

Visual FoxPro 程序设计



◆ 岳海云 主编 ◆

Visual FoxPro

 甘肃人民出版社

图书在版编目(CIP) 数据

Visual FoxPro程序设计 / 岳海云主编. -- 兰州 :
甘肃人民出版社, 2014.12
ISBN 978-7-226-04701-9

I. ①V… II. ①岳… III. ① 关系数据库系统—程序
设计 IV. ①TP311.138

中国版本图书馆CIP数据核字 (2014) 第 282320 号

出 版 人: 吉西平

责任编辑: 牟克杰

封面设计: 党 杰

Visual FoxPro程序设计

岳海云 主编

甘肃人民出版社出版发行

(730030 兰州市读者大道568号)

甘肃北辰印务有限公司印刷

开本797毫米×1092毫米 1/16 印张19 字数549千

2015年11月第1版 2015年11月第1次印刷

印 数: 1~500

ISBN 978-7-226-04701-9 定 价: 46.00 元

前 言

随着信息化社会的来临,作为信息化社会主要技术标志的计算机技术、知识和应用能力已成为当代人特别是当代大学生知识和能力的重要组成部分。在这种形势下,数据库管理技术作为信息存储和信息加工与处理的主要技术,自然也越来越受到重视。

在数据库应用技术领域,Visual FoxPro 6.0 中文版是适用于微型计算机系统的最优秀的小型关系型数据库管理系统,同时它也是一种程序设计语言。它具有较高的处理速度、友好的界面以及开发成本低、简单易学、方便用户等特点。Visual FoxPro 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,它提供了一个集成化的开发环境,既支持面向过程的编程技术,也支持面向对象的可视化编程技术,并拥有功能强大的可视化程序设计工具。

本书是本人参照《全国计算机等级考试大纲》(二级 Visual FoxPro)的要求结合本人多年的教学实践经验编写而成的。

本书从 Visual FoxPro 6.0 数据库的开发环境以及基本操作入手,结合了大量的数据库应用实例,深入浅出、通俗易懂,注重实用性和可操作性,比较系统地介绍了数据库基础、Visual FoxPro 系统初步、数据库、表及索引的操作、查询与视图、程序设计、表单和报表等。

本书主要介绍 Visual FoxPro 6.0 数据库的基础知识和操作方法。全书共计十一章,第一章是 Visual FoxPro 数据库基础知识,主要介绍数据库管理系统的基本概念和基本理论及 Visual FoxPro 的初步知识。第二章是 Visual FoxPro 关系数据库系统,首先介绍界面组成、功能和工作方式,其次介绍项目管理器以及系统提供的可视化设计辅助工具。第三章着重说明数据与数据运算技术。第四章集中论述 Visual FoxPro 数据库和表及其操作方法,以及数据表结构的创建和数据表记录的添加、维护、排序、索引等方面的内容。第五章重点介绍结构化查询语言 SQL,包括数据查询、数据定义、数据操纵和数据控制功能。第六章简略说明查询和视图的概念、建立和使用。第七章集中介绍面向过程的程序设计方法,包括程序与程序文件、程序的基本结构、多模块程序以及程序调试的方法和步骤。第八章重点介绍面向对象的基本概念和程序设计方法及 Visual FoxPro 中的基类、表单的创建和管理、表单设计器环境以及一些常用的表单控件。第九章是菜单设计,简单介绍 Visual FoxPro 系统菜单的基本情况,然后依次介绍下拉式菜单与快捷菜单的设计。第十章简单介绍报表设计,各种报表的创建和设计方法。第十一章通过实例介绍开发一个数据库应用系统的全过程。

本书适合作为高等院校非计算机专业公共基础课程方面的教材,也可作为计算机等级考试的培训教材以及自学人员自学参考用书。

限于作者水平,书中遗漏和不妥之处敬请读者批评指正。

编 者

2014 年 12 月

目 录

第一章 Visual FoxPro 数据库基础	1
1.1 数据库基础知识	1
1.2 关系数据库	8
1.3 数据库设计基础	10
1.4 Visual FoxPro 系统概述	13
习题一	15
第二章 Visual FoxPro 关系数据库系统	16
2.1 Visual FoxPro 6.0 系统运行环境、启动与退出	16
2.2 Visual FoxPro 6.0 的工作界面	16
2.3 项目管理器	24
2.4 Visual FoxPro 辅助工具	31
习题二	38
第三章 数据与数据运算	39
3.1 数据类型	39
3.2 常量	40
3.3 变量	42
3.4 表达式	49
3.5 常用函数	54
习题三	63
第四章 Visual FoxPro 数据库及其操作	65
4.1 Visual FoxPro 数据库及其建立	65
4.2 使用数据库	67
4.3 建立数据库表	71
4.4 表结构的基本操作	81
4.5 表记录的基本操作	85
4.6 排序和索引	96

4.7 数据完整性	103
4.8 自由表	108
4.9 多个表的同时使用	109
习题四	112
第五章 关系数据库标准查询语言 SQL	115
5.1 SQL 概述	115
5.2 SQL 查询功能	116
5.3 操作功能	127
5.4 定义功能	130
5.5 视图	133
习题五	138
第六章 查询与视图	140
6.1 查询	140
6.2 视图	155
习题六	166
第七章 程序设计基础	168
7.1 程序与程序文件	168
7.2 简单的输入输出命令	172
7.3 程序的基本结构	176
7.4 过程与过程调用	189
7.5 程序调试	194
习题七	198
第八章 表单设计	200
8.1 面向对象的概念	200
8.2 Visual FoxPro 基类简介	201
8.3 表单的创建与运行	205
8.4 表单设计器	212
8.5 表单属性和方法	219
8.6 常用控件	221
习题八	233
第九章 菜单设计与应用	235
9.1 Visual FoxPro 系统菜单	235
9.2 菜单的设计	237
9.3 快捷菜单设计	248
习题九	251

第十章 报表设计	252
10.1 创建报表文件	252
10.2 利用快速报表创建报表	259
10.3 利用报表设计器创建报表	262
10.4 数据分组和多栏报表	269
10.5 报表输出	273
习题十	274
第十一章 应用程序开发实例	276
11.1 数据库应用系统的开发	276
11.2 学生管理系统功能及运行效果	277
11.3 实例——学生管理系统的开发	281
习题十一	291

第一章 Visual FoxPro 数据库基础

Visual FoxPro 是新一代小型数据库管理系统的杰出代表，它采用了可视化的、面向对象的程序设计方法，大大简化了应用系统的开发过程，并提高了系统的模块性和紧凑性。Visual FoxPro 以其开发成本低、简单易学、方便用户等优点得到广泛应用。

数据库管理系统属于数据库管理技术，是作为一门数据处理技术发展起来的，经过 40 多年的发展，理论体系完善，应用广泛，不仅应用于事务处理，并且进一步应用到人工智能、情报检索、计算机辅助设计等各个领域，是计算机科学的核心技术和重要基础。

本章主要介绍数据库管理系统的基本概念和基本理论及 Visual FoxPro 的初步知识。

1.1 数据库基础知识

数据库管理系统是对数据库进行管理的技术，是数据处理的有效工具。数据库是把大量的数据按一定的结构存储在计算机的外存储设备上，类似存放数据的仓库。数据处理的基本问题是为数据的组织、存储、检索、维护和加工利用提供操作手段。

1.1.1 计算机数据管理的发展

1. 数据和数据处理

数据是指存储在某一媒体上能够识别的物理符号。是一切文字、符号、声音、图像等有意义的组合。数据包括描述事物特性的数据内容和存储在某一媒体上的数据形式两个内容。在数据内容一定的情况下，数据的形式可以是多种多样的，例如一个人的出生日期可以用多种方式表示。

数据处理是指数据加工并转换成信息的过程。信息是现实世界在人脑中的抽象反映，是经过加工处理的有用数据，信息是以数据的形式表示。通常所说的“信息处理”，其真正含义是为了产生信息而处理数据。通过处理数据可以获得信息，通过分析和筛选信息可以产生决策。

在计算机中，使用计算机外存储器来存储数据；通过计算机软件来管理、加工、处理和分析数据。

2. 计算机数据管理及发展

计算机数据处理的核心问题是数据管理。计算机对数据的管理是指如何对数据组织、分类、编码、存储、检索和维护提供操作手段。

计算机在数据管理方面经历了由低级到高级的发展过程。计算机数据管理随着计算机软件和硬件技术的发展，数据管理技术的发展大致经历了人工管理、文件系统、数据库系统、分布式数据库系统和面向对象数据库系统 5 个阶段，如图 1.1 所示。

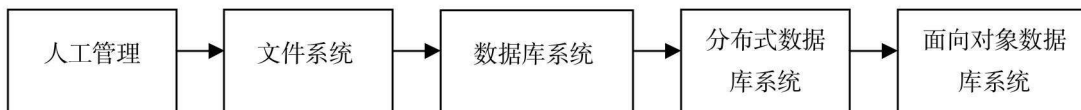


图 1.1 计算机数据管理发展阶段

(1) 人工管理

20 世纪 50 年代中期以前，计算机主要用于科学计算。计算机的软硬件均不完善。硬件的外存

储器只有纸带、卡片、磁带，没有直接存储设备，也还没有操作系统和管理数据的软件。

此阶段数据管理的特点：数据与程序不具有独立性，一组数据对应一组程序。数据不能长期保存，由计算或处理它的程序自行携带，程序运行结束后就退出计算机系统，一个程序中的数据不能被其它程序使用，因此程序与程序之间存在着大量重复数据即冗余数据。

（2）文件系统

20 世纪 50 年代后期至 60 年代中后期，计算机技术大量应用于管理中。在硬件上出现了磁鼓、磁盘等直接存取数据的存储设备；在软件方面，出现了操作系统和专门的数据管理软件——文件系统；处理方式上不仅有了文件批处理，而且能够联机实时处理。

在文件系统阶段：数据与程序有了一定的独立性，程序和数据分开存储，有了数据文件和程序文件的区分。数据文件可以长期保存在外存储器上被多次存取，可以实现以文件为单位的数据共享。

但数据的组织仍然是面向应用程序，数据和程序相互依赖，同一数据可以重复出现在多个文件中，导致数据共享性差、冗余大，占用存储空间，增加更新开销。另外，数据的逻辑结构和应用程序的改变相互影响，数据与程序缺乏独立性。

（3）数据库系统

20 世纪 60 年代后期，计算机性能得到提高，存储容量大大增加且价格下降，需要计算机处理和统一管理的数据资源也越来越多，并且对数据共享的需求日益增强。显然文件系统作为数据管理的手段已经不能满足应用的需要，为了解决多用户、多应用共享数据的需求，发展了数据库技术和统一管理的数据的专门软件系统——数据库管理系统（DataBase Management System, DBMS）。

数据库系统阶段：数据和程序完全分开，数据可以供多个程序使用，从而实现有效管理和存取大量数据资源的目的，主要包括：使不同用户能同时访问数据库中的数据；减少数据的冗余，以提高数据的一致性和完整性；提供数据与应用程序的独立性，从而减少应用程序的开发和维护费用。

数据库管理系统利用了操作系统提供的输入/输出控制和文件访问功能，因此，它需要在操作系统的支持下运行。Visual FoxPro 就是一种在微机上运行的数据库管理系统软件。在数据库管理系统的支持下，数据和程序的关系如图 1.2 所示。

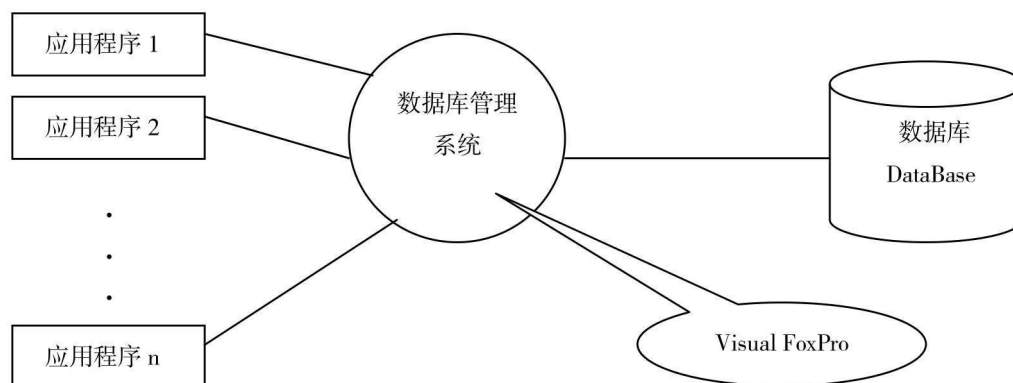


图 1.2 数据与程序的关系

（4）分布式数据库系统

分布式数据库系统是数据库技术与计算机网络技术紧密结合的产物。20 世纪 70 年代后期之前，大多数数据库系统是集中式的。随着网络技术的发展为数据库提供了分布式运行环境，从主机-终端系统结构发展为客户/服务器（client/server）系统结构。分布式数据库系统适合于单位分散的部门，

其分为以下两种数据库结构。

物理上分布、逻辑上集中的分布式数据库结构：是一个逻辑上统一、地域上分布的数据集合，即把全局数据模式按数据来源和用途，合理分布在系统的多个节点上，是大部分数据可以就地或就近存取，而用户不感到数据的分布。

物理上分布、逻辑上分布的分布式数据库结构：是把多个集中式数据库系统通过网络连接，各个节点上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。

（5）面向对象的数据库系统

面向对象数据库是数据库技术与面向对象程序设计技术相结合的产物，是面向对象方法在数据库领域中实现和应用，其采用面向对象的观点来描述现实世界实体的逻辑组织、对象之间的限制和联系，把对数据对象及对它的操作处理结合成为一个有机整体，从而提高数据库管理效率，降低用户使用的复杂性。

Visual FoxPro 提供了面向对象程序设计的强大功能和灵活性，而且仍然支持标准的过程化程序设计。

1.1.2 数据库系统

数据库系统（DataBase System，DBS）是指引入数据库并利用数据库技术进行数据管理的计算机系统。

数据库系统由 5 部分组成：数据库、数据管理系统以及建立在数据库之上的数据库应用系统、硬件系统、用户、数据库管理员。

1. 数据库相关概念

（1）数据

数据是指描述事物的符号记录。在计算机中的文字，图形，图像，声音等都是数据，另外学生档案，教师基本情况，货物运输情况等也都是数据。

（2）数据库（DataBase DB）

数据库是指数据库系统的数据源，是长期存储在计算机内的有组织、可共享的数据集合。它不仅包括描述事物的数据本身，而且包括相关事物之间的联系。实际上数据库就是为了实现一定的目的按某种规则组织起来的数据集合。

（3）数据库管理系统（DBMS）

数据库管理系统是指数据库系统的重要组成部分，位于用户和操作系统之间，统一管理和控制数据库的建立、运用和维护。常用的数据库管理系统还有 Access、SQL Server、Oracle 等。

（4）数据库系统（DBS）

数据库系统是指在计算机系统中引入数据库技术后的系统，是实现有组织、动态地存储大量相关数据，提供数据处理、信息资源共享的便利手段。

（5）数据库应用系统（DBAS）

系统开发人员利用数据库系统资源开发的面向某一类具体实际应用的软件系统。如学校教学管理系统、图书管理系统等。

（6）数据库管理员（DBA）

数据库管理员是负责建立、维护和管理数据库系统的操作人员。DBA 的职责包括定义并存储数据库的内容、建立和控制数据库的使用、负责数据库的日常维护、必要时重新组织和改进数据库。

（7）用户（User）

数据库系统的用户分为两类：一类是最终用户，主要对数据库进行联机查询或通过数据库应用系统提供的界面来使用数据库。如操作员、各类管理和工程技术人员，他们不必了解数据库系

系统的结构和模式；另一类为专业用户即程序员，他们有较强的计算机专业知识，负责设计应用系统的程序模块，对数据库进行如查询、插入、删除、修改等操作。

数据库系统层次结构图如图 1.3 所示，数据库管理系统是数据库系统的核心。

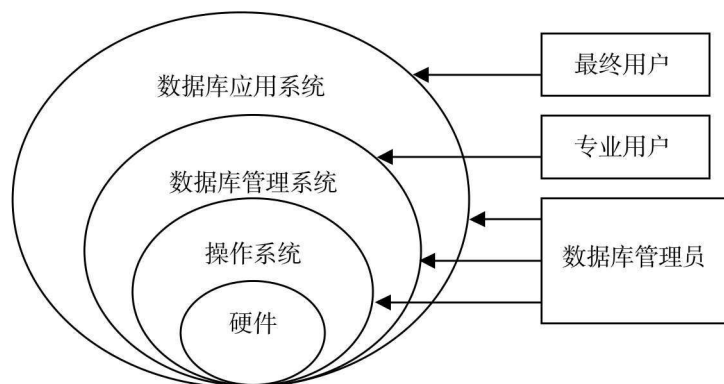


图 1.3 数据库系统层次结构图

2. 数据库系统的特点

数据库系统的主要特点如下：

（1）实现数据共享、减少数据冗余

数据库系统是从整体角度看待和描述数据，数据不再面向某个应用而是面向整个系统，即同一组基本记录，就可以被多个应用程序共享使用。这样可以大大减少数据冗余，节约存储空间，又能够避免数据之间的不相容性和不一致性。

（2）采用特定数据模型

数据库中的数据是有结构的，这种结构由数据库管理系统所支持的数据模型表现出来的。数据库系统不仅可以表示事物内部数据项之间的联系，而且可以表示事物与事物之间的联系，从而反映出现实世界事物之间的联系。因此，任何数据库管理系统都支持一种抽象的数据模型。

（3）具有较高的数据独立性

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。数据库系统提供了两方面的映像功能，使得程序与数据库中的逻辑结构和物理结构有高度的独立性。用户只以简单的逻辑结构来操作数据，无需考虑数据在存储器中的物理位置与结构。

（4）有统一的数据控制功能

为保证多个用户同时正确地使用同一个数据库，数据库系统提供数据的统一管理与控制功能，包括数据的完整性检查、安全性检查和并发控制 3 个方面。数据库管理系统能统一控制数据库的建立、运用和维护，使用户能方便地定义数据和操作数据，并能够保证数据的安全性、完整性、多用户对数据的并发使用及发生故障后的系统恢复。

安全性检查：数据库设置一套安全保护措施，保证只有合法用户才能进行指定权限的操作，防止非法使用所造成的数据泄密和破坏。

完整性检查：数据库系统提供必要措施来保证数据的正确性、有效性和相容性，当计算机系统出现故障时，提供将数据恢复到正确状态的相应机制。

并发控制：当多用户并发进程同时存取、修改数据库时，可能会发生相互干扰使数据库的完整性遭到破坏，因此，数据库系统提供了对并发操作的控制功能，对多用户的并发操作予以控制和协调，保证多个用户的操作不相互干扰。

1.1.3 数据模型

数据模型是指数据库中表示数据以及数据与数据之间的关系的工具。

数据库需要根据应用系统中数据的性质、内在联系，按照管理的要求来设计和组织。进一步说，就是由于计算机不能够直接处理现实世界中的具体事物，所以需要将客观存在的现实世界的事物抽象为一种既不依赖于具体的计算机系统又不为某一数据库管理系统支持的概念模型，然后再把概念模型转换为计算机世界某一数据库管理系统支持的数据模型。从客观现实到计算机的描述，数据的转换过程如图 1.4 所示。

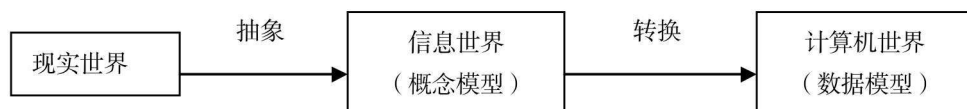


图 1.4 数据的转换过程

从现实世界抽象各种事物到信息世界时，通常采用实体来描述现实世界中具体的事物或事物之间的联系。

1. 实体描述

(1) 实体

指客观存在并且可以相互区别的事物，实体既可以是具体事物，也可以是抽象事件。例如，学生、课程、教师都是属于实际存在的事物，而学生选课和教师授课就是抽象事件。

(2) 实体的属性

描述实体的特性称为属性。例如，学生实体用（学号、姓名、性别、出生日期、籍贯、所属院系）等属性来描述。属性有属性名和属性值，属性的具体取值称为属性值，例如，学生“性别”属性取值“女”，其中“性别”为属性名，“女”为属性值。

(3) 实体集

同类型的实体的集合。例如，对于学生实体而言，全体学生就是一个实体集。

(4) 实体型

属性的集合表示一种实体的类型，称为实体型。例如，学生（学号、姓名、性别、出生日期、籍贯、所属院系）就是一个实体型。

属性值的集合表示一个实体。例如，属性值的集合（2013001，李建，男，1996-11-20，江苏）就是代表一个具体的学生。

在 Visual FoxPro 中，用“表”来存放同一实体，即实体集，例如学生表、教师表。Visual FoxPro 的一个“表”包含若干个字段，“表”中所包含的“字段”就是实体的属性集合，即实体型，字段值的集合组成表中的一条记录，代表一个具体的实体，即每一条记录代表一个实体。

2. 实体间联系模型及联系种类

(1) 联系的定义

实体之间的对应关系称为联系，它反映现实世界之间的相互联系。如读者与图书之间的关系。

(2) 实体间联系模型

实体间联系模型又称 E-R 模型或 E-R 图，它是描述概念世界、建立概念模型的工具。

E-R 图包括三个要素：

- 实体：用矩形框表示，框内标注实体名称。
- 属性：用椭圆形框表示，框内标注属性名。E-R 图中用连线将椭圆形框与矩形框（实体）连接起来。

- 实体之间的联系：用菱形框表示，框内标注联系名称。E-R 图中用连线将菱形框与有的矩形框（实体）相连，并在连线上注明实体之间的联系类型。图 1.5 所示为两个简单的 E-R 图（a、实体与属性，b、实体与实体之间联系）。

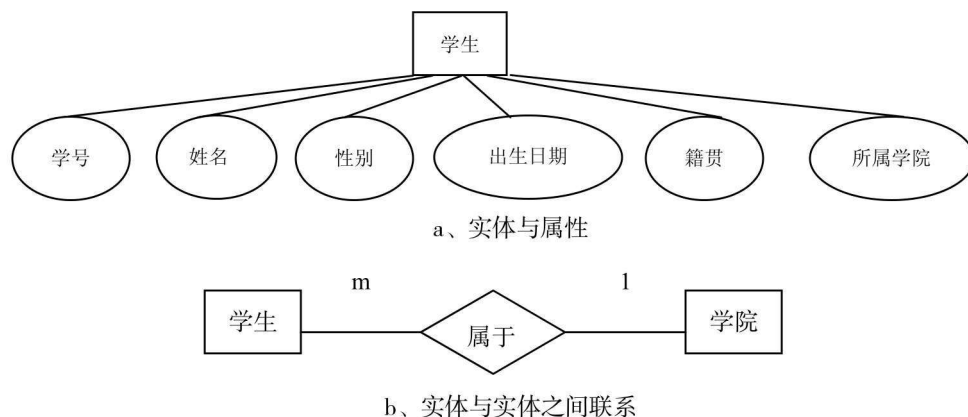


图 1.5 E-R 图

（3）联系的种类

两个实体之间的联系有以下三种类型：一对一联系、一对多联系和多对多联系。

- 一对一联系（1:1）

若两个不同型实体集中，任一方的一个实体只与另一方的一个实体相对应，称这种联系为一对一联系。如班长与班级的联系，一个班级只有一个班长，一个班长对应一个班级。还有校长与学校、公司与总经理。

- 一对多联系（1:n）

若两个不同型实体集中，一方的一个实体对应另一方若干个实体，而另一方的一个实体只对应本方一个实体，称这种联系为一对多联系。如班长与学生的联系，一个班长对应多个学生，而本班每个学生只对应一个班长。还有学院与学生、部门和职工。

- 多对多联系（m:n）

若两个不同型实体集中，两实体集中任一实体均与另一实体集中若干个实体对应，称这种联系为多对多联系。如教师与学生的联系，一位教师为多个学生授课，每个学生也有多位任课教师。还有学生与课程。

3. 数据模型

为了反映事物本身及事物之间的各种联系，需要将现实世界客观存在的具体事务进行有效的描述和抽象，转换成计算机能够处理的数据，即数据库中的数据必须有一定的结构，这种结构用数据模型来表示。数据库不仅管理数据本身，而且要使用数据模型这个抽象工具表示出数据之间的联系。可见，数据模型是指数据库管理系统用来表示实体及实体之间联系的方法，即计算机中组织数据的方法。

数据模型是数据库系统中一个关键概念，数据模型不同，相应的数据库系统就完全不同，任何一个数据库管理系统都是基于某种数据模型的。数据库管理系统常用的数据模型有下列三种：层次模型、网状模型、关系模型。

使用支持以上特定数据模型的数据库管理系统开发出来的应用系统相应地称为层次数据库管理系统、网状数据库管理系统和关系数据库管理系统。

关系模型对数据库的理论和实践产生很大的影响，成为当前最流行的数据库模型，目前，大

多数数据库都是基于关系模型建立的数据库。

（1）层次数据模型（Hierarchical Model）

用树形结构表示数据及其联系的数据模型称为层次模型。此模型中的树是由结点和连线组成，结点表示数据集，连线表示数据之间的联系。树形结构只能表示一对多联系。通常将表示“一”的数据放在上方，称为父结点；而表示“多”的数据放在下方，称为子结点。树的最高位置只有一个结点，称为根结点。根结点以外的其它结点都有一个父结点与它相连，同时可能有一个或多个子结点与它相连。没有子结点的结点称为叶结点，它处于分枝的末端。

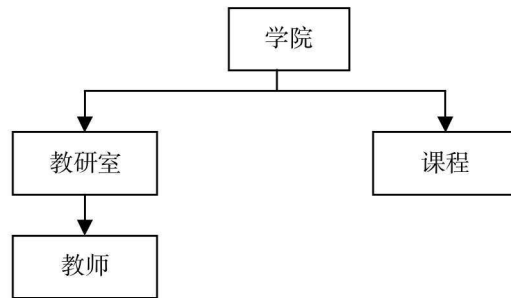


图 1.6 层次数据模型示意图

层次模型的基本特点：

- 有且仅有一个结点无父结点，称其为根结点。
- 其他结点向上有且只有一个父结点，向下可有若干个子结点。

支持层次数据模型的 DBMS 称为层次数据库管理系统，在这种系统中建立的数据库是层次数据库。层次模型可以直接方便地表示一对一联系和一对多联系，但不能用它直接表示多对多联系。图 1.6 给出了一个学院的层次数据模型。

（2）网状数据模型（Network Model）

用网络结构表示数据及其联系的数据模型称为网状模型。网状模型是层次模型的拓展，网状模型的结点间可以任意发生联系，能够表示各种复杂的联系。

网状模型的基本特点：

- 可以有一个以上结点无父结点。
- 至少有一结点有多于一个的父结点。
- 结点之间允许有两种或两种以上的联系。

网状模型和层次模型在本质上是一样的，从逻辑上看，它们都是用结点表示数据，用连线表示数据间的联系；从物理上看，层次模型和网络模型都是用指针来实现两个文件之间的联系。这种用指针将所有数据记录都存放在一起的特点，使得层次模型和网状模型存在难以实现系统的修改与扩充等缺陷。层次模型是网状模型的特殊形式，网状模型是层次模型的一般形式。

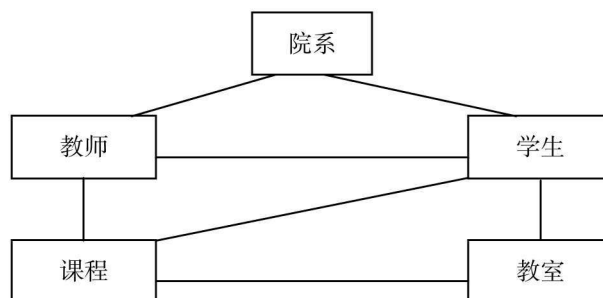


图 1.7 网状数据模型示意图

支持网状模型的 DBMS 称为网状数据库管理系统，在这种系统中建立的数据库是网状数据库。网络结构可以直接表示多对多联系，这也是网状模型的主要优点。图 1.7 给出了一个学校的网状数据模型。

(3) 关系模型 (Relational Model)

人们习惯用表格形式表示一组相关的数据，既简单又直观，如图 1.8 就是一张学生基本情况表和一张成绩表。这种由行与列构成的二维表，在数据库理论中称为关系，用关系表示的数据模型称为关系模型。在关系模型中，实体和实体间的联系都是用关系表示的，也就是说，二维表格中既存放着实体本身的数据，又存放着实体间的联系。关系不但可以表示实体间一对多的联系，通过建立关系间的关联，也可以表示多对多的联系。

关系模型是建立在关系代数基础上的，因而具有坚实的理论基础。与层次模型和网状模型相比，具有数据结构单一、理论严密、使用方便、易学易用的特点，因此，目前绝大多数数据库系统的数据模型，都是采用关系数据模型，它已成为数据库应用的主流。

Visual FoxPro 是一种典型的关系型数据库管理系统。图 1.8 给出了一个学校的关系数据模型。

学号	姓名
9607046	郑益莹
9907002	倪雯娟
9901055	赵东
9902006	和音
9704001	克敏敏
9603002	申强
9606005	迟大为
9803001	欧阳小清
9908088	毛杰
9608066	康红
9805026	夏天
9901055	李力
1087	李丽
1088	张友谊
1006	黄华
1088	张友谊

学号	课程编号	成绩
9607039	40711	88
9907002	10715	86
9901055	20111	85
9902006	10218	52
9704001	30423	85
9903001	40331	70
9606005	40625	84
9603011	20314	76
9908088	10811	89
9608066	30819	82
9805026	20534	84
9702033	30211	90
9607039	40722	82
9907002	10712	97
9901055	20115	87

图 1.8 关系数据模型示意图

1.2 关系数据库

用关系数据模型建立的数据库就是关系数据库 (Relational Database RDB)。Visual FoxPro 就是一种关系数据库管理系统。本节结合 Visual FoxPro 集中介绍关系数据库管理系统的基本概念。

1.2.1 关系数据模型

关系数据模型的用户界面非常简单，一个关系的逻辑结构就是一张二维表。这种用二维表的形式表示实体和实体间联系的数据模型称为关系数据模型。

1. 关系术语

在 Visual FoxPro 中，一个“表”就是一个关系。图 1.8 中的学生表和成绩表表示两个关系，这两个表中都有唯一标识，即学号属性，通过学号可以将两个关系联系起来。

(1) 关系

一个关系就是一张二维表，通常将一个没有重复行、重复列的二维表看成一个关系，每个关系都有一个关系名。在 Visual FoxPro 中，一个关系对应于一个表文件，扩展名为.dbf。简称为表。关系名则对应于表文件名或表名，表中的第一行构成了表的结构，其余各行构成了表的内容。

(2) 元组

二维表中,除第一行以外,水平方向的行在关系中称为元组,每一行称为一个元组。元组对应表中的一条记录。

在 Visual FoxPro 中,一个元组对应表中一条记录。例如,学生表和成绩表两个关系各包括多条记录(或多个元组),这些记录组成表的内容。

(3) 属性

二维表中垂直方向的列在关系中称为属性,每一列即每个属性都有一个属性名,属性值则是各个元组属性的取值。

在 Visual FoxPro 中,一个属性对应表中一个字段,属性名对应字段名,属性值对应于各个记录的字段值。例如,学生表中的学号、姓名、性别等字段名及其相应的数据类型组成表的结构。

(4) 域

属性的取值范围称为域,也就是不同元组对同一个属性的取值所限定的范围。域作为属性值的集合,其类型与范围具体由属性的性质及其所表示的意义确定。同一属性只能在相同域中取值。例如,性别取值范围是字符型,只能取“男”和“女”,若是逻辑型,只能取“真”和“假”。

(5) 关键字

关系中能唯一区分、确定不同元组的属性或属性组合,称为该关系的一个关键字。单个属性组成的关键字称为单关键字,多个属性组合的关键字称为组合关键字。需要强调的是,关键字的属性值不能取“空值”,所谓空值就是“不知道”或“不确定”的值,因而无法唯一地区分、确定元组。例如,学生表中的学号。

在 Visual FoxPro 中,主关键字和候选关键字就起着唯一标示一个元组的作用。

(6) 外部关键字

如果表中的一个字段不是本表的主关键字,而是另外一个表的主关键字和候选关键字,这个字段(属性)就称为外部关键字。用外部关键字可以保证有关联的表之间联系。

(7) 关系模式

对关系的描述称为关系模式,一个关系模式对应一个关系的结构。其格式为:

关系名(属性名 1, 属性名 2, ..., 属性名 n)

在 Visual FoxPro 中表示为表结构:

表名(字段名 1, 字段名 2, ..., 字段名 n)

图 1.8 中的学生表对应的关系,其关系模式可以表示为:

学生(学号, 姓名, 性别, 出生日期, 籍贯, 所属学院)。其中,“学生”为关系名,括号中各项为该关系所有的属性名。

关系是元组的集合,关系模式是命名的属性集合,元组是属性值的集合。一个具体的关系模型就是若干个有联系的关系模式的集合。

关系既可以用二维表格描述,也可以用数学形式的关系模式来描述。一个关系模式对应一个关系的数据结构,也就是表的数据结构。

在 Visual FoxPro 中,把相互之间存在联系的表放在一个数据库中统一管理。数据库文件的类型为.dbc。例如教学管理数据库中可以加入学生表、成绩表和课程表等。

2. 关系的特点

在关系模型中,关系具有以下基本特点:

(1) 关系必须规范化,属性不可再分割

规范化是指关系模型中每个关系模式都必须满足一定的要求,最基本的要求是关系必须是一张二维表,每个属性值必须是不可分割的最小数据单元,即表中不能再包含表。

(2) 在同一关系中不允许出现相同的属性名, Visual FoxPro 不允许同一个表中有相同的字段名。

(3) 在同一关系中不允许出现相同的元组, 即不允许有冗余。

(4) 在同一关系中元组及属性的顺序可以任意。

(5) 任意交换两个元组(或属性)的位置, 不会改变关系模式。

以上是关系的基本性质, 也是衡量一个二维表格是否构成关系的基本要素。在这些基本要素中, 有一点是关键, 即属性不可再分割, 也即表中不能嵌套表。

1.2.2 关系运算

在关系数据库中查询用户所需数据时, 需要对关系进行一定的关系运算。关系运算既包括传统的集合运算(并、交、差)等, 也包括专门的关系运算。专门的关系运算主要有选择、投影和联接三种。

1. 选择(Select)运算

是从关系中查找符合指定条件元组的操作。选择的条件以逻辑表达式给出, 使得逻辑表达式的值为真的元组将被选取。例如, 要从学生表中找出性别为“男”的学生, 所进行的查询操作就属于选择运算。

选择是从行的角度进行的运算, 即从水平方向抽取记录。经过选择运算得到的结果可以形成新关系, 其关系模式不变, 但其中的元组是原关系的一个子集。

2. 投影(Projection)运算

是从关系中选择若干个属性组成新的关系操作。

投影是从列的角度进行的运算, 相当于对关系进行垂直分解。经过投影运算可以得到一个新的关系, 其关系模式所包含的属性个数往往少于原关系, 或者属性的排列顺序不同。投影运算提供了垂直调整关系的手段, 体现出关系中列的次序无关紧要这一特点。例如, 从学生表中查询学生的“姓名”和“出生日期”, 所进行的查询操作就属于投影运算。

3. 联接(Join)运算

是将两个关系模式的若干属性拼接成一个新的关系模式的运算, 对应的新关系中, 包含满足联接条件的所有元组。

联接过程是通过联接条件来控制的, 联接条件中将出现两个表中的公共属性名, 或者具有相同的语义、可比的属性。联接的结果是满足条件的所有记录。

选择和投影运算的操作对象只是一个表, 相当于对一个二维表进行切割, 联接运算需要两个表作为操作对象。如果需要联接两个以上的表, 应当两两进行联接。

总之, 在对关系数据库的查询中, 利用关系的投影、选择和联接运算可以方便地分解、组合构造新的关系。

1.3 数据库设计基础

开发数据库应用系统首先要对数据库进行设计, 然后再创建数据库。只有使用一个可靠的数据库设计步骤, 就能够快捷、高效地创建一个设计完善的数据库, 才能为用户日后访问所需的信息提供方便。

1.3.1 数据库设计步骤

利用 Visual FoxPro 来开发数据库应用系统, 可以按照以下步骤来设计:

1. 需求分析

确定所建数据库所需要完成任务的目的, 用户需要明确希望从数据库中得到什么信息, 需要解决什么问题, 充分与用户交流, 并收集当前使用的各种记录数据的表格与信息。

2. 确定需要的表

把需求信息划分成各个独立的实体，每个实体设计为数据库中的一个表，每个表应当只包含一个主题的信息，且各个表不包含重复的信息。

3. 确定需要的字段

一个表包含一个主题的信息，表中的各个字段都是该主题的各个组成部分，且每个字段应满足以下要求：

- 每个字段应直接与表的主题相关。
- 字段不是推导或计算出的数据。
- 应该包含所需要的所有信息。
- 字段是不可分割的数据单位。

4. 定义主关键字

为了连接保存在不同表中的数据，即为了唯一地确定一条记录，需要为每一个表确定一个主关键字。

5. 确定需要的联系

将数据按不同主题保存在不同的表中，并确定了主关键字后，要通过外部关键字将相关的数据重新结合起来，即定义表和表之间的联系。建立了表之间的联系后，就可以充分组合实际数据库中的数据资源，大大降低数据库中的数据冗余。

6. 设计优化

初步设计后，需对设计进一步分析，检查存在的缺陷和需要改进的地方，使设计更合理，更符合用户和输出信息的需要，同时提高系统的性能，便于数据的使用和维护。

1.3.2 数据库设计原则

为了合理组织数据，数据库设计应遵循以下原则：

- 一个关系模式描述一个实体或实体间的一种联系。
- 避免在表之间出现重复字段，除了保证表中有反映与其它表之间存在联系的外部关键字之外，应尽量避免在表之间出现重复字段。
- 表中的字段必须是原始数据和基本数据元素，而不是通过推导或计算出的数据。例如，若已经存在“生日”字段，就不应该有“年龄”字段，因为年龄可以由“生日”字段推导出来。
- 用外部关键字保证有关联的表之间联系。

1.3.3 数据库设计过程

以下将遵循上两节给出的设计原则和步骤，以“教学管理”数据库的设计为例，具体介绍在 Visual FoxPro 中设计数据库的过程。“教学管理”数据库在本书的后续章节也将作为教学示例。

1. 需求分析

在“教学管理”数据库中，教学管理人员的主要工作任务包括学生信息管理、学生成绩管理、课程信息管理、教师信息管理、课程安排管理和教室信息管理等内容，如图 1.9 所示。

2. 确定需要的表

确定数据库需要的表就是将“教学管理”系统所涉及的信息以单个主题进行分离，形成一个一个个独立的实体。每个实体就是一个表，根据“教学管理”系统内容经分析可以划分为六个实体，对应的就是学生信息管理表、学生成绩管理表、课程信息管理表、教师信息管理表、课程安排管理表和教室信息管理表。