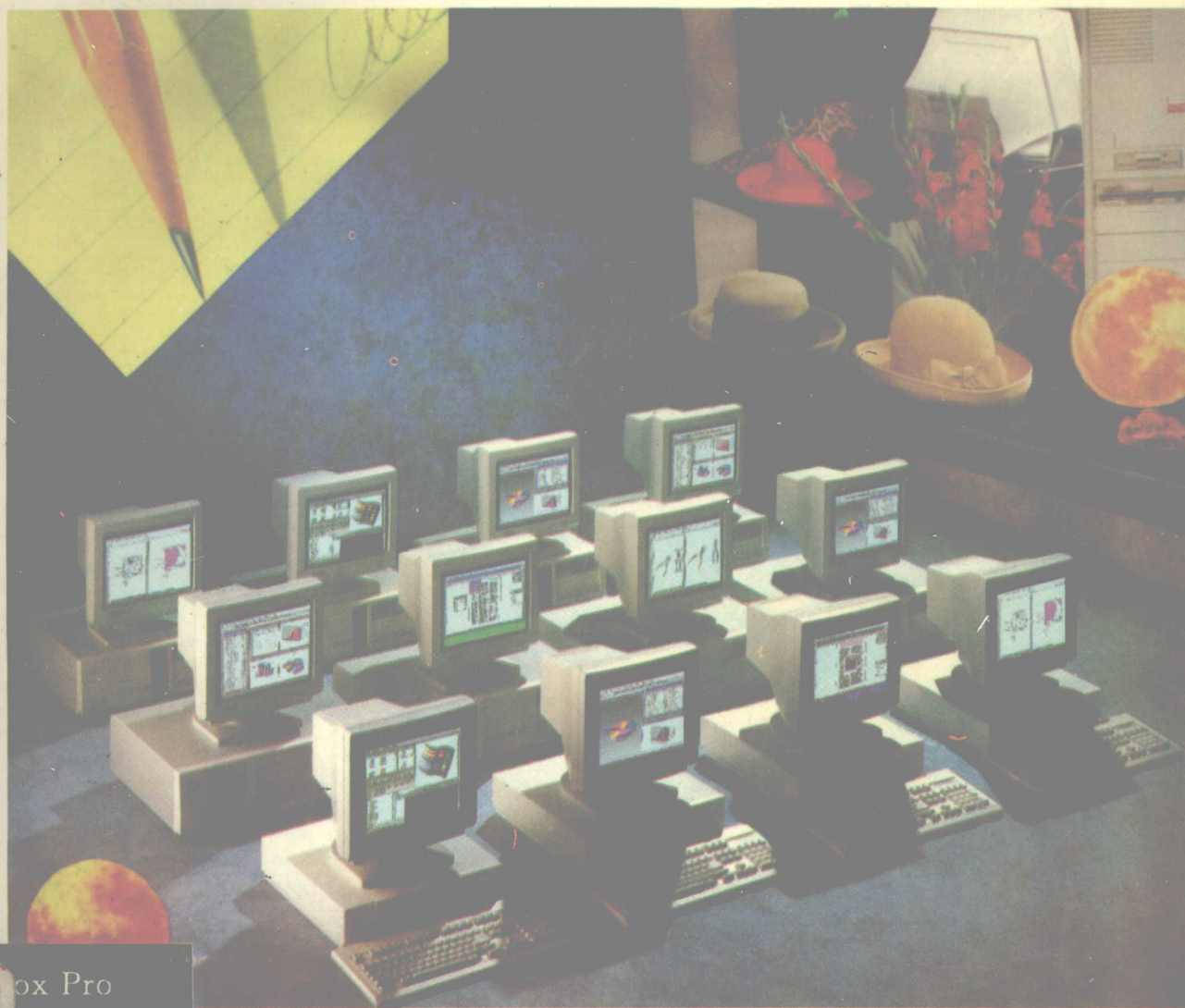


《现代电子技术》增刊

Fox Pro(2.0~2.5)培训教程

倪惠丽 编写



ox Pro

2.0~2.5)

培训教程

陕西电子杂志社

FoxPro(2.0~2.5)

培 训 教 程

倪惠丽 编写



陕西电子杂志社

目 录

第一章 绪论—FOXPRO 初步

1.1 数据概述	(1)
1.1.1 何谓数据库	(1)
1.1.2 记录和字段	(3)
1.1.3 数据库管理系统	(4)
1.1.4 关系数据库管理系统	(5)
1.2 FoxPro 的由来	(8)
1.3 FoxPro 系统的运行环境	(10)
1.4 FoxPro 的安装、启动和退出	(11)
1.4.1 安装 FoxPro	(11)
1.4.2 运行 FoxPro	(11)
1.4.3 退出 FoxPro	(11)
1.5 FoxPro 的若干术语及其作用	(12)
1.5.1 FoxPro 屏幕的术语及其作用	(12)
1.5.2 鼠标和键盘	(14)
1.5.3 FoxPro 窗口	(14)
1.6 菜单和对话框	(16)
1.6.1 菜单系统的结构	(16)
1.6.2 菜单项的选择	(17)
1.6.3 使用对话框	(18)
1.6.4 系统菜单介绍	(19)
1.7 使用命令窗口	(33)
1.8 其它功能的使用	(34)
1.8.1 ESC 键	(34)
1.8.2 “对话”	(34)
1.8.3 报警	(35)
1.8.4 系统信息	(35)
1.8.5 联机帮助	(35)
1.9 FoxPro 基础	(37)
1.9.1 数据类型	(37)
1.9.2 文件类型	(38)
1.9.3 变量和函数	(39)
1.9.4 运算符和表达式	(39)
1.9.5 数据库工作区	(40)
1.9.6 命令的作用域	(40)

目 录

1.9.7 语法描述的约定	(42)
---------------------	------

第二章 数据库的设计和建立

2.1 数据库设计	(43)
2.1.1 确定数据库需求	(43)
2.1.2 确定数据库结构	(45)
2.2 数据库建立	(46)
2.2.1 数据库结构的建立	(48)
2.2.2 保存数据库文件	(49)
2.2.3 建立数据库的实例	(56)
2.3 打开和关闭数据库	(58)
2.4 修改和复制数据库结构	(61)
2.5 显示和打印数据库结构	(64)

第三章 数据库记录操作

3.1 Browsc 方式	(64)
3.1.1 Browsc 方式及其功能	(65)
3.1.2 用 Browsc 浏览数据	(68)
3.1.3 Browsc 菜单功能	(71)
3.2 添加数据库记录	(71)
3.3 剪裁和粘贴记录	(77)
3.4 修改记录	(79)
3.5 删除和恢复数据库记录	(80)
3.6 备注型字段的数据输入和编辑	(82)

第四章 计算与查询

4.1 内存变量	(84)
4.1.1 何谓内存变量	(84)
4.1.2 建立和显示内存变量	(84)
4.1.3 保存和释放内存变量	(85)
4.2 统计数据库记录数-count	(85)
4.3 数值字段求和-SUM	(85)
4.4 计算平均数-AVERAGE	(86)
4.5 计算统计量-Calculate	(87)

第五章 索引与排序

5.1	索引类型	(88)
5.2	使用简单索引	(89)
5.2.1	用 Index On 对话框建立简单索引	(89)
5.2.2	Index On 对话框的其它功能	(91)
5.3	索引命令	(91)
5.4	建立多字段表达式索引键	(92)
5.5	选择主索引	(92)
5.6	用索引进行移动操作	(93)
5.7	保持索引文件为最新状态	(93)
5.8	用索引来搜寻数据	(93)
5.9	排序	(94)
5.9.1	按多字段进行排序	(94)
5.10	索引和排序的选择	(96)

第六章 关系型数据库和 View 窗口

6.1	理解关系型数据	(97)
6.2	使用 View 窗口	(98)
6.2.1	建立关系型数据库	(98)
6.2.2	使用关系型数据库	(102)
6.3	保存视图	(103)
6.4	“一多对应”关系	(104)
6.5	使用环境设置	(105)
6.5.1	on / off 控制板	(106)
6.5.2	FILES 控制板	(107)
6.5.3	MISC 控制板	(108)

第七章 生成报表

7.1	建立报表	(110)
7.2	报表布局窗	(111)
7.3	设计报表	(111)
7.3.1	使用 Page Preview	(112)
7.3.2	对象的操作	(113)
7.3.3	数据分组 (Data Grouping)	(114)
7.3.4	标题和总结	(115)
7.4	报表环境设置与保存	(115)
7.5	打印报表	(116)

第八章 按例关系查询

(08)	8.1 建立查询	(119)
(10)	8.1.1 RQBE 窗口	(120)
(10)	8.1.2 选择数据库	(121)
(50)	8.1.3 指定输出字段	(121)
(50)	8.1.4 查询结果排序	(121)
(00)	8.1.5 指定选择条件	(121)
(00)	8.1.6 查询条件表达式	(122)
(00)	8.1.7 分组查询	(122)
(40)	8.1.8 使用 Having	(123)
(40)	8.2 选择输出目标	(124)
(00)	8.3 SQL SELECT 命令	(124)

第九章 建立用户屏幕

(70)	9.1 使用屏幕构造器	(125)
(80)	9.1.1 Screen 菜单	(126)
(80)	9.2 操作对象	(127)
(50)	9.3 LAYOUT 对话框	(127)
(00)	9.3.1 桌面	(127)
(40)	9.3.2 窗口	(127)
(20)	9.3.3 屏幕代码	(129)
(00)	9.3.4 READ 语句	(130)
(70)	9.3.5 添加代码片段	(130)
(80)	9.3.6 环境	(130)
	9.4 框子	(131)
	9.5 字段	(131)
	9.5.1 说明显示的字段或变量	(132)
(00)	9.5.2 图形模板与函数	(132)
(11)	9.5.3 选择语句	(134)
(11)	9.5.4 FIELD 对话框	(135)
(51)	9.6 文本	(136)
(01)	9.7 控制	(136)
(41)	9.7.1 按钮	(136)
(21)	9.7.2 无线按钮	(138)
(01)	9.7.3 检测框	(138)
(01)	9.7.4 源出控制	(139)

9.7.5	滚动列表	(139)
9.7.6	不可见按钮	(141)
9.8	控制屏幕设计	(141)
9.9	快速屏幕	(142)
9.10	生成代码	(143)
9.11	屏幕构造器使用实例	(144)
9.11.1	实例要求	(144)
9.11.2	建立或修改屏幕参数	(145)
9.11.3	定义窗口、确定 READ 子项	(145)
9.11.4	建立按钮开关	(147)
9.11.5	建立文本编辑区	(147)
9.11.6	建立记录显示和记录删除状态	(148)
9.11.7	建立方框	(149)
9.11.8	生成代码	(150)

第十章 生成用户菜单

10.1	菜单生成器	(152)
10.1.1	菜单设计窗口	(152)
10.1.2	定义热键	(154)
10.2	MENU 菜单	(154)
10.2.1	General Options 对话框	(155)
10.2.2	Menu Option	(156)
10.2.3	Quick Menu 快速菜单	(156)
10.3	生成代码	(157)

第十一章 FoxPro 程序设计入门

11.1	什么是程序	(158)
11.2	建立和编辑程序	(159)
11.2.1	使用 FoxPro 文本编辑器	(159)
11.2.2	输入程序	(160)
11.2.3	设置编辑优先	(161)
11.2.4	打开多个编辑窗口	(162)
11.2.5	使用 Find 和 Replace 定位和修改文本	(162)
11.3	执行程序	(163)
11.4	设计程序	(163)
11.5	Debug 和 Trace 窗口	(165)
11.6	FoxPro 语言基础	(166)

第一章 绪论 FoxPro 初步

作为本教程的入门内容，本章将首先向初次学习数据库的读者介绍数据库的一些基本概念，然后简单叙述一下由 DBASE 到 FoxPro 的发展过程，本章的最后部分将引导读者进入 FoxPro 的运行环境，熟悉使用 FoxPro 的操作技术，为读者学习本教程的其它章节创造一个实习的“环境”。

§ 1.1 数据库概述

当今社会已是进入架设“高速信息公路”的时代，可以说是“数据库无处不在”！凡是以列表、表格和卡片形式存储信息的，按相同格式归类的文件柜都是“数据库”。尽管“数据库管理”是一个计算机术语，但却反映了老式办公文件的信息分类、存储和使用的功能。基于计算机的数据库比基于老式办公文件的“数据库”——由相同格式的索引卡片的文件柜使用起来更快捷、更灵活、使用范围更广。事实上，当纸面上的数据量大到难以维护和保证可靠性时，使用象 FoxPro 这样的数据库管理系统于你的事务管理之中，是一种最好的、明智的选择。

1.1.1 何谓数据库

数据库是由收集的任何组合在一起的作为一项整体相关的信息的集合。据此，如具有姓名、地址、城市、省名、邮政编码、电话号码和用户代码的某公司销售部门的客户档案卡文件，如具有姓名、地址、邮政编码、电话号码的一个通讯录等。虽然可能没有查阅所有列出的条目，但它们共同的一些项将适应数据库的定义。显然，这两个数据库都是由收集来的信息构成的。

为便于理解数据库概念及相关的其它概念。下面详细描述一个数据库的例子。如下图 1-1 所示：

换言之，图 1-1 的“职工登记表”是“职工登记卡”文件柜中所有卡片的汇总表。

计算机化的数据库的功能和一个典型的办公室的文件柜类似。事实上我们会发现自己一直在人工维护自己的一些数据库。例如，如果在图 1-1 中存储了职工的信息，这实际上就是在维护一个职工数据库。每次在职工数据库中添加、修改或删除信息，实际上完成了一次与计算机系统数据库操作相对应的普通的数据库操作。由于几乎是每个人都有过从文件柜中的文件中或从壁橱中的鞋盒中取东西的经历，因此现在每个人都可想象出手工维护数据库的例子。

使用存在文件柜中的信息工作时，一定是为了完成某些工作。例如：

- 为新职工在文件柜中添加新的卡片

- 修改已存在的职工的信息来记录工作部门或工资的变化
- 查找具有某些特别能力的职工

在归档时应遵守一套规则。例如，应将某个职工的信息存放在它自己的文件柜中，与其它文件分开。用户应为每个职工建立单独的“职工登记卡”，并按职工的姓氏笔划来排序，以便查找用户刚才在人工数据库上所做的工作，所遵循的规则与将要使用的计算机数据库管理系统没有什么区别。

姓 名	部门	性别	出生年月	婚否	职 务	工作日期	基本工资	简历	编号
丁虹	公关	女	05 / 16 / 72	否	助工	03 / 18 / 93	156.00		0001
王初	销售	男	02 / 10 / 64	是	工程师	02 / 10 / 87	287.00		0003
申利	开发	女	06 / 10 / 67	否	工程师	02 / 09 / 90	287.00		0009
刘祥	销售	男	04 / 15 / 73	否	助工	07 / 15 / 94	156.00		0004
沈明	开发	男	07 / 25 / 70	否	助工	04 / 18 / 92	156.00		0010
李吉平	公司	男	01 / 06 / 71	否	经理助理	03 / 16 / 91	250.00		0005
余文豪	销售	男	07 / 11 / 69	否	工程师	06 / 21 / 89	287.00		0007
张进	销售	男	03 / 18 / 64	是	经理	02 / 11 / 84	456.00		0004
黄模	开发	男	09 / 17 / 54	是	高工	03 / 27 / 76	750.00		0006
郭亮	公关	女	08 / 14 / 68	是	部长	03 / 17 / 90	256.00		0002

图 1-1

是某公司的“职工登记表”。实际上是由一叠“职工登记卡”组成，如图 1-2 所示：

职工登记表文件柜

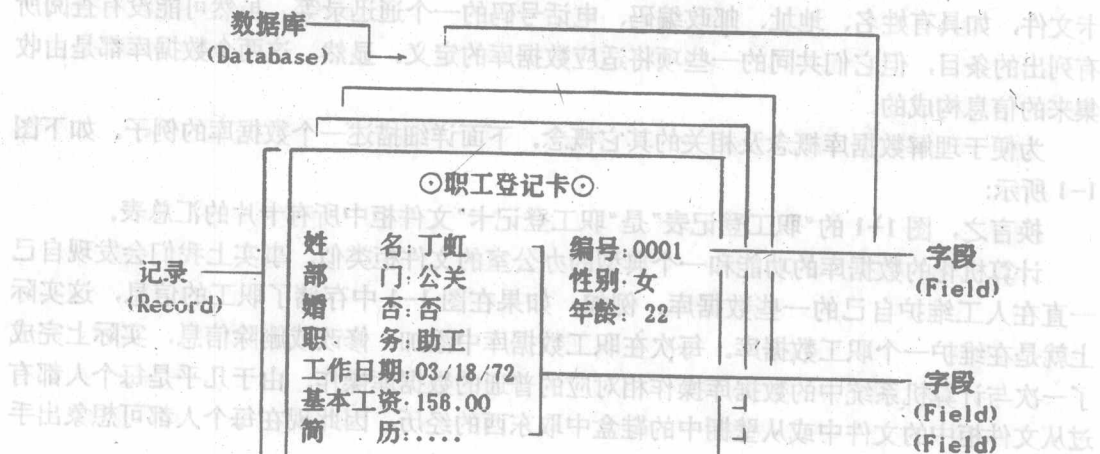


图 1-2

1.1.2 记录和字段

使数据库中的信息更直观的方法是一系列索引卡。回到前面例中的职工数据库，假设将有关职工的信息存入职工登记卡，则在数据库概念中，每个职工的索引卡片叫做记录，信息的每一记录是将关于某事的信息与存储单元联接起来的简单而又富有意义的方法，就是对索引卡所做的工作。字段是有用信息的单独的项，数据库的每个记录中都有这些项目。

浏览数据库的一个方法是象会计簿或典型的电子表格那样列出数据库的行和列。这里是列出了每个职工占用一行的情况，并且列出每一列中每一项信息。见图 1-1 所示。

读者也许会注意到上表中每一行对应一个记录，每一列对应的一个字段。在 FoxPro 术语中，行与记录是一样的，列与字段也是相同的。

FoxPro 将记录存入磁盘中一个单独的文件，叫做数据库文件，或简称为表。有时需要存入多于类的信息，则需要多个表。例如，一个职工管理系统中还可以有一个业务进修情况表等。

如图 1-3 和图 1-4 所示：

编号	DBASE	FOXBASE	FOXPRO
0001	Yes	Yes	Yes
0002	Yes	Yes	Yes
0003	Yes		Yes
0004	Yes	Yes	
0005	Yes	Yes	Yes
0006		Yes	Yes
0007			Yes
0008	Yes	Yes	
0009	Yes	Yes	Yes
0010	Yes	Yes	

图 1-3

注意：在数据库理论中，数据库是对象的容器。例如，一个职工数据库系统使用二个数据库文件。有些数据库软件将用户的数据库文件存入一个单独的磁盘文件，而由几个表构成的数据库系统更容易理解。而 FOXPRO 则将每个表存入磁盘上的一个单独的文件。用户有时会在 FOXPRO 的文档中的 database file（数据库文件）、table/DBF、data table（数据表）或 DBF 文件中看到这种方法。这是因为 FOXPRO 缺省地给这类文件加上 DBF 扩展名。

当建立数据库文件时，应给每个字段一个唯一的名字叫字段名。再看我们的职工业务进修索引卡片文件。职工登记卡的每个字段都有一个标题，例如职工姓名或职工编号。字段名就象标题，只要在选择字段名时遵循某种规则，例如，字段名不能超过十个字符。

业务进修登记卡文件柜

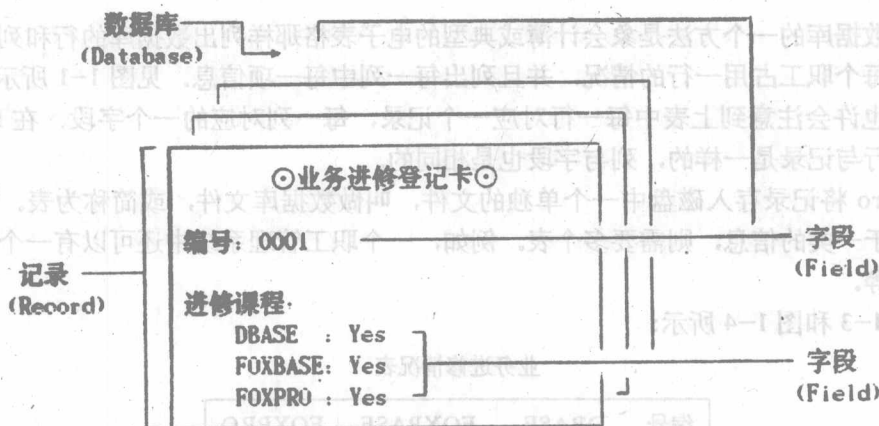


图 1-4

不妨用通俗的例子，说明字段名的目的与在存衣房中按标识检查衣服中的标识的作用相同。就象侍者按照标识很快地为客人找到大衣一样，数据库管理系统需要字段名，以便能很快地找到并处理信息，并且能保持数据的独立性（当然，FOXPRO 与侍者不同，它是需要技术的）。

1.1.3 数据库管理系统

通过上一小节（1.1.2）对数据库例子的使用操作分析，可以说对数据库的概念及其应用有了进一步认识。实际上，数据库是一个通用化的综合性的数据集合，可以提供各种用户共享，而且有最小的冗余度和较高的数据与程序的独立性，它由多种程序并发地使用，能够有效及时地处理数据，并提供安全性和完整性。数据库有数据结构化、最低冗余度、较高的程序与数据库管理系统（DBMS）和数据库管理员（DBA）组成。数据库管理系统是一个允许单个、多个用户对数据库实行建立、运用和维护的软件系统，它允许用户逻辑地、抽象地处理数据，而不必涉及这些数据在计算机中是怎样存放的。数据库管理员是负责整个系统的建立、维护、协调工作的专门人员，他们也是整个数据库系统中至关重要的一员。

当用户访问数据库时，FOXPRO 是用户和数据库之间的界面（数据其实是信息的另一种方式）。如下图 1-5 所示：

使用 FOXPRO 用户可以

- (1) 在数据库中添加新数据，如在职工数据库中添加新职工的信息
- (2) 编辑存在数据库中的数据，如在职工数据库中修改某职工的基本工资
- (3) 从数据库中删除信息，比如某职工的退休
- (4) 在数据库中查找或恢复信息，例如查找某个工资水平的所有职工
- (5) 用不同的方法组织浏览数据库，例如以工资水平或年龄排序。
- (6) 设计并打印有吸引力的报表，将数据库中的信息分组或总计

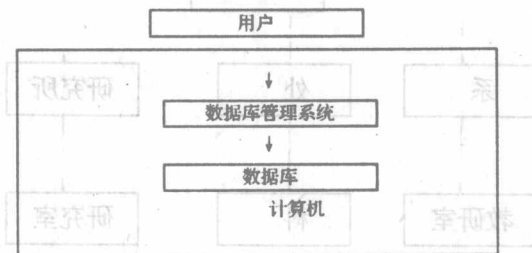


图 1-5 用户 FOXPRO 数据库三者关系示意图

这没有完全列出 FOXPRO 的功能，但是只要是能说出来的，FOXPRO 就能模仿用户对人工系统的操作。用 FOXPRO 这样的计算机数据库管理系统来代替文件夹和文件柜，是有重大意义的。

使用数据库管理系统有很多的优点。

计算机的数据库管理系统（比如 FoxPro）的最大优点是速度快。许多用户在人工数据库上做的工作，计算机都可以做得更快。例如，要查找所有使用计算机的有经验的职工（天才的、杰出的人群）。这样的查找包括取出每个职工的职工登记卡，并查阅每个职工的业务进修卡。如果职工的数目较多，这个查找将需要很长时间并要付出很大的人力。而另一方面，使用 FoxPro 只需要几分钟（甚至几秒钟）就可得到结果。

计算机的数据库管理系统的其它优点包括：

- (1) 改进了的精确度：计算上减少了错误的可能
- (2) 更高的灵活性：使用不同的方法浏览和分析数据库中的信息是很容易的。
- (3) 更及时的报告：产生定时报表是很容易的。
- (4) 更高的消费比：该 DBMS 提供了更快的信息处理速度，因而节省了用户的时间，也就等于节省了用户的金钱。
- (5) 更高的数据保密性：备份数据是很容易的。

当用户使用不切实际的人工方式分析数据时，将会学到一些有价值的东西。用户能明白其中的意义。人工分析某类问题很困难，往往用计算机分析将被证实是可行的。

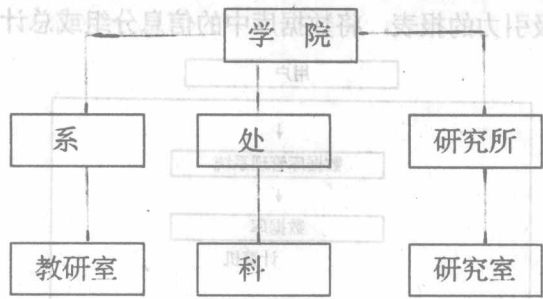
1.1.4 关系数据库管理系统

数据库一般分为层次数据库、网络数据库和关系数据库。这些分类是如何产生的呢？为了阐明这个问题，我们首先要引入实体的概念。实体（Entity）是客观存在并可相互区别的物体。实体通常指可以触及、描述的对象，如一本书、一个书架、一次借书等。数据库系统的一个核心问题就是研究如何表示和处理实体间的联系。表示实体及实体之间联系

的模型叫数据模型。数据模型常分为层次模型、网络模型和关系模型。

(1) 层次模型

用树形结构来表示实体之间联系的模型是层次模型。树的结点表示各个实体，连线表示实体之间的关系，这种关系只能是一对多的关系。在现实社会中许多关系本身就是一个自然的层次关系。一般高等院校的组织机构，其树形结构图为：



这种模型的特点是：

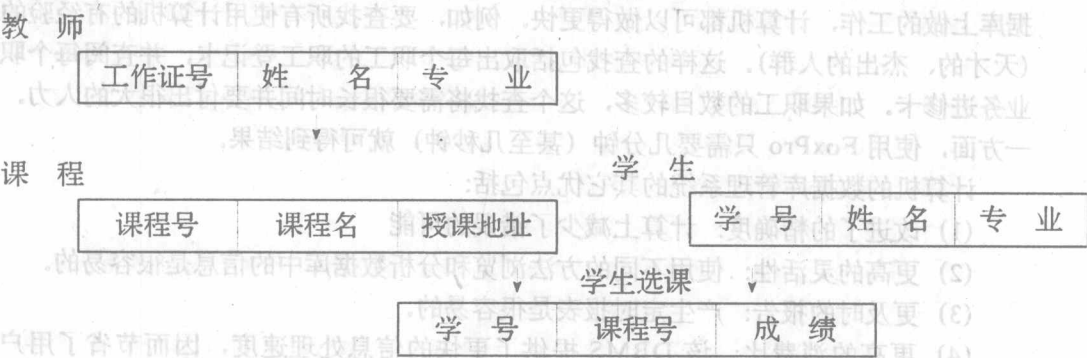
(1) 有且仅有一个结点无父结点。

(2) 其它结点有且仅有一个父结点。

可见，层次模型简单、清晰、一目了然但有一定局限。

(2) 网络模型

用网状结构来表示实体之间联系的模型称为网络模型。一般网络可反映多对多的关系。例如，教师任课和学生选课的情况，其网状结构图为：



这种模型的特点是：

(1) 可以有一个以上的结点无父结点。

(2) 至少有一个结点有多于一个的父结点。

(三) 关系模型

关系模型是用表格数据来表示实体和实体之间的联系。这种模型是最常见的一种数据模型。如 1.1.1 节中图 1-1 的“职工登记表”和 1.1.2 节中图 1-3 的“业务进修情况表”都是关系模型的例子。

关系模型是数学化的，它把数据看作二维表中的元素，而这个表就是关系。它的特点是：

(a) 表中每一列属性都是不能再分的基本字段；

(b)不允许有重复的字段名(列);

(c)不允许有相同的记录(行);

(d)行、列次序均无关。

职工登记表

姓名	部门	性别	年龄	婚否	职务	工作日期	基本工资	简历	编号
丁虹	公关	女	22	否	助工	03/18/93	156.00		0001
王初	销售	男	30	是	工程师	02/10/87	287.00		0003
申利	开发	女	27	否	工程师	02/09/90	287.00		0009
刘祥	销售	男	21	否	助工	07/15/94	156.00		0004
沈明	开发	男	24	否	助工	04/18/92	156.00		0010
李吉平	公司	男	23	否	经理助理	03/16/91	250.00		0005
余文豪	销售	男	26	否	工程师	06/21/89	287.00		0007
张进	销售	男	30	是	经理	02/11/84	456.00		0004
黄模	开发	男	40	是	高工	03/27/76	750.00		0006
郭亮	公关	女	26	是	部长	03/17/90	256.00		0002

业务进修情况表

编号	DBASE	FOXBASE	FOXPRO
0001	Yes	Yes	Yes
0002	Yes	Yes	Yes
0003	Yes		Yes
0004	Yes	Yes	
0005	Yes	Yes	Yes
0006		Yes	Yes
0007			Yes
0008	Yes	Yes	
0009	Yes	Yes	Yes
0010	Yes	Yes	

图 1-6

在以上数据模型基础上建立起来的数据库分别为层次数据库、网络数据库和关系数据库。其中关系数据库以其结构简明、理论严谨、实用性强的特点深受用户青睐，已成为目前最流行的一种数据库系统。

关系数据库管理系统经常写为 RDBMS，它比较容易把人弄糊涂，因为它有两种含义，一种是严格的含义，另一种是普遍的含义。从严格意义上讲，关系数据库管理系统支持 Edgar F.Codd 博士在《The Relational Mode for Database Management, Version 2》（《数据库管理的关系模式（第二版）》）一书（参阅麻省 Addison-Wesley 出版公司，1990）中定义的理论。而从简单的含义上讲，关系模式着重于一套复杂的规则并保持了存在数据库的信息的精确性与完整性的严格的方法。许多数据管理系统基于 Codd 博士的关系模式并支持他推荐的一些特性。

Foxpro 就是一个具有普遍意义下的关系数据库管理系统，并且它支持 Codd 博士推荐的某些特性。其特性之一，象 Foxpro 这样的关系数据库管理系统可以同时多个数据库上工作，并可以从基于一个公共字段的两个表取出信息（或使有一个公共字段的两个表相联接）。例如，假设一个职工管理系统有两个表，图 1-1 的“职工登记表”和图 1-3 的“业务进修情况表”（在真实情况下会多于两个表）。

在这里，每个职工存在“职工登记表”的一个记录中，而每个业务进修情况存在“业务进修情况表”的一个记录中。关系数据库管理系统可以用职工的“编号”这一公共段将两个表联接起来。

假如现在要查阅姓名为“申利”的职工的业务进修情况，我们可以从图 1-1 的“职工登记表”中找到“申利”，然后可以用该职工记录的公共字段编号“0002”查阅到该职工的所有进修情况了。如下图 1-6 所示：

一个不太精细的系统将使用户将所有的东西存入一个表，这意味着将职工申利的姓名、所在部门等信息输入到每个职工的“业务进修情况表”记录中。显然，这样的需求造成大量重复信息，它将占用宝贵的磁盘空间并减慢处理速度。这种方法容易造成信息的异步。例如，如果改变了，就不得不在不同的表中做多处修改。在关系模型中，则只需将职工登记表中的职工申利的记录修改一次就可以了。

关系数据库管理系统的这些特点，读者将会在本教程以后章节中学习。

§ 1.2 FOXPRO 的由来

自 20 世纪六十年代后期诞生“数据库”以来，出现了成千上万种形形色色的数据库管理系统。一直到八十年代初，随着 IBM-PC 及其兼容机的广泛使用，基本 PC 上的数据库管理系统得到迅速的发展，于 1981 年，美国的 Ashton-Tate 公司发表了由命令驱动的数据库管理系统 DBASE II，继 DBASE II 之后，经过改进，已先后推出了 DBASE III、DBASE IV 等几种版本。由于 DBASE 系列的数据库管理系统的实用性强、易学易懂易用，被广泛应用于各行各业的事务管理中，成为流行最广、应用面最大的 PC 数据库管理系统。据估计，在顶峰期时，约有 80% 至 85% 的 IBM-PC 及其兼容机的数据库应用程序都采用 DBASE 系列。DBASE 成了一种工业标准。

DBASE 与其它计算机语言相比,其最大特点是,用户可以通过 DBASE 的交互式命令建立、编辑和打印数据库报表。用户也可以使用 DBASE 所包含的程序设计语言,编制各种管理的应用程序,并可用它来为用户建立应用菜单驱动程序,使用户在操作时,只要在菜单上进行选择便能进行数据编辑或者精确地输出所需的报表。很明显,用 DBASE 完成这类编程任务远比用其它计算机语言要方便、要快。

随着 DBASE 系列各种版本的推出和广泛的流行,DBASE 所固有的弱点,也随之表现了出来。例如:

- 许多用户认为掌握各种版本的 DBASE 很困难。早期的 DBASE 版本仅为用户提供圆点提示符,用户必须记住全部进入圆点提示符之后的命令,方能管理数据库。

- DBASE 告诉用户的信息量最少,仅仅显示一个信息量最小的点“.”提示符。新的 DBASE 版本虽然增加了一个“帮助”状态,可以通过使用菜单完成一些基本的工作,但其信息量还是很有限的。

- DBASE 的计算能力弱,处理速度太慢。

- DBASE 的限制较多,如没有建立数组的能力,没有自定义函数,没有图形功能,没有多用户功能等。

- 目前 DBASE 存在的最主要问题是不带编译器。因而用 DBASE 语言编写的应用程序必须由 DBASE 来运行。即用户拥有用 DBASE 语言编写的应用程序必须在安装有 DBASE 的计算机上才能使用。这就影响了应用程序的商业性推广。

面对这些问题,一些与 Ashton-Tate 无关的公司,它们提供了一批与 DBASE 语言兼容或有更好的命令以及许多优点的产品,如 DBASE III PLOS, Clipper, FOXBASE 等等。其中最成功、性能最优良的是美国 Fox Software 公司推出的最新关系数据库管理系统 FOXBASE,它以其与 DBASE III 完全兼容,又具有功能强、速度快等特点而在 PC 关系数据库管理系统中独占鳌头。自 Fox Software 公司于 1987 年首先推出 FOXBASE 关系数据库管理系统以来,已逐步形成了三大版本: FOXBASE+1.00 系列、FOXBASE+2.00 版及 FOXBASE+2.10 版本。FOXBASE+2.00 版本比 FOXBASE+1.00 系列约晚半年,于 1987 年 7 月出版,比 FOXBASE+2.00 版晚一年,于 1988 年 7 月推出了 FOXBASE+2.10 版。每一种新版本与旧版本相比在功能及性能上都有大幅度的改进和提高,而 FOXBASE+2.10 是 FOXBASE 关系数据库管理系统中最快、最好的一个。它主要具有下列特性:

- 向用户提供了非编程的用户界面。使用户无须编写任何程序便可容易地通过下拉和弹出菜单来完成数据库的建立、打开、维护、检索等功能,还可以建立数据库的索引、过滤器以及多数据库的关联等。用户还能通过非编程用户界面直接调用各种辅助工具。

- 向用户提供了四个功能很强的辅助工具:

屏幕设计工具 (FoxVIEW) 能帮助用户方便地定义屏幕显示格式。建立美观的输入画面,并可自动生成屏幕格式文件。

应用程序生成器 (FoxCODE) 能帮助用户根据要求生成各种应用程序,如文件的维护、数据库生成及制订报表等程序。可有效地减少用户编程的工作量。

程序文档产生器 (FoxDOC) 能把用户的应用程序编成技术文档输出。能绘制出某个单独程序的流程,依据用户给出的规格产生完整的程序,并按程序产生数据字典、树状