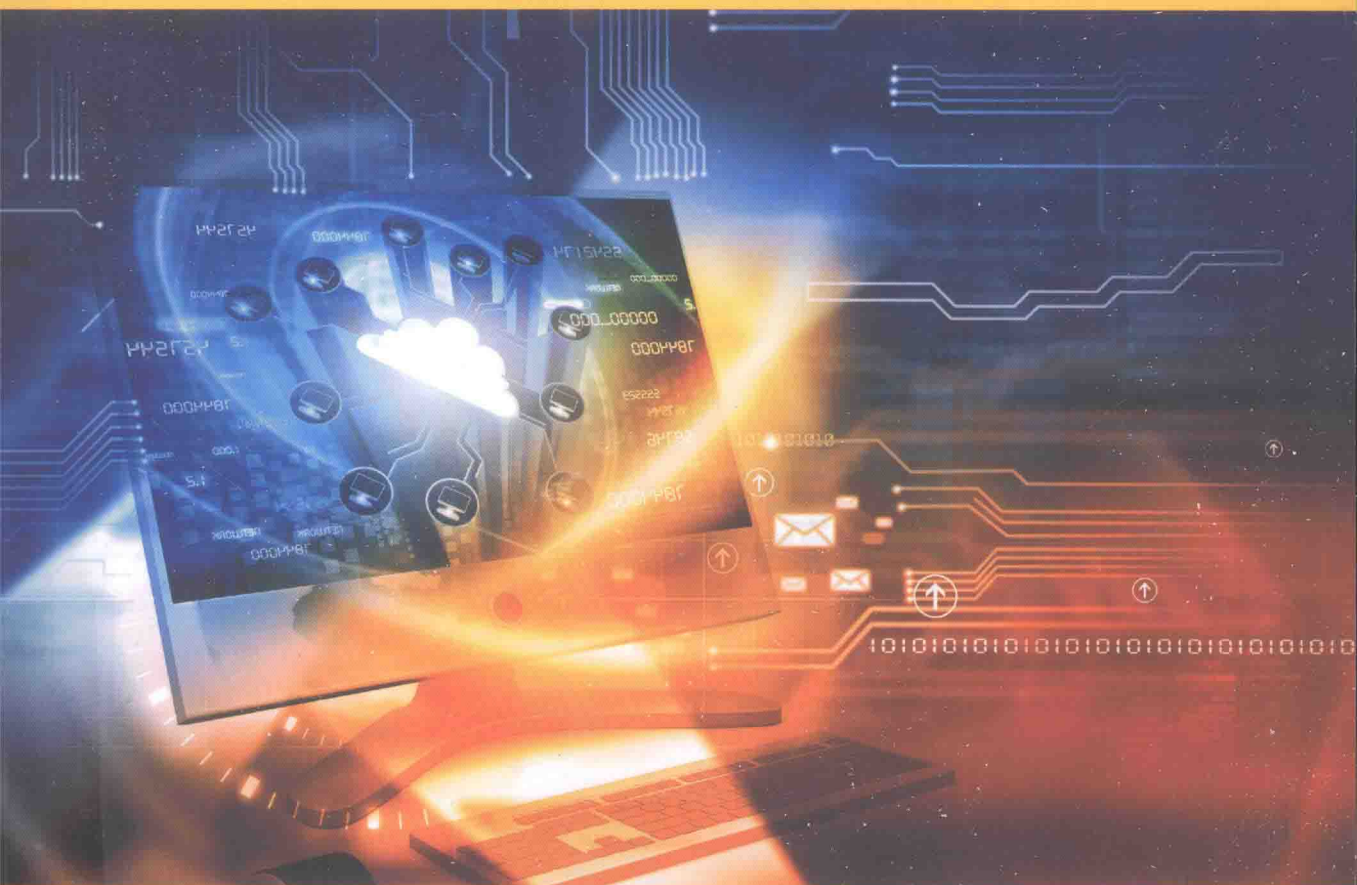


主编 贡黎明 周 蕾 施梅芳

# Visual FoxPro

## 程序设计基础教程



苏州大学出版社  
Soochow University Press

# Visual FoxPro 程序设计基础教程

主编 贲黎明 周 蕾 施梅芳

苏州大学出版社

图书在版编目(CIP)数据

Visual FoxPro 程序设计基础教程 / 贲黎明, 周蕾,  
施梅芳主编. —苏州: 苏州大学出版社, 2015. 12  
ISBN 978-7-5672-1586-3

I. ①V… II. ①贲… ②周… ③施… III. ①关系数  
据库系统—程序设计—教材 IV. ①TP311.138

中国版本图书馆 CIP 数据核字(2015)第 283251 号

**Visual FoxPro 程序设计基础教程**

贲黎明 周蕾 施梅芳 主编

责任编辑 周建兰

---

苏州大学出版社出版发行

(地址: 苏州市十梓街 1 号 邮编: 215006)

苏州恒久印务有限公司印装

(地址: 苏州市友新路 28 号东侧 邮编: 215128)

---

开本 787 mm×1 092 mm 1/16 印张 17 字数 422 千

2015 年 12 月第 1 版 2015 年 12 月第 1 次印刷

ISBN 978-7-5672-1586-3 定价: 37.00 元

---

苏州大学版图书若有印装错误, 本社负责调换

苏州大学出版社营销部 电话: 0512-65225020

苏州大学出版社网址 <http://www.sudapress.com>

# 前 言

数据库技术已成为当今信息社会的基础技术,是管理类专业人员必须掌握的基础知识。Visual FoxPro 是微软公司推出的数据库管理系统,它采用面向对象的程序设计思想以及可视化的操作方法,易学易用。

本书根据高校计算机公共基础教学的需要,并参照全国计算机等级考试大纲的要求编写,不仅可作为高等院校非计算机专业学生学习计算机课程的教材,也可作为计算机等级考试和计算机爱好者的自学教材。

本书共分 10 章,第 1 章介绍了数据库概论,第 2 章介绍了 Visual FoxPro 基础,第 3 章介绍了表的创建与使用,第 4 章介绍了数据库的创建与使用,第 5 章介绍了查询与视图,第 6 章介绍了程序设计基础,第 7 章介绍了表单的创建与使用,第 8 章介绍了控件,第 9 章介绍了菜单设计,第 10 章介绍了报表的创建与使用。为了便于教学,本书还有与之配套的实验教材《Visual FoxPro 程序设计基础实验教程》,供教师和学生实验环节使用。

本书由贲黎明、周蕾、施梅芳主编,朱苗苗、肖乐、宗德才副主编,全书由贲黎明统稿。

本书在编写过程中,由于时间仓促,加上水平有限,书中难免有错漏之处,恳请广大读者批评指正。

编 者

2015 年 12 月

# 目 录

## Contents .....

### 第 1 章 数据库概论

|                          |    |
|--------------------------|----|
| 1.1 数据管理技术 .....         | 1  |
| 1.1.1 数据与数据处理 .....      | 1  |
| 1.1.2 数据管理技术的发展 .....    | 2  |
| 1.2 数据库系统的组成 .....       | 5  |
| 1.3 数据模型 .....           | 7  |
| 1.3.1 数据抽象的过程 .....      | 7  |
| 1.3.2 概念模型 .....         | 7  |
| 1.3.3 逻辑模型 .....         | 9  |
| 1.4 关系数据库 .....          | 11 |
| 1.4.1 关系模型的基本概念 .....    | 11 |
| 1.4.2 关系数据库 .....        | 13 |
| 1.4.3 将概念模型转化为关系模型 ..... | 13 |
| 1.4.4 关系运算 .....         | 15 |
| 1.4.5 关系模型的完整性 .....     | 18 |
| 习题 1 .....               | 20 |

### 第 2 章 Visual FoxPro 基础

|                                   |    |
|-----------------------------------|----|
| 2.1 Visual FoxPro 操作环境 .....      | 23 |
| 2.1.1 Visual FoxPro 操作界面 .....    | 24 |
| 2.1.2 命令说明 .....                  | 24 |
| 2.1.3 配置 Visual FoxPro 操作环境 ..... | 25 |
| 2.2 Visual FoxPro 的项目管理器 .....    | 26 |
| 2.2.1 项目及有关概念 .....               | 26 |
| 2.2.2 项目文件的创建与打开 .....            | 27 |
| 2.2.3 “项目管理器”窗口组成及设置 .....        | 27 |
| 2.2.4 项目管理器的使用 .....              | 29 |
| 2.3 Visual FoxPro 的数据类型 .....     | 30 |
| 2.4 Visual FoxPro 的常量与变量 .....    | 32 |
| 2.4.1 名称的命名规则 .....               | 32 |
| 2.4.2 常量 .....                    | 32 |
| 2.4.3 变量 .....                    | 33 |



|                               |    |
|-------------------------------|----|
| 2.5 Visual FoxPro 的数组 .....   | 36 |
| 2.5.1 数组的声明 .....             | 36 |
| 2.5.2 数组元素的赋值 .....           | 37 |
| 2.6 Visual FoxPro 的函数 .....   | 38 |
| 2.6.1 数值函数 .....              | 38 |
| 2.6.2 字符函数 .....              | 40 |
| 2.6.3 日期与时间函数 .....           | 42 |
| 2.6.4 数值类型转换函数 .....          | 43 |
| 2.6.5 其他常用函数 .....            | 45 |
| 2.7 Visual FoxPro 的表达式 .....  | 50 |
| 2.7.1 数值运算符与表达式 .....         | 50 |
| 2.7.2 字符运算符与表达式 .....         | 51 |
| 2.7.3 日期运算符与表达式 .....         | 51 |
| 2.7.4 关系运算符与表达式 .....         | 51 |
| 2.7.5 逻辑运算符与表达式 .....         | 52 |
| 2.7.6 名称表达式 .....             | 53 |
| 2.7.7 宏替换 .....               | 54 |
| 2.7.8 空值处理 .....              | 55 |
| 2.8 Visual FoxPro 的文件类型 ..... | 56 |
| 2.9 构造应用程序 .....              | 57 |
| 2.9.1 构造应用程序框架 .....          | 57 |
| 2.9.2 将文件添加到项目中 .....         | 60 |
| 2.9.3 连编应用程序 .....            | 62 |
| 习题2 .....                     | 63 |

### 第3章 表的创建与使用

|                     |    |
|---------------------|----|
| 3.1 表的创建 .....      | 66 |
| 3.1.1 表结构 .....     | 66 |
| 3.1.2 表结构的创建 .....  | 67 |
| 3.1.3 表结构的修改 .....  | 70 |
| 3.2 表的打开与记录维护 ..... | 71 |
| 3.2.1 表的打开与关闭 ..... | 71 |
| 3.2.2 表的浏览 .....    | 72 |
| 3.2.3 记录的输入 .....   | 75 |
| 3.2.4 记录的定位 .....   | 77 |
| 3.2.5 记录的修改 .....   | 79 |
| 3.2.6 记录的删除 .....   | 80 |
| 3.3 多表操作 .....      | 82 |
| 3.3.1 工作区 .....     | 82 |

|                          |    |
|--------------------------|----|
| 3.3.2 多表操作 .....         | 83 |
| 3.4 记录的索引 .....          | 84 |
| 3.4.1 索引概述 .....         | 84 |
| 3.4.2 索引文件的创建 .....      | 86 |
| 3.4.3 索引的修改 .....        | 87 |
| 3.4.4 索引的删除 .....        | 88 |
| 3.4.5 结构复合索引文件的使用 .....  | 88 |
| 3.4.6 索引定位 .....         | 89 |
| 3.5 记录的统计与表的复制 .....     | 90 |
| 3.5.1 记录的统计 .....        | 90 |
| 3.5.2 表的复制 .....         | 91 |
| 3.5.3 表与内存变量间的数据交换 ..... | 91 |
| 3.6 与表操作有关的常用函数 .....    | 92 |
| 习题3 .....                | 93 |

## 第4章 数据库的创建与使用

|                         |     |
|-------------------------|-----|
| 4.1 数据库的创建与操作 .....     | 96  |
| 4.1.1 数据库的创建 .....      | 96  |
| 4.1.2 数据库的打开 .....      | 97  |
| 4.1.3 当前数据库的设置 .....    | 97  |
| 4.1.4 数据库的关闭 .....      | 98  |
| 4.1.5 数据库的删除 .....      | 98  |
| 4.2 数据库表 .....          | 98  |
| 4.2.1 数据库表的创建 .....     | 98  |
| 4.2.2 数据库表的字段扩展属性 ..... | 99  |
| 4.2.3 数据库表的表属性 .....    | 101 |
| 4.2.4 数据库表的索引 .....     | 102 |
| 4.2.5 数据库表的添加及移去 .....  | 102 |
| 4.3 永久性关系及参照完整性 .....   | 104 |
| 4.3.1 永久性关系 .....       | 104 |
| 4.3.2 参照完整性 .....       | 105 |
| 4.4 与数据库操作有关的常用函数 ..... | 106 |
| 习题4 .....               | 108 |

## 第5章 查询与视图

|                         |     |
|-------------------------|-----|
| 5.1 查询设计 .....          | 111 |
| 5.1.1 查询的基本概念 .....     | 111 |
| 5.1.2 利用查询设计器创建查询 ..... | 111 |
| 5.1.3 使用查询向导创建查询 .....  | 123 |



|                                  |     |
|----------------------------------|-----|
| 5.2 视图设计 .....                   | 126 |
| 5.2.1 视图的基本概念 .....              | 126 |
| 5.2.2 使用视图设计器创建本地视图并更新源表数据 ..... | 126 |
| 5.2.3 创建参数化视图 .....              | 129 |
| 5.2.4 查询与视图的异同 .....             | 131 |
| 5.3 SELECT-SQL 命令 .....          | 131 |
| 5.3.1 SQL 语言概述 .....             | 131 |
| 5.3.2 SELECT-SQL 命令 .....        | 131 |
| 习题 5 .....                       | 137 |

## 第 6 章 程序设计基础

|                              |     |
|------------------------------|-----|
| 6.1 结构化程序设计概述 .....          | 142 |
| 6.1.1 结构化程序设计的概念 .....       | 142 |
| 6.1.2 程序文件的建立、修改与运行 .....    | 142 |
| 6.2 顺序结构 .....               | 145 |
| 6.3 分支结构 .....               | 145 |
| 6.3.1 IF 语句 .....            | 145 |
| 6.3.2 IIF() 语句 .....         | 147 |
| 6.3.3 DO CASE 语句 .....       | 148 |
| 6.4 循环结构 .....               | 149 |
| 6.4.1 DO WHILE 循环 .....      | 149 |
| 6.4.2 FOR 循环 .....           | 151 |
| 6.4.3 SCAN 循环 .....          | 154 |
| 6.4.4 FOR 循环与 DO 循环的比较 ..... | 155 |
| 6.4.5 循环的嵌套 .....            | 155 |
| 6.5 过程 .....                 | 157 |
| 6.5.1 过程 .....               | 157 |
| 6.5.2 自定义函数 .....            | 160 |
| 习题 6 .....                   | 162 |

## 第 7 章 表单的创建与使用

|                         |     |
|-------------------------|-----|
| 7.1 面向对象的程序设计基础 .....   | 169 |
| 7.1.1 对象、属性、事件与方法 ..... | 169 |
| 7.1.2 类 .....           | 173 |
| 7.2 表单的创建与运行 .....      | 175 |
| 7.2.1 利用表单向导创建表单 .....  | 175 |
| 7.2.2 利用表单设计器创建表单 ..... | 181 |
| 7.2.3 表单的运行 .....       | 186 |

|                        |     |
|------------------------|-----|
| 7.3 表单的使用 .....        | 187 |
| 7.3.1 表单的常用属性 .....    | 187 |
| 7.3.2 表单的常用事件与方法 ..... | 188 |
| 7.3.3 表单的数据环境 .....    | 189 |
| 习题7 .....              | 191 |

## 第8章 控 件

|                           |     |
|---------------------------|-----|
| 8.1 标签、文本框与编辑框 .....      | 194 |
| 8.1.1 标签 .....            | 194 |
| 8.1.2 文本框 .....           | 195 |
| 8.1.3 编辑框 .....           | 197 |
| 8.2 命令按钮与命令按钮组 .....      | 197 |
| 8.2.1 命令按钮 .....          | 197 |
| 8.2.2 命令按钮组 .....         | 198 |
| 8.3 列表框与组合框 .....         | 200 |
| 8.3.1 列表框 .....           | 200 |
| 8.3.2 组合框 .....           | 202 |
| 8.4 选项按钮组与复选框 .....       | 204 |
| 8.4.1 选项按钮组 .....         | 204 |
| 8.4.2 复选框 .....           | 205 |
| 8.5 表格 .....              | 206 |
| 8.6 微调框 .....             | 210 |
| 8.7 计时器 .....             | 212 |
| 8.8 线条与形状 .....           | 213 |
| 8.8.1 线条 .....            | 213 |
| 8.8.2 形状 .....            | 213 |
| 8.9 页框 .....              | 214 |
| 8.10 管理控件 .....           | 216 |
| 8.10.1 访问键管理 .....        | 216 |
| 8.10.2 工具提示信息管理 .....     | 216 |
| 8.10.3 控件 TAB 键次序管理 ..... | 216 |
| 8.10.4 OLE 控件管理 .....     | 217 |
| 习题8 .....                 | 218 |

## 第9章 菜单设计

|                       |     |
|-----------------------|-----|
| 9.1 菜单设计概述 .....      | 221 |
| 9.1.1 菜单系统的类型 .....   | 221 |
| 9.1.2 规划菜单系统 .....    | 222 |
| 9.1.3 设计菜单系统的步骤 ..... | 223 |



|                                |     |
|--------------------------------|-----|
| 9.1.4 Visual FoxPro 系统菜单 ..... | 223 |
| 9.2 下拉式菜单设计 .....              | 225 |
| 9.2.1 使用菜单设计器 .....            | 225 |
| 9.2.2 使用菜单设计器设计菜单 .....        | 230 |
| 9.2.3 为顶层表单添加下拉式菜单 .....       | 235 |
| 9.3 快捷菜单设计 .....               | 236 |
| 习题 9 .....                     | 237 |

## 第 10 章 报表的创建与使用

|                          |     |
|--------------------------|-----|
| 10.1 报表的类型 .....         | 240 |
| 10.2 报表的创建 .....         | 242 |
| 10.2.1 使用向导创建报表 .....    | 242 |
| 10.2.2 使用快速报表创建报表 .....  | 244 |
| 10.2.3 使用报表设计器创建报表 ..... | 245 |
| 10.3 数据分组报表 .....        | 253 |
| 10.3.1 单级数据分组 .....      | 254 |
| 10.3.2 多级数据分组 .....      | 255 |
| 10.4 报表的输出 .....         | 257 |
| 10.4.1 页面设置 .....        | 257 |
| 10.4.2 报表的预览与打印 .....    | 257 |
| 习题 10 .....              | 258 |

# 第1章

## 数据库概论

### 1.1 数据管理技术

#### 1.1.1 数据与数据处理

数据是描述事物的符号记录,是指用物理符号记录下来的可以鉴别的信息。物理符号可以是数字、文字、图形、图像、声音及其他特殊符号。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作是数据。

如表 1.1 所示,可以用学号、姓名、性别、身高、出生年份、籍贯、所在系来描述一个学生,表的首行称为“属性”,其他各行称为“记录”。表 1.1 学生表中包括 7 个属性(学号,姓名,性别,身高,出生年份,籍贯和所在系)。

表 1.1 中的每一行(除首行之外)用来描述一个具体的学生,表中的第一条记录(020311101,张飞,男,1.72,1992,江苏,计算机)描述的是张飞这个学生,这个学生的学号是“020311101”,姓名是“张飞”,性别是“男”,身高 1.72,1992 年出生,籍贯“江苏”,是计算机系的学生,这里的“020311101”“张飞”“男”“1.72”“1992”“江苏”“计算机”就是数据。

表 1.1 学生表

| 学号        | 姓名    | 性别    | 身高    | 出生年份  | 籍贯    | 所在系   |
|-----------|-------|-------|-------|-------|-------|-------|
| 020311101 | 张飞    | 男     | 1.72  | 1992  | 江苏    | 计算机   |
| .....     | ..... | ..... | ..... | ..... | ..... | ..... |
| 020311120 | 王小莉   | 女     | 1.62  | 1991  | 北京    | 管理    |
| .....     | ..... | ..... | ..... | ..... | ..... | ..... |

在实际应用中,将数据分成两种:一种是可以参与数值运算的数值型数据,如表示成绩、身高的数据;另一种是非数值型数据,包括文字、图形、图像、声音等多媒体数据,如表示姓名、性别、籍贯、所在系、照片等的的数据。

信息是数据中所包含的有意义的内容。信息和数据这两个概念既有联系又有区别。数据是按一定的格式对信息进行的符号化表示,是信息的载体;而信息是数据的内涵,是数据的语义解释。

不经过加工处理的数据只是一种原始材料,对人类活动产生不了决策作用,只有经过提炼和加工,原始数据才能变成对人类有用的知识。

一般来说,数据处理过程分为如下 5 个基本环节:原始数据的收集,数据的规范化及其



编码,数据输入,数据处理,数据输出。

数据处理是指将数据转换成信息的过程。

数据处理的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),从而作为决策的依据。

随着计算机的普及发展,现在有很多课程采用了网上考试的形式,并且由计算机自动阅卷。计算机阅卷后得到每位学生的考试成绩,属于原始数据,由考试成绩分析系统对考试成绩进行分析和处理,如按考试成绩从高到低顺序排列、统计各分数段的人数并且用图表的形式显示结果等,输出的数据中包含了有价值、有意义的信息。

### 1.1.2 数据管理技术的发展

数据处理的一个重要方面就是数据管理。数据管理包括数据的分类、组织、编码、存储、检索和维护等操作过程,其基本目的是从大量的、杂乱无章的、难以理解的数据中抽取并导出对那些特定的应用来说有价值的、有意义的数据。数据管理经历了由低级到高级的发展过程,随着计算机硬件、软件技术的发展而不断提高,大体上经历了三个阶段,即人工管理阶段、文件系统阶段以及数据库系统阶段。

#### 1. 人工管理阶段

20 世纪 50 年代中期以前,计算机主要用于科学计算。当时的硬件状况很差,外存储器只有纸带、卡片、磁带等,没有磁盘等直接存取的存储设备,输入的数据和程序都是二进制代码,只有专业人员才能操作,而且极易出错,不便于输入大量数据。软件状况也很不好,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。

人工管理数据具有以下特点:

- 数据不保存。

由于当时计算机主要用于科学计算,数据一般不需要长期保存,只是在计算某一课题需要时将数据输入,用完就可以将数据撤走,而且不仅对用户数据这样处置,有时对系统软件也是这样处置。

- 由应用程序管理数据。

数据需要由应用程序自己管理,没有相应的软件系统负责数据的管理工作,应用程序中不仅要规定数据的逻辑结构,而且要负责设计数据的物理结构,包括存储结构、存取方法、输入方式等。所以人工管理阶段程序员的负担往往很重。

- 数据有冗余,无法实现共享。

数据是面向应用的,一组数据只能对应一个程序。当多个应用程序都需要某些相同的数据时,也必须各自定义,不能互相利用、互相参照,也就是不能共享。所以,程序与程序之间有大量数据冗余。

- 数据不具有独立性。

数据和应用程序相互关联,当数据的逻辑结构或者物理结构发生变化后,必须对应用程序做相应的修改。这也就进一步加重了程序员的负担。

在人工管理阶段,数据与程序一一对应,各个应用程序按计算要求组织各自需要的数据,程序与数据的对应关系如图 1.1 所示。

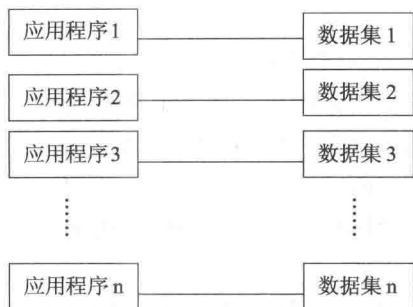


图 1.1 人工管理阶段应用程序与数据之间的对应关系

## 2. 文件系统阶段

20 世纪 50 年代后期至 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,而且逐渐扩大到非计算领域,如用于数据管理。这时硬件方面有了很大改善,已经有了磁盘、磁鼓等直接存取存储设备,尤其是磁盘已经成为联机应用的主要存储设备。在软件方面,有了操作系统和高级语言,而且有了专门的数据管理软件,也就是文件管理系统(或操作系统的文件管理部分),处理方式不仅有了文件批处理,而且能够联机实时处理。

这一时期的数据管理和数据处理有自身的优点,但也仍然存在一些缺点,下面分别介绍。

### (1) 文件系统管理数据的优点

- 数据可以长期保存。

由于计算机应用范围逐渐扩大,大量用于数据处理,数据需要长期保留在外存上,以便于反复进行查询、插入、删除和修改等操作。数据也不再仅仅属于某个特定的程序,而可以由多个程序反复使用。

- 有专门的软件(即文件系统)管理数据。

文件系统把数据组织成相互独立的数据文件。程序和数据之间由软件提供的存取方法进行转换,使数据与应用程序之间有了一定的独立性,程序员可以不必过多地考虑物理细节,可以将精力集中于算法,而且数据在存储上的改变不一定反映在程序上,大大节省了维护程序的工作量。

- 文件的形式多样化。

由于有了磁盘等直接存取存储设备,文件不再局限于顺序文件,有了索引文件、链表文件等,因而对文件的访问可以是顺序访问,也可以是直接访问,但文件之间是独立的,它们之间的联系通过程序去构造。

### (2) 文件系统管理数据的缺点

- 数据共享性差,冗余度大。

在文件系统中,一个文件基本上对应于一个应用程序,也就是说文件仍然是面向应用的。当不同的应用程序具有部分相同的数据时,也必须各自建立自己的文件,而不能共享相同的数据,因此数据的冗余度也就增大了,从而导致存储空间浪费。同时,由于相同数据的重复存储以及各自进行管理,增加了修改数据、维护数据的难度,容易造成数据的不一致。



- 数据独立性差。

文件系统中的文件是为某一特定应用程序服务的,文件的逻辑结构对该应用程序来说是优化的,因此,要想对现有的数据再增加一些应用是很困难的,系统不容易扩充,一旦数据的逻辑结构发生改变,就必须对应用程序及文件结构的定义做出相应的修改。同样,应用程序的修改也会引起文件数据结构的改变,如应用程序改用不同的高级语言,文件的数据结构也就必须做出相应修改。因此数据与程序之间仍缺乏独立性。

- 数据联系弱。

文件与文件之间是独立的,文件之间的联系必须通过程序来构造,可见,文件是一个不具有弹性的、无结构的数据集合,不能反映现实世界事物之间的内在联系。

在文件系统阶段,程序与数据之间的关系如图 1.2 所示。

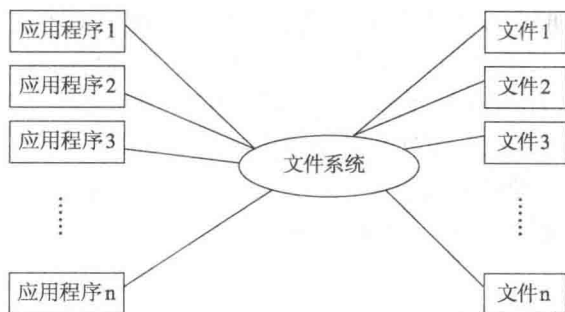


图 1.2 文件系统阶段应用程序与数据文件的关系

### 3. 数据库系统阶段

20 世纪 60 年代后期,计算机用于管理的规模日益庞大,应用也越来越广泛,数据量急剧增长,对数据管理提出了更高的要求,要求数据具有更高的独立性,同时多种应用、多种语言互相覆盖地共享数据集合的要求也越来越强烈。这时计算机硬件和软件技术都有了更大的发展,硬件方面出现了大容量直接存取设备,硬件价格下降;而软件价格则逐渐上升,为编制和维护系统软件及应用程序所需的成本相应增加;在处理方式上,联机实时处理要求更多,并开始提出分布处理的概念和怎样进行分布处理。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广大用户的需求,于是数据库应运而生,解决了多用户、多应用共享数据的要求,使数据为尽可能多的应用服务。

到 20 世纪 80 年代初,随着计算机科学技术的进一步发展,数据库技术和计算机网络、人工智能、软件工程、面向对象技术等相互结合,使数据库进入了高级发展阶段,其标志就是分布式数据库系统和面向对象数据库系统的出现。此后,数据库技术蓬勃发展,并在并行数据库技术、模糊数据库技术等新一代数据库技术与理论方面得到了更大的发展。

数据库技术克服了以前管理方式的缺点,试图提供一种完善的、更高级的数据管理方式,标志着数据库管理技术的飞跃。与人工管理阶段和文件系统阶段相比,数据库系统管理阶段具有数据结构化、共享性高、冗余度低、易扩充、独立性高以及数据由 DBMS 统一管理和控制等特点。

数据库系统阶段数据与应用程序的关系如图 1.3 所示。

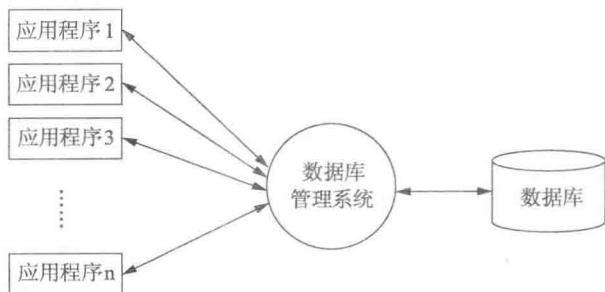


图 1.3 数据库系统阶段应用程序与数据之间的对应关系

## 1.2 数据库系统的组成

数据库系统是指在计算机系统中引入数据库后的系统。数据库系统通常由 5 部分组成：硬件系统、数据库、数据库管理系统及相关软件、数据库管理员（DBA）和用户。

### 1. 计算机硬件

运行数据库系统的计算机需要有足够大的内存、足够大容量的磁盘等联机直接存取设备和较高的通道能力，以及支持对外存的频繁访问，还需要足够数量的脱机存储介质，如软盘、光盘、磁带等存放数据库的备份。

### 2. 计算机软件

数据库系统中的软件包括操作系统、数据库管理系统和数据库应用系统等。

数据库管理系统(Database Management System, 简称 DBMS) 是用于建立、使用和维护数据库的系统软件，是数据库系统的核心。DBMS 能够对数据库中的数据资源进行统一的管理和控制，以保证数据库的安全性和完整性。用户通过 DBMS 访问数据库中的数据，数据库管理员也通过 DBMS 进行数据库的维护工作。DBMS 必须有操作系统和相关系统软件的支持才能运行。

DBMS 是位于用户与操作系统之间的一层数据管理软件，它具有以下功能：

#### (1) 数据定义功能

DBMS 提供数据定义语言(DDL)，通过它可以定义数据库结构和存储结构、定义数据库中数据之间的联系、定义数据完整性约束条件和保证完整性的触发机制等。

#### (2) 数据操纵功能

DBMS 提供数据操纵语言(DML)，通过它可以操纵数据，完成对数据库中数据的输入、查询、插入、删除、修改等基本操作。

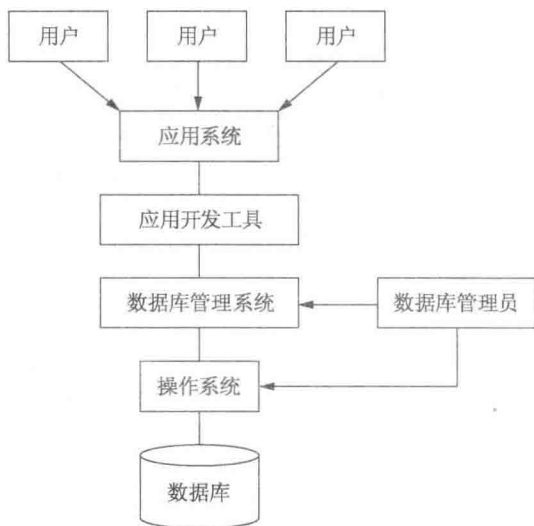


图 1.4 数据库系统



### (3) 数据库运行管理功能

包括多用户环境下的并发控制、安全性检查和存取限制控制、完整性检查和执行、运行日志的组织管理、事务的管理和自动恢复。这些功能保证了数据库系统的正常运行。

### (4) 数据的组织和存储管理

DBMS 要分类组织、存储和管理各种数据,包括数据字典、用户数据、存取路径等。数据组织和存储的基本目标是提高存储空间利用率,选择合适的存取方法提高存取效率。

### (5) 数据库的维护

为数据库管理员提供软件支持,包括数据库的数据载入、转换、转储、数据库备份、数据库重组以及性能监控等维护工具。

目前较流行的数据库管理系统有 Visual FoxPro、Access、Oracle、Sybase 等。

数据库应用系统是指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。

## 3. 数据库

数据库(Database,简称 DB)是一个长期存储在计算机内的、有组织的、可共享的、统一管理的数据集合。

数据库是相互关联的数据的集合。数据库中的数据不是孤立的,数据与数据之间是相互关联的,在数据库中不仅要能够表示数据本身,还要能够表示数据与数据之间的联系。

数据库不仅需要存储用户的数据,还需要存储有关数据的结构描述信息,即元数据。

数据库中的数据按一定的数据模型组织和存储,具有较小的冗余度、较高的数据独立性,并可以供多个用户和多个应用所共享。

## 4. 数据库系统的有关人员

数据库系统的有关人员主要有三类:最终用户、数据库应用系统开发人员和数据库管理员(DBA: Database Administrator)。最终用户指通过应用系统的用户界面使用数据库的人员,他们一般对数据库知识了解不多。数据库应用系统开发人员包括系统分析员、系统设计和程序员:系统分析员负责应用系统的分析,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。

数据库管理员是数据管理机构的一组人员,他们负责对整个数据库系统进行总体控制和维护,以保证数据库系统的正常运行。其主要职责如下:

- 决定数据库中的信息内容和结构。

DBA 要参与数据库的分析和设计,决定数据库中存放哪些信息,并确定数据的逻辑结构和物理结构。

- 确定数据的安全保密要求和完整性约束。

DBA 为了保证数据库的安全性和完整性,要恰如其分地规定各个用户的存取权限、保密级别等。

- 监控数据库的使用和运行。

DBA 要及时处理运行中出现的问题,如故障恢复、数据的定期转储等。

- 数据库的改进和重组、重构。

通过对 DBMS 的运行统计数据进行分析,监视系统的空间利用率和处理效率。当效率下降时,采取适当的措施,如对数据库的重组、重构等。

## 1.3 数据模型

数据库是一个企业、组织、部门或领域所涉及的数据的集合,它不仅要反映数据本身的内容,而且要反映数据之间的联系。由于计算机不能直接处理现实世界中的具体事物,所以必须将现实世界中各种复杂的事物转换成计算机能够处理的数据。

数据模型是描述数据、数据间的联系、数据的语义以及数据一致性约束的概念工具的集合。在数据库技术中,用数据模型这个工具来抽象、表示和处理现实世界中的数据。现有的数据库管理系统均是基于某种数据模型实现的。

### 1.3.1 数据抽象的过程

将现实世界中的客观事物转换成数据库中的数据是一个逐步抽象的过程,这个过程经历了现实世界、信息世界和数据世界三个阶段,首先把现实世界中存在的客观事物通过概念抽象转换成不依赖于具体计算机系统(DBMS)的数据结构(称为概念模型),然后再将其转换成计算机系统中DBMS所支持的数据模型。

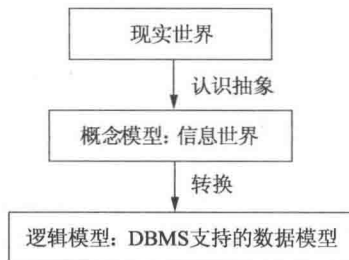


图 1.5 数据抽象的过程

概念模型是按用户的观点对数据建模,是系统用户对整个应用项目涉及的数据的全面描述。概念模型是对现实世界的第一层抽象,是用户和数据库设计人员之间进行交流的工具,与具体的DBMS无关。概念模型通常用于建立信息世界的数据库模型,强调其语义表达能力,因此,模型中的概念应该简单、清晰、易于理解。长期以来,在数据库设计中广泛使用的概念模型是“实体-联系模型(简称E-R模型)”,E-R模型是一种高层数据库模型,用于表示现实世界中实体和实体间的联系。

现实世界中的客观事物及其联系,在数据世界以逻辑模型描述。在选定数据库管理系统之后,就要将E-R图表示的概念模型转换为具体的数据库管理系统支持的逻辑模型。逻辑模型是按计算机实现的观点对数据进行建模,有严格的形式化定义,便于在计算机系统中实现,它是现实世界的第二层抽象。

通常,也把数据的逻辑模型直接称为数据模型。数据库系统中主要的逻辑模型有层次模型、网状模型和关系模型。

目前,层次模型和网状模型已基本退出历史舞台,而关系模型因其简单且有严格的数学定义,已成为当今主要的数据库模型。关系模型是一种底层模型,它是用表的集合来表示数据和数据间的联系。目前流行的数据库管理系统都是基于关系模型的。在数据库设计时,通常是先用E-R模型在高层对数据建模,然后再将其转换为关系模型。

### 1.3.2 概念模型

#### 1. E-R模型中的基本概念

E-R模型中有三个基本的概念:实体、属性和联系。

##### (1) 实体(Entity)

实体是现实世界中任何可以相互区分和识别的事物,且这些事物应该是用户感兴趣的