数据库技术与应用（VFP6.0 版）

主　编：曲桂东

副主编：郝坤、孙思炯、毕燕丽

编　委：尤慧芳、张贤坤、李玉华、邹本强、

陈　辉、刘国林、唐庆晨、王月敏

上海交通大学出版社

内容提要

Microsoft Visual FoxPro6.0 关系数据库系统是新一代小型数据库管理系统的杰出代表，它以其卓越的数据库处理性能、良好的开发环境、完整而又丰富的工具、友好的界面以及完备的兼容性等特点，备受广大用户欢迎。

本书从实例出发，全面细致地介绍了 Visual FoxPro6.0 中文版的使用方法。内容包括数据库基础知识、Visual FoxPro6.0 中文版简介、Visual FoxPro6.0 语言快速入门、数据库及表的设计、查询与视图、创建应用程序、面向对象的程序设计入门、表单、报表、菜单、帮助文件的设计、安装盘的制作及数据的导入、导出操作，最后一章还以简单的图书借阅系统为例讲解了一个管理系统的开发过程。

丰富的图示、流畅的语言和较多的实例是本书一大特色。本书适宜高职、高专的计算机及其相关专业做为教材，也适合于有志青年自学之用。

书　名：数据库技术与应用（VFP6.0）

作　者：曲桂东

出版社：上海交通大学出版社

出版日期：2003

ISBN：7-313-03310-9/TP211.138F0

定 价：26.00

前 言

Microsoft Visual FoxPro6.0 关系数据库系统是新一代小型数据库管理系统的杰出代表，它以其卓越的数据库处理性能、良好的开发环境、完整而又丰富的工具、友好的界面以及完备的兼容性等特点，备受广大用户欢迎。

Visual FoxPro6.0 中文版是可运行于 Windows/95/98/2000/NT/XP 平台的 32 位数据库开发系统，它不仅可以简化数据库管理，而且能使应用程序的开发流程更为合理。用户可通过 Visual FoxPro 的开发环境方便地设计数据库结构、管理数据库、设计应用程序界面和设计查询、报表、菜单，并可利用项目管理器对数据库和程序进行管理、生成可执行文件、进行发布等。

Visual FoxPro6.0 不仅提供了集成化的开发环境，而且还支持过程编程技术，并在语言方面作了强大的扩充，支持面向对象可视化编程技术，并拥有功能强大的可视化程序设计工具。

本书以 Visual FoxPro6.0 中文版为背景，通过大量的实例介绍程序设计的基础、方法，避免枯燥空洞的理论，容易上手，于不知不觉中使读者学会 Windows 环境中的可视化编程。本书图文并茂，所有操作均按实际屏幕显示一步一步讲述，通过范例和实际操作，理解基本概念，学会操作方法。

本书每章均配有习题，填空、选择、简答题可适当选择。特别是大部分章节均配有实验、操作、设计题，这部分题目对培养高职高专学生的动手、开发能力，很有帮助。建议学生在做这些题目的时候，可以几个人为一个小组，共同研究解决其中的问题，这样有助于学生将来在开发项目时，能有较好的分工与协作。

本书可作为高职、高专计算机等相关专业的教材或教学参考书，也可以作为有志青年自学之用。

本书由长期战斗在高职、高专教学第一线的教师们编写，他们是主编：曲桂东，副主编郝坤、孙思炯、毕燕丽，此外还有尤慧芳、李玉华、张贤坤、邹本强、陈辉、刘国林、唐庆晨、王月敏参加编写工作。其中，第 1 章由孙思炯编写，第 2、3 章由尤慧芳编写，第 4 章由

毕燕丽编写，第 5、6、7 章由郝坤、陈辉编写，第 8 章由刘国林、唐庆晨编写，第 9、10、11 章由李玉华编写，第 12 章由邹本强、王月敏编写，13、14 章由曲桂东编写，在初稿完成后，郝坤、李玉华、孙思炯、毕燕丽参加了统稿工作，最后由曲桂东审阅定稿。在此书编写过程中,得到了上海交大出版社东鲁红编辑及各位参编老师所在院校领导的大力支持与协助，在此一并表示衷心地感谢。

由于水平所限，书中难免有错误和不当之处，望读者不吝赐教。

E-mail:qugd@sina.com

编者

2002 年 10 月 28 日

目　　录

第 1 章 数据库基础知识.....1

1.1 信息、数据和数据处理.....1

1.1.1 信息与数据.....1

1.1.2 数据处理.....2

1.2 数据模型.....2

1.3 数据库应用系统.....5

1.3.1 数据库.....5

1.3.2 关系数据库.....5

1.3.3 数据的规范化.....6

1.4 数据库管理系统.....9

1.5 数据库系统的体系结构.....9

1.6 数据库应用系统的构成.....10

习题一.....11

第 2 章 Visual FoxPro 6．0 中文版简介.....13

2.1FoxPro 的发展历史.....13

2.2Visual FoxPro6.0 中文版的特点及新增功能.....14

2.2.1 Visual FoxPro 6.0 的特点.....14

2.2.2 Visual FoxPro 6.0 的新增功能.....14

2.3Visual FoxPro6.0 常用文件类型.....16

2.4 Visual FoxPro6.0 性能指标.....17

2.4.1 表文件及索引文件的指标.....17

2.4.2 字段有关指标.....17

2.4.3 内存变量与数组的指标.....18

2.4.4 程序和过程文件有关指标.....18

2.4.5 “报表设计器” 的指标.....18

2.4.6 其他指标.....18

2.5 Visual FoxPro6.0 安装与运行.....18

2.5.1 Visual FoxPro 6.0 安装.....19

2.5.2 Visual FoxPro 6.0 运行.....21

2.5.3 Visual FoxPro 6.0 的用户界面.....23

2.6 Visual FoxPro6.0 菜单.....24

2.6.1 “文件”菜单.....24

2.6.2 “编辑”单.....25

2.6.3 “显示”菜单.....26
2.6.4 “格式”菜单.....26
2.6.5 “工具”菜单.....27
2.6.6 “程序”菜单.....28
2.6.7 “窗口”菜单.....28
2.6.8 “帮助”菜单.....28
2.6.9 工具栏.....29
习题二.....30
第3章 Visual FoxPro6.0语言快速入门.....32
3.1 Visual FoxPro6.0的数据类型.....32
3.2 操作符与表达式.....34
3.2.1 字符型表达式.....34
3.2.2 数值型表达式.....35
3.2.3 日期型表达式.....35
3.2.4 逻辑型表达式.....35
3.3 变量与数组.....36
3.3.1 变量.....36
3.3.2 数组.....37
3.4 过程与函数.....38
3.4.1 过程.....38
3.4.2 函数.....39
3.5 常用命令分类与用法.....39
3.5.1 调试、显示常用命令.....39
3.5.2 数据库(表)操作命令.....41
3.6 常用函数分类与用法.....49
3.7 常用程序结构.....53
3.7.1 顺序结构.....53
3.7.2 分支结构.....53
3.7.3 循环结构.....56
习题三.....60
第4章 表及索引.....65
4.1 建立表文件.....65
4.1.1 定义表中数据类型.....65
4.1.2 用表设计器创建表.....66
4.1.3 用表设计器修改表结构.....68
4.1.4 利用表向导创建表.....69
4.2 数据的输入与修改.....72

4.2.1 数据的输入……...73
4.2.2 数据的修改……...77
4.3控制字段和记录的访问……77
4.3.1 限制对字段的访问……...77
4.3.2 限制对记录的访问……...79
4.4表中记录的定位……79
4.5记录的删除与恢复……82
4.5.1 删除一条记录……...82
4.5.2 删除多条记录……...83
4.5.3 恢复删除的记录……...85
4.5.4 彻底删除记录……...86
4.6浏览数据表……87
4.6.1 浏览窗口的显示模式……...87
4.6.2 定制浏览窗口……...87
4.7建立索引……89
4.7.1 何谓索引……...90
4.7.2 索引类型……...90
4.7.3 建立索引……...91
4.7.4 索引的使用……...92
习题四……..93
第5章 设计、管理数据库和表……..97
5.1 数据库向导和数据库设计器……..97
5.1.1 使用数据库向导建立数据库……97
5.1.2 用数据库设计器设计数据库……99
5.1.3 “数据库设计器”工具栏介绍……99
5.2 设计数据库……100
5.2.1 数据库的设计步骤……..100
5.2.2 向数据库中添加和删除表……..101
5.3 表的字段特性……102
5.3.1字段的标题与注释……102
5.3.2字段默认值……104
5.3.3数据有效性规则……105
5.4 管理和维护数据库表……106
5.4.1为数据库创建索引……106
5.4.2 为数据库表建立关系……..107
5.4.3 数据库中数据排序……..109
5.4.4 数据库中数据的增加、删除、清除等操作……109
习题五……110

第 6 章 查询与视图.....113
6.1 创建查询.....113
6.1.1 查询向导.....113
6.1.2 查询设计器.....116
6.2 查询的输出和运行.....118
6.3 SELECT SQL 命令.....120
6.4 本地视图的简单应用.....120
6.5 建立用户数据库视图.....121
6.6 使用视图访问数据库.....123
6.6.1 数据库中视图的操作.....123
6.6.2 利用视图更新数据.....124
6.7 远程视图向导简介.....126
习题六.....128
第 7 章 创建应用程序.....130
7.1 项目管理器简介.....130
7.1.1 使用应用程序向导.....130
7.1.2 项目管理器简介.....131
7.2 创建项目文件.....132
7.3 向项目中增加资源.....133
7.4 项目连编.....134
7.5 关于项目信息和项目文档.....136
习题七.....136
第 8 章 面向对象程序设计入门.....138
8.1 面向对象程序设计的基本概念.....138
8.1.1 对象.....138
8.1.2 类.....139
8.1.3 属性.....141
8.1.4 事件与方法.....141
8.2 设计类.....142
8.2.1 创建类.....142
8.2.2 类属性的定义.....143
8.2.3 类的方法和事件的定义.....145
8.2.4 通过编程定义类.....146
8.3 对象的操作.....147
8.3.1 由类创建对象.....148
8.3.2 设置属性.....148

8.3.3 调用对象的方法和触发对象的事件.....149
8.3.4 对象设计实例.....149
习题八.....153
第 9 章 创建表单.....154
9.1 用表单向导设计表单.....154
9.2 表单设计器介绍.....156
9.2.1 表单设计器的基本操作.....157
9.2.2 表单举例.....158
9.3 表单常用属性及事件.....160
9.3.1 表单常用属性.....160
9.3.2 表单常用事件.....160
9.4 表单控件.....161
9.4.1 标签.....162
9.4.2 图象.....162
9.4.3 线条.....162
9.4.4 形状.....163
9.4.5 文本框.....164
9.4.6 编辑框.....165
9.4.7 微调控件.....165
9.4.8 列表框.....165
9.4.9 组合框.....169
9.4.10 命令按钮.....170
9.4.11 命令按钮组.....171
9.4.12 复选框.....173
9.4.13 选项按钮组（OptionGroup）.....174
9.4.14 计时器（Timer）.....175
9.4.15 表格（Grid）.....175
9.4.16 页框（PageFrame）.....179
9.4.17 ActiveX 控件（Active Control）.....180
9.4.18 ActiveX 绑定控件（Active Bound Control）.....183
9.4.19 超级链接.....183
习题九.....184
第 10 章 设计报表.....186
10.1 报表设计器.....186
10.1.1 快速报表.....186
10.1.2 报表带区.....188
10.1.3 报表控件.....191

10.1.4 报表设计器工具栏.....195
10.1.5 页面设置.....196
10.1.6 报表输出.....198
10.2 综合举例.....199
10.2.1 一对多报表.....199
10.2.2 在程序中建立数据环境.....203
习题十.....207
第11章 设计菜单.....209
11.1 菜单设计器.....209
11.1.1 打开菜单设计器.....209
11.1.2 菜单设计器窗口.....210
11.1.3 保存菜单.....212
11.1.4 生成菜单.....213
11.1.5 运行菜单.....213
11.2 快捷菜单.....214
11.3 菜单命令.....215
11.3.1 下拉式菜单命令.....215
11.3.2 弹出式菜单命令.....216
习题十一.....217
第 12 章 设计帮助文件.....218
12.1 创建图形样式帮助文件.....218
12.1.1 HTML 帮助样式.....218
12.1.2 访问在线 HTML 样式帮助.....219
12.1.3 发布编译 HTML 样式帮助系统.....220
12.1.4 关于 WinHelp4.0.....221
12.2 设计.DBF 样式帮助文件.....221
12.2.1 设计.DBF 样式帮助.....221
12.2.2 帮助表的要求.....222
12.2.3 使用.DBF 样式帮助.....222
习题十二.....224
第 13 章 制作安装盘及数据的导入与导出.....225
13.1 安装向导介绍.....225
13.1.1 定位文件.....225
13.1.2 指定组件.....226
13.1.3 磁盘映象.....228
13.1.4 安装选项.....228

13.1.5默认目标目录……229
13.1.6改变文件设置……229
13.1.7完成……231
13.2 使用安装向导……231
13.3 检测安装程序……232
13.4数据的导入……233
13.4.1 数据导入……233
13.4.2 数据追加……237
13.5数据的导出……239
13.5.1 启动数据“导出”对话框……239
13.5.2设置导出选项……239
13.5.3完成操作……240
习题十三……241
第14章 应用举例……242
14.1系统开发的一般过程……242
14.2 简易图书管理系统开发……242
14.2.1 系统总体设计及开发任务……243
14.2.2 设计表结构……243
14.2.3 建立数据库及数据表……244
14.2.4 建立视图……247
14.2.5 建立读者信息表单……250
14.2.6 建立图书信息表单……251
14.2.7 建立个人借阅情况查询表单……252
14.2.8 建立打印报表……253
14.2.9 建立菜单系统……255
14.2.10 数据库的运行……257
习题十四……257

第 1 章 数据库基础知识

计算机作为具有极高的运算速度、巨大的数据存储能力、可以准确地进行各种算术运算和逻辑运算的现代化计算工具，已被广泛地应用于各个领域。计算机技术的高速发展被认为是人类进入信息时代的标志。在信息时代，人们需要对大量的信息进行加工处理，在这一过程中应用数据库技术，一方面促进了计算机技术的高速发展，另一方面也形成了专门的信息处理理论及数据库管理系统。从某种意义上说，数据库管理系统正是计算机技术和信息时代相结合的产物，它是信息处理或数据处理的核心，是研究数据共享的一门科学，是计算机科学的一个重要分支。本章将从数据库基本概念出发，逐一讲解信息、数据、数据处理、数据模型、数据库、数据库设计等基础知识和概念。

1.1 信息、数据和数据处理

在数据处理这一计算机应用领域，人们首先遇到的基本概念是信息和数据，它们是两个不同的术语，却有着密不可分的联系。

1.1.1 信息与数据

信息（information）是客观事物属性的反映。它所反映的是关于某一客观系统中某一事物的某一方面属性或某一时刻的表现形式。通俗地讲，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。

数据（data）是反映客观事物属性的记录，是信息的载体。对客观事物属性的记录是用一定的符号来表达的，因此说数据是信息的具体表现形式。数据所反映的事物属性是它的内容，而符号是它的形式。

数据与信息在概念上是有区别的。从信息处理角度看，任何事物的属性都是通过数据来表示的，数据经过加工处理后，使其具有知识性并对人类活动产生决策作用，从而形成信息。用数据符号表示信息，其形式通常有三种：数值型数据，即对客观事物进行定量记录的符号，如体重、年龄、价格的多少等；字符型数据，即对客观事物进行定性记录的符号，如姓名、单位、地址的标志等；特殊型数据，如声音、视频、图像等。从计算机的角度看，数据泛指那些可以被计算机接受并能够被计算机处理的符号。

总之，信息是有用的数据，数据是信息的表现形式。信息是通过数据符号来传播的，数据如不具有知识性和有用性，则不能称其为信息。

1.1.2 数据处理

数据处理也称为信息处理。所谓数据处理实际上就是利用计算机对各种类型的数据进行处理。它包括对数据的采集、整理、存储、分类、排序、检索、维护、加工、统计和传输等一系列操作过程。数据处理的目的是从大量的、原始的数据中获得我们所需要的资料并提取有用的数据成份，作为行为和决策的依据。

随着电子计算机软件和硬件技术的发展，数据处理过程发生了划时代的变革，而数据库技术的发展，又使数据处理跨入了一个崭新的阶段。数据管理技术的发展大致经历了从人工管理方式、文件管理方式到数据库系统管理方式三个阶段。

人工管理方式出现在计算机应用于数据管理的初期。由于没有必要的软件、硬件环境的支持，用户只能直接在裸机上操作。用户的应用程序中不仅要设计数据处理的方法，还要阐明数据在存储器上的存储地址。在这一管理方式下，用户的应用程序与数据之间的相互结合不可分割，当数据有所变动时程序则随之改变，独立性差；另外，各程序之间的数据不能相互传递，缺少共享性，因而这种管理方式既不灵活，也不安全，编程效率极差。

文件管理方式即把有关的数据组织成一种文件,这种数据文件可以脱离程序而独立存在，由一个专门的文件管理系统实施统一管理。文件管理系统是一个独立的系统软件，它是应用程序与数据文件之间的一个接口。在这一管理方式下，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。应用程序的数据具有一定的独立性，也比手工管理方式前进了一步。但是，数据文件仍高度依赖于其对应的程序，不能被多个程序所通用，由于数据文件之间不能建立任何联系，因而数据的通用性仍然较差，冗余量大。

数据库系统管理方式即对所有的数据实行统一规划管理，形成一个数据中心，构成一个数据仓库，数据库中的数据能够满足所有用户的不同要求，供不同用户共享。在这一管理方式下，应用程序不再只与一个孤立的数据文件相对应，可以取整体数据集的某个子集作为逻辑文件与其对应，通过数据库管理系统实现逻辑文件与物理数据之间的映射。在数据库系统管理的系统环境下，应用程序对数据的管理和访问灵活方便，而且数据与应用程序之间完全独立，使程序的编制质量和效率都有所提高；由于数据文件之间可以建立关联关系，使得数据的冗余大大减少，数据共享性显著增强。

1.2 数据模型

现实世界中的客观事物是彼此相互联系的。一方面，某一事物内部的诸因素和诸属性根据一定的组织原则相互具有联系，构成一个相对独立的系统；另一方面，某一事物同时也作为一个更大系统的一个因素或一种属性而存在，并与系统的其他因素或属性发生联系。客观事物的这种普遍联系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性。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。

从理论上讲，数据模型是指反映客观事物及客观事物之间联系的数据组织结构和形式。

客观事物是千变万化的，各种客观事物的数据模型也是千差万别的，但也有其共同性。常用的数据模型有如下三种。

一、层次模型

层次模型（hierarchical model）表示数据间的从属关系结构，是一种以记录某一事物的类型为根结点的有向树结构。层次模型像一棵倒置的树，根结点在上，层次最高；子结点在下，逐层排列。其主要特征如下：

1．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。

层次模型表示的是从根结点到子结点的一个结点对多个结点，或从子结点到父结点的多个结点对一个结点的数据间的联系。层次模型的示例如图 1-1 所示。

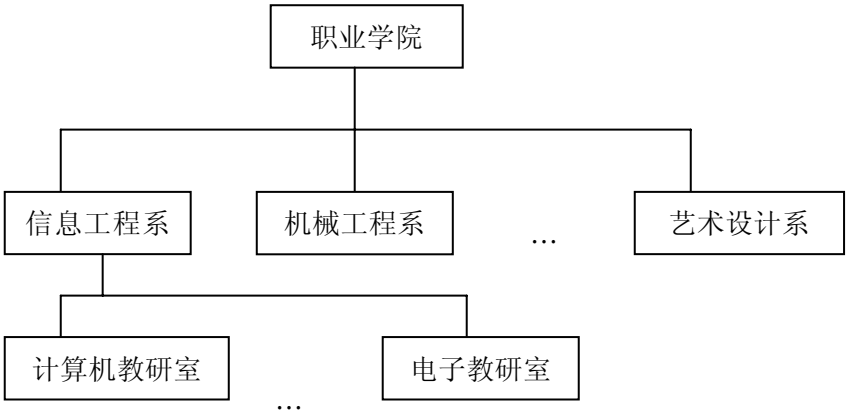


图 1-1　层次模型示例

二、网状模型

网状模型（network model）是层次模型的扩展，它表示多个从属关系的层次结构，呈现一种交叉关系的网络结构。网状模型是以记录为结点的网络结构。其主要特征如下：

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。

网状模型可以表示较复杂的数据结构，即可表示数据间的纵向关系与横向关系。这种数据模型在概念上、结构上都比较复杂，操作上也有很多不便。网状模型的示例如图 1-2 所示。

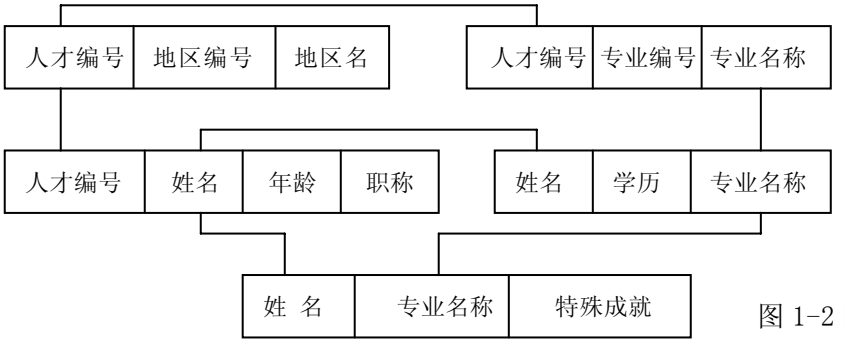


图 1-2 网状模型示例

三、关系模型

关系模型（relational model）的所谓“关系”是有特定含义的，广义地说，任何数据模型都描述一定事物数据之间的关系。层次模型描述数据之间的从属层次关系；网状模型描述数据之间的多种从属的网状关系。关系模型的所谓“关系”虽然也适用于这种广义的理解，但同时又特指那种虽具有相关性而非从属性的平行的数据之间按照某种序列排列的集合关系。

例如，有数据记录如下：甲，女，20 岁；乙，男，21 岁；丙，女，19 岁；丁，男，18 岁。这四组数据之间是平行的，从层次从属角度看也是无关系的，但假如我们知道他们是同一个班级的学生，就可以建立一个关系（一张二维表），如表 1-1 所示。

表 1-1 某班级学生基本情况表

姓 名	性 别	年 龄
甲	女	20
乙	男	21
丙	女	19
丁	男	18

表中的这些数据虽然是平行的，不代表从属关系，但它们构成了某班级学生的属性关系结构。同样，也可以假设上述四组数据不是来自同一个班级，他们所学习的专业也不同，如此便构成了某学校学生基本情况表，又可以建立一个关系（一张二维表），如表 1-2 所示。

表 1-2 某学校学生基本情况表

系别	专业	姓名	性别	年龄
信息工程系	计算机	甲	女	20
信息工程系	软件工程	乙	男	21
信息工程系	信息管理	丙	女	19
经济管理系	国际贸易	丁	男	18

以上两表其实就是关系模型。表格中的每一数据都可看成独立的数据项，它们共同构成了该关系的全部内容。

表格中的每一横行称为一个记录（record）。记录用来表示关系模型中若干平行的、相对独立的个体事物的属性（attribute），每一记录由若干数据项组成，横向排列该事物的诸种属性。

表格中的每一竖列称为一个字段（field）。字段表示关系模型中全部数据项（属性）的类型，每一字段由若干按照某种界域划分的相同类型的数据项组成，竖向列出其诸种类型（属性），一般在表格的第一行（即每一列的最上面）标示属性类型的名称，即字段名。

从总体上说，以竖向的数据项（属性）分类的若干个记录的集合，构成一个关系模型，或称为一个关系（ relation）。在某种意义上可以说，关系模型就是一张二维表，用来描述客观事物属性的关系。

关系模型的主要特点有：

1．关系中每一数据项不可再分，是最基本的单位；
2．每一竖列数据项是同属性的。列数根据需要而设，且各列的顺序是任意的；
3．每一横行记录由一个个体事物的诸多属性项构成。记录的顺序可以是任意的；
4．一个关系是一张二维表，不允许有相同的字段名，也不允许有相同的记录行。

1.3 数据库应用系统

数据库应用系统是一个复杂的系统，它是由硬件、操作系统、数据库管理系统、编译系统、用户应用程序和数据库组成。

1.3.1 数据库

数据库是数据库应用系统的核心和管理对象。前面介绍的数据模型是对数据库如何组织的一种模型表示，它不仅包括客观事物本身的信息，还包括各事物间的联系。数据模型的主要特征是其数据结构，因此数据模型的确定，就等于确定了数据间的关系，即数据库的框架。有了数据间的关系框架，再把表示客观事物具体特征的数据装入框架中，就形成了数据库。

所谓数据库，就是以一定的组织方式将相关的数据组织在一起存放在计算机存储器上形成的、能为多个用户共享的、与应用程序彼此独立的一组相关数据的集合。

数据库的性质是由其中的数据模型决定的。在数据库中的数据如果依照层次模型进行数据存储，则该数据库为层次数据库；如果依照网络模型进行数据存储，则该数据库为网络数据库；如果依照关系模型进行数据存储，则该数据库为关系数据库。Visual FoxPro 数据库管理系统所管理的数据，都是依照关系模型进行存储的，因此其数据库为关系数据库。

1.3.2 关系数据库

关系数据库(relation database)是若干个依照关系模型设计的数据表文件的集合。也就是说，关系数据库是由若干张完成关系模型设计的二维表组成的。

与文件系统的数据文件不同，我们称一张二维表为一个数据表，数据表包含数据及数据间的关系。一个关系数据库由若干个数据表组成，数据表又由若干个记录组成，而每一个记录是由若干个以字段属性加以分类的数据项组成的。

在关系数据库中，每一个数据表都具有相对的独立性，这一独立性的唯一标志是数据表的名字，称为表文件名。也就是说，每一个数据表是靠自身的文件名与其他文件保持独立，一个文件名代表一个独立的表文件。数据库中不允许有重名的数据表，因为对数据表中的数据的访问首先是通过表文件名来实现的。关系数据库中各个数据表的独立性，使用户在使用数据表中的数据时，可以简捷、方便地存取和传输。

在关系数据库中，有些数据表之间是具有相关性的。数据表之间的这种相关性是依靠每一个独立的数据表内部具有相同属性的字段建立的。一般地，一个关系数据库中会有许多独立的数据表是相关的，这为数据资源实现共享及充分利用，提供了极大的方便。

关系数据库由于以具有与数学方法相一致的关系模型设计的数据表为基本文件，不但每个数据表之间具有独立性，而且若干个数据表间又具有相关性，这一特点使其具有极大的优越性，并能得以迅速普及。

关系数据库具有以下特点：

1．以面向系统的观点组织，使数据具有最小的冗余度，支持复杂的数据结构；

2．具有高度的数据和程序的独立性，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；

3．由于数据具有共享性，使数据库中的数据能为多个用户服务；

4．关系数据库允许多个用户同时访问，同时提供了各种控制功能，保证数据的安全性、完整性和并发性控制。安全性控制可防止未经允许的用户存取数据；完整性控制可保证数据的正确性、有效性和相容性；并发性控制可防止多用户并发访问数据时由于相互干扰而产生的数据不一致。

1.3.3 数据的规范化

数据以什么样的结构存入到关系数据库中是 Visual FoxPro 最重要的操作之一，它是应用程序开发的关键。通常，我们首先把收集来的数据存储在一个二维表中，但是有许多相关的数据集合到一个二维表后，数据的关系会变得很复杂，表中的字段个数和数据数量很大，很多时候为了使一个事物表达清楚会有大量数据重复的现象。特别是在进行应用程序设计时，用户组织的数据表如不理想，轻者会大大增加编程及维护程序的难度，重者会使应用程序无法实现。

一个组织良好的数据表，不仅可以方便地解决应用问题，还可以为解决一些不可预测的问题带来便利，同时可以大大加快编程的速度。这就要求数据库中的数据实现规范化，只有实现数据规范化，才能使数据库形成组织良好的局面。

关系模型是以关系集合理论中重要的数学原理为基础的，通过创建某一关系中的规范化准则，既可以方便数据库中数据的处理，又可以给程序设计带来方便。这一规范化准则称为数据规范化（data normalization）。

关系模型的规范化理论是研究如何将一个不好的关系模型转化为一个好的关系模型的理论，它是围绕范式而建立的。

规范化理论认为，关系数据库中的每一个关系都要满足一定的规范。根据满足规范条件不同，可以化分为五个等级，分别称为第一范式（1NF），第二范式（2NF）……第五范式（5NF），其中，NF 是（normal form）的缩写。通常在解决一般性问题时，只要把数据规范到第三个范式标准就可以满足需要。

需要特别指出的是，在实际操作中，不是数据规范的等级越高就越好，具体问题还要具体分析。

关系模型规范化的三条原则如下：

第一范式：在一个关系中，消除重复字段，且各字段都是不可分的基本数据项；

第二范式：若关系模型属于第一范式，则关系中每一个字段都完全依赖于主关键字段的每一部分；