据库系统是将数据集中在一个数据库中。数据在逻辑上和物理上都是集中存放的。所有的用户在存取和访问数据时, 都要访问这个数据库。例如一个银行储蓄系统, 如果系统的数据存放在一个集中式数据库中, 那么所有储户在存款和取款时都要访问这个数据库。这种方式访问方便, 但通信量大, 速度慢。

② 分布式数据库系统是将多个集中式的数据库通过网络连接起来, 使各个节点的计算机可以利用网络通信功能访问其他节点上的数据库资源, 使各个数据库系统的数据实现高度共享。分布式数据库系统是在 20 世纪 70 年代后期开始使用的, 由于网络技术的发展为数据库提供了良好的运行环境, 使数据库系统从集中式发展到分布式, 从主机/终端系统发展到客户机/服务器系统。在网络环境中, 分布式数据库在逻辑上是一个集中式数据库系统, 而在物理上, 数据是存储在计算机网络的各个节点上。每个节点的用户并不需要了解他所访问的数据究竟在什么地方, 就如同在使用集中式数据库一样, 因为在网络上的每个节点都有自己的数据库管理系统, 都具有独立处理本地事务的能力, 而且这些物理上分布的数据库又是共享资源。分布式数据库特别适合地理位置分散的部门和组织机构, 如铁路民航订票系统、银行业务系统等。分布式数据库系统的主要特点是: 系统具有更高的透明度, 可靠性与效率更高, 局部与集中控制相结合, 系统易于扩展。

1.3 数据库系统基本概念

1.3.1 数据库

数据库 (Data Base, 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彼此相关, 而且数据记录之间在结构上也是有机地联系在一起。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

数据库管理系统是负责数据库的定义、建立、操纵、管理和维护的一种计算机软件，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进行工作的，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 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。

数据库管理系统通常由以下几个部分组成：

- ① 数据定义语言 DDL (Data Definition Language) 及其编译和解释程序 —— 主要用于定义数据库的结构。
- ② 数据操纵语言 DML (Data Manipulation Language) 或查询语言及其编译或解释程序 —— 提供了对数据库中的数据存取、检索、统计、修改、删除、输入、输出等基本操作。
- ③ 数据库运行管理和控制例行程序 —— 是数据库管理系统的核心部分，用于数据的安全性控制、完整性控制、并发控制、通信控制、数据存取、数据库转储、数据库初始装入、数据库恢复和数据的内部维护等，这些操作都是在该控制程序的统一管理下进行的。
- ④ 数据字典 DD (Data Dictionary) —— 提供了对数据库数据描述的集中管理规则，对数据库的使用和操作可以通过查阅数据字典来进行。

1.3.3 数据库系统

数据库系统 (Data Base System, DBS) 是指计算机系统引入数据库后的系统构成，是一个具有管理数据库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、数据库管理系统和建立在该数据库之上的相关软件、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。

- ① 硬件系统 —— 数据库系统的物理支持，包括主机、显示器、外存储器、输入/输出设备等。
- ② 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。
- ③ 数据库 —— 数据库系统的管理对象，为用户提供数据的信息源。
- ④ 数据库管理员 —— 负责管理和控制数据库系统的主要维护管理人员。
- ⑤ 用户 —— 数据库的使用者，可以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，是负责开发应用系统程序的设计人员；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。

1.3.4 数据库应用系统

数据库应用系统 (Data Base 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

1. 数据库系统的三级模式

从数据库管理系统的角度看,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 **模式** —— 也称逻辑模式或概念模式,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,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 提供模式定义语言 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;数据项的名字、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。

② **内模式** —— 又称为存储模式,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。

③ **外模式** —— 又称子模式或用户模式,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所使用,但一个应用程序只有一个外模式。

2. 数据库系统的两层映像

为了实现三个抽象级别的联系和转换,数据库管理系统在三级结构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,指出映像双方如何进行转换。数据库的三级结构靠映像联结。

① **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,相应的应用程序也不必修改,可保持外模式不变,实现数据和程序的逻辑独立性。

② **模式/内模式映像** —— 定义数据全局逻辑结构与存储结构间的对应关系。当数据库的物理存储结构改变时,只需修改模式与内模式之间的对应关系,即可保持模式不变,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,主要用于数据库设计。而逻辑数据模型常称为**数据模型**,是一种面向数据库系统的模型,主要用于数据库管理系统的实现。

1.4.1 数据模型的三要素

数据模型通常由数据结构、数据操作和完整性约束三个要素组成。

① **数据结构** —— 所研究的对象类型的集合,它用来描述系统的集合结构,规定了数据模型的静态特性。数据结构规定了如何把基本的数据项组织成较大的数据单位,以描述数据的类型、内容、性质和数据之间的相互关系。在数据库系统中,通常按照数据结构的类型来命名数据模型。

② **数据操作** —— 用于描述数据模型的动态特性。数据操作是指对数据库中各种对象(型)的实例(值)允许执行的操作的集合,包括操作及有关的操作规则。数据库中主要的操作有查询和更新(包括插入、删除和修改)两大类。数据模型要定义这些操作的确切含义、操作规则以及实现操作的语言。

③ **数据的约束条件** —— 一组完整性规则的集合。完整性约束的定义对数据模型的动态特性作了进一步的描述和限定。完整性规则定义了给定数据模型中数据及其联系所具有的制约和依存规则,用以限

定符合数据模型的数据库状态以及状态的变化,以保证数据库中数据的正确性、有效性和相容性。

1.4.2 实体的描述

信息世界中的基本概念包括实体、属性、码、域、实体型及实体集。

① **实体** —— 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。

② **属性** —— 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。例如铅笔有商标、软硬度、颜色、价格、生产厂家等属性;学生有学号、姓名、性别、出生日期、籍贯、专业、是否团员等属性。

③ **码(关键字)** —— 唯一标识实体的一个属性或属性集称为码。例如学号是学生实体的码。

④ **域** —— 属性的取值范围。例如:学生性别的域为(男,女)。

⑤ **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:学生以及学生的属性名集合构成学生实体型,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);铅笔(商标,软硬度,颜色,价格,生产厂家)表示铅笔实体型。

⑥ **实体集** —— 同类型的实体的集合称为实体集。例如:全体学生就是一个实体集。

1.4.3 实体间的联系及联系的种类

两个实体间的联系可以分为三类:

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:在学校里面,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对的,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 ($n \geq 0$) 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 与实体集 B 有一对多联系。

考查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(n:m)** —— 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 ($n \geq 0$) 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 ($m \geq 0$) 与之联系,则称实体集 A 与实体集 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,一个项目可有多个职工参加,则职工与项目之间具有多对多联系。

实体型之间的一对一、一对多、多对多联系不仅存在于两个实体型之间,也存在于两个以上的实体型之间。同一个实体集内的各实体之间也可以存在一对一、一对多、多对多的联系,称为**自联系**。

1.4.4 常用数据模型

客观事物的普遍联系性,决定了作为事物属性记录符号的数据与数据之间也存在着一定的联系。具有联系性的相关数据总是按照一定的组织关系排列,从而构成一定的结构,对这种结构的描述就是数据模型。数据模型是数据库系统中用于提供信息表示和操作手段的结构形式。简单地说,数据模型是指数

数据库的组织形式，它决定了数据库中数据之间联系的方式。在数据库系统设计时，数据库的性质是由系统支持的数据模型来决定的。不同的数据模型以不同的方式把数据组织到数据库中。

常见的数据模型有 3 种，即：层次模型、网状模型和关系模型。如果数据库中的数据是依照层次模型存储的数据，该数据库称为**层次数据库**；如果是依照网状模型进行存储，该数据库称为**网状数据库**；如果是依照关系模型进行存储，该数据库称为**关系数据库**。

1. 层次模型

层次模型是数据库系统最早使用的一种模型。层次模型表示数据间的从属关系结构，它是以树型结构表示实体（记录）与实体之间联系的模型。层次模型的主要特征是：

- ① 层次模型像一棵倒立的树，仅有一个无双亲的根结点。
- ② 根结点以外的子结点，向上仅有一个父结点，向下有若干子结点。

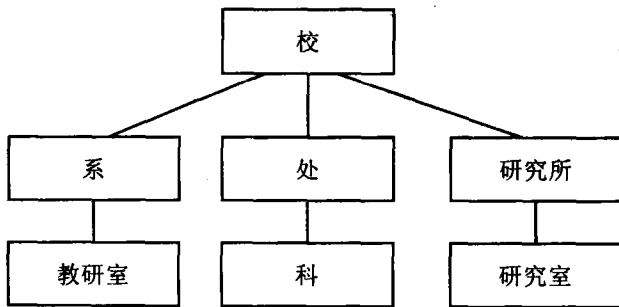


图 1.4 学校行政机构的层次模型

层次数据模型只能直接表示一对多（包括一对一）的联系，但不能表示多对多联系。例如：学校的行政机构（如图 1.4 所示）、企业中的部门编制等都是层次模型。支持层次模型的数据库管理系统称为**层次数据库管理系统**。

2. 网状模型

网状模型是一种比较复杂的数据模型，它是以网状结构表示实体与实体之间联系的模型。网状模型可以表示多个从属关系的层次结构，也可以表示数据间的交叉关系，是层次模型的扩展。网状模型的主要特征是：

- ① 有一个以上的结点无双亲。
- ② 至少有一个结点有多个双亲。

网状数据模型的结构比层次模型更具普遍性，它突破了层次模型的两个限制，允许多个结点没有双亲结点，允许结点有多个双亲结点。此外，它还允许两个结点之间有多种联系。因此网状数据模型可以更直接地描述现实世界。图 1.5 给出了一个简单的网状模型。

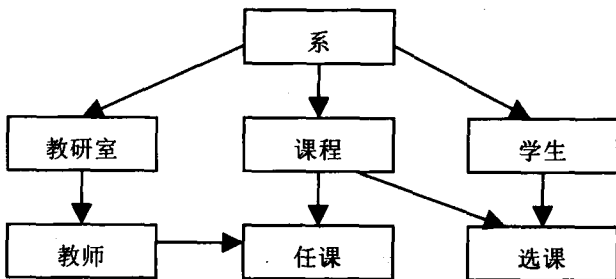


图 1.5 网状模型示例

网状模型是以记录为结点的网络结构。支持网状数据模型的数据库管理系统称为**网状数据库管理系统**。

3. 关系模型

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有：

- ① 关系中的每一分量不可再分，是最基本的数据单位。
- ② 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。
- ③ 关系中每一行由一个个体事物的诸多属性构成，且各行的顺序可以是任意的。
- ④ 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 所示的是一张学生情况表。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。

关系模型对数据库的理论和实践产生了极大的影响，它与层次模型和网状模型相比有明显的优势，是目前最流行的数据库模型。支持关系模型的数据库管理系统称为**关系数据库管理系统**。Visual FoxPro 采用的数据模型是关系模型，因此它是一个关系数据库管理系统。