

北京希望电脑公司微机技术丛书

HOPE

FoxPro2.0程序设计

与dBASEⅢ、FoxBase全兼容的编译型的集成环境式数据库软件



海洋出版社

北京希望电脑公司微机技术丛书

与 dBASE III、FoxBase 全兼容的编译型的集成环境式数据库软件

FoxPro 2.0 程序设计

焦小明	张华凯	编 译
李 振 格		审 校

海 洋 出 版 社

内 容 提 要

FoxPro 2.0 数据库管理系统是美国 Fox Software 公司 1991 年推出的最新软件产品。由于其采取了一系列先进技术,目前, FoxPro 已成为新型数据库管理系统的标准。

本书共分三部分,全面系统地介绍了 FoxPro2.0 的使用方法 & 高级操作技术,并结合实例详细论述了 FoxPro2.0 程序设计方法。

本书适用于具有 FoxPro 数据库管理系统中级以上水平的读者,可供广大数据库管理系统程序人员 & 大专院校师生作为自学教材或参考书。

北京希望电脑公司微机技术丛书

FoxPro 2.0 程序设计

焦 小 明 张华凯 编 译

阎 世 尊 责任编辑

海洋出版社出版(北京市复兴门外大街 1 号)

海洋出版社发行

双青印刷厂印刷

开本: 787x1092毫米 1/16

印张: 22.375 字数: 529千字

1991年5月第1版

1991年5月第1次印刷

印数: 1-3000

ISBN7-5027-2055-3/TP·28

定价: 15.00 元

编译者序

本书是根据美国 SYBEX 公司 1991 年出版的《Mastering FoxPro2》一书编译而成的。FoxPro2.0 是美国 Fox Software 公司于 1991 年中期推出的最新数据库管理系统软件。由于采用了 Rushmore 专利技术, 以及菜单及屏幕生成器等先进手段, FoxPro2.0 具有运行速度高、数据处理功能强、用户界面友好、软件开发工具丰富等优点, 因而受到广大数据库用户和程序设计人员的好评。FoxPro2.0 是一种比较理想的数据库管理系统, 它已成为新的数据库管理系统的标准。本书的主要特点如下:

- 详细介绍了 FoxPro2.0 的窗口、弹出式菜单及对话框的结构和使用技巧。
- 通过循序渐进的指导, 使用户学会使用 FoxPro2.0 对关系型数据库进行检索、查询以及输入/输出等技术。
- 从结构化程序设计入手, 通过使用菜单及屏幕生成器建立实用的范例程序, 从而理解和掌握 FoxPro2.0 程序设计技术。
- FoxPro2.0 是基于类似 Macintosh 的强大用户界面来设计的。因此, 本书对键盘及鼠标器操作均做了介绍。

通过对本书的学习, 可帮助您成为高水平的数据库管理系统程序设计员。

由于水平有限, 错误和不妥之处在所难免, 敬请读者批评指正。

编译者

目 录

前言	1
0.1 概述	1
0.2 版本范围与设备需求	1
0.3 本书的编排说明	1
0.4 本书中的约定	2
0.5 有关数据库的背景信息	2
0.6 简单数据库	2
0.7 关系型数据库	3
0.8 DBASE 与 FOXPRO	4

第一部分 使用 FoxPro

第一章 熟悉 FoxPro	7
1.1 FoxPro 菜单结构介绍	7
1.1.1 完成菜单选择	8
1.1.2 菜单系统速成	9
1.2 FoxPro 对话框介绍	14
1.2.1 对话框控制	14
1.2.2 使用对话框	16
1.3 使用窗口	17
1.3.1 鼠标器用户的专用窗口控制	18
1.3.2 使用键盘控制窗口	19
1.4 如何使用编辑器	19
1.4.1 EDIT 菜单	21
1.5 使用 COMMAND 窗口	22
1.6 其它功能	24
1.6.1 获得帮助	25
1.6.2 退出	27
第二章 建立数据库结构	28
2.1 建立一个典型的数据库文件	28
2.1.1 如何建立一个新文件	28
2.1.2 如何定义一个数据库的结构	29
2.1.3 输入字段名和类型	34
2.1.4 保存数据库文件	36
2.2 背景数据库文件	40
2.2.1 打开和关闭一个数据库文件	40

2.3	拷贝和修改一个数据库文件的结构	43
第三章	增加、编辑及查看数据	48
3.1	增加数据	50
3.1.1	用 Change 显示方式增加数据	51
3.1.2	用 Browse 显示方式增加数据	56
3.2	改变(或编辑)数据	57
3.3	浏览数据	59
3.3.1	重定字段宽度与改变字段次序	60
3.3.2	划分窗口	61
3.4	删除一个记录	62
3.5	移动指针	65
3.6	使用 COMMAND 窗口的捷径	67
第四章	索引与表达式	70
4.1	索引类型	70
4.2	使用简单索引	71
4.2.1	借助 Index On 对话框建立简单索引	71
4.2.2	INDEX ON 对话框的其它功能	74
4.2.3	索引命令	74
4.2.4	用 Structure 对话框建立简单索引	75
4.3	理解表达式	76
4.3.1	常量	77
4.3.2	函数与运算符	78
4.4	索引中使用表达式	88
4.5	使用索引	93
4.5.1	选取控制索引	93
4.5.2	使用其它类型的索引	94
4.5.3	SETUP 对话框小结	97
4.6	排序	98
第五章	查询与逻辑表达式	100
5.1	使用逻辑表达式	100
5.1.1	逻辑函数	101
5.1.2	关系运算符	101
5.1.3	逻辑运算符	103
5.2	加索引或不加索引	106
5.2.1	FOR 和 WHILE 语句	106
5.2.2	开始使用 WHILE 语句	107
5.2.3	RUSHMORE 技术	108
5.3	对单记录和多重记录进行查询	109
5.3.1	对单个记录的非索引查询	109
5.3.2	对单个记录的索引查询	112

5.3.3	多重记录查询	113
5.3.4	查询其它数据类型	117
5.3.5	处理删除记录	117
5.4	特殊技术	119
5.4.1	建立一个过滤器	119
5.4.2	在索引中建立一个查询	121
5.5	限制查询	123
5.5.1	范围	123
5.5.2	字段	125
5.5.3	最简单的查询报表: 选择项列表	126
第六章	生成报表和邮政标签	128
6.1	建立报表	128
6.1.1	报表布局窗	130
6.1.2	REPORT 菜单	132
6.1.3	一个范例报表	145
6.2	建立邮政标签	150
6.2.1	标签布局	152
6.3	后续内容简介	154

第二部分 提高能力

第七章	使用关系型数据库、View 窗口和 RQBE	157
7.1	理解关系型数据库	158
7.2	使用 VIEW 窗口	159
7.2.1	建立一个关系型数据库	159
7.2.2	使用一个关系型数据库	164
7.2.3	使用环境设置	174
7.3	范例关系查询及 SQL	178
7.3.1	RQBE 窗口	178
7.3.2	查询范例	182
7.3.3	SELECT 命令	183
第八章	菜单系统的其它功能	185
8.1	键盘宏指令	185
8.1.1	记录与使用一个新的宏指令	187
8.1.2	编辑一个宏指令	190
8.1.3	使用当前的和已保留的宏指令	191
8.2	高级编辑器技术	191
8.2.1	建立一个文本或程序文件	192
8.2.2	设置打印机与打印文件	192
8.2.3	EDIT 菜单	194
8.3	高级数据操作技术	198

8.3.1	添加	198
8.3.2	总计	200
8.3.3	利用存储器变量进行计算	201
8.3.4	替换	206
8.4	设置颜色	207
第九章	通过编程提高能力	209
9.1	结构化程序设计	209
9.2	预备知识	211
9.3	与用户对话：输入/输出	212
9.3.1	非格式化输入/输出	213
9.3.2	格式化输入/输出	218
9.4	控制流	220
9.4.1	循环控制	220
9.4.2	选择控制	224
9.4.3	EXIT 与 LOOP	228
9.5	过程与参数	230
9.5.1	变量范围与参数传递	231
9.5.2	用户定义的函数	234

第三部分 FoxPro 程序设计

第十章	理解屏幕生成器	235
10.1	建立屏幕与生成代码	235
10.2	操作目标	236
10.3	LAYOUT 对话框	237
10.3.1	桌面	237
10.3.2	窗口	238
10.3.3	屏幕代码	240
10.3.4	READ 语句	240
10.3.5	添加代码片段	241
10.3.6	环境	242
10.4	框子	242
10.5	字段	243
10.5.1	说明显示的字段或变量	244
10.5.2	图形模板与函数	244
10.5.3	选择语句	246
10.5.4	FIELD 对话框的其它功能	248
10.6	文本	249
10.7	控制	249
10.7.1	按钮	249
10.7.2	无线按钮	251

10.7.3	检测框	251
10.7.4	弹出控制	253
10.7.5	滚动列表	254
10.7.6	不可见按钮	255
10.8	控制屏幕设计	255
10.9	快速屏幕	256
10.10	生成代码	257
第十一章	编写应用程序	260
11.1	分析	260
11.1.1	画结构图	261
11.2	主菜单	263
11.3	分枝测试	267
11.4	报表与邮政标签	269
11.4.1	REPORT 菜单	270
11.4.2	LABEL 菜单	275
11.5	输出合并邮件	279
11.6	DATA 子菜单	280
11.6.1	测试所用的分枝程序	282
11.6.2	EDIT 屏幕程序	283
11.6.3	LOOKUP 模块	288
11.6.4	APPEND 屏幕程序	293
11.6.5	DATA 菜单的其它功能	296
第十二章	使用菜单生成器	298
12.1	菜单设计窗口	298
12.1.1	热键	301
12.1.2	OPTION 检测框	301
12.2	MENU 菜单	302
12.2.1	通用选择项	302
12.2.2	菜单选择项	304
12.2.3	快速菜单	305
12.3	生成代码	306
12.4	菜单命令	306
12.4.1	定义一个菜单	307
12.4.2	利用菜单生成器建立非系统菜单	308
12.5	菜单系统范例	309
附录 A	在 PC 机上安装 FoxPro	315
A.1	设置路径	317
附录 B	FoxPro 实用程序	319
B.1	文件管理程序	319
B.1.1	Files 控制板	319

B.1.2	TREE 控制板	328
B.2	桌面实用程序	334
B.2.1	计算器	334
B.2.2	日历/日记	336
B.2.3	特殊字符	337
B.2.4	ASCII 码表	338
B.2.5	捕捉实用程序	339
附录 C	建立应用程序和 EXE 文件	340
C.1	项目管理程序	340
C.1.1	使用 Project 窗口	340
C.2	项目菜单	345

前 言

0.1 概述

无论您是一位使用计算机数据库程序的初学者，还是数据库管理系统的长期用户，本书均将使您对最新版本的 **FoxPro** 数据库管理系统有一个全面的了解。从最基本的内容开始——使用键盘和鼠标器在显示菜单上或框子中做移动——通过循序渐进的指导，本书将使您取得迅速的进步，并能利用本程序的主要特性建立和使用一个实际数据库。本书中全部过程的介绍均是从有经验的用户或程序员的角度来描述，所以，只要您了解基础知识，就能理解数据库技术所隐含的基本原理。在掌握了数据库的基本使用方法后，您可以继续学习，并成为高级用户；当然，您也可以着手编写自己的程序。结构化程序设计方法，以及快速交互的编程工具，使最新版本的 **FoxPro** 功能十分强大。

0.2 版本范围与设备需求

本书介绍了 **FoxPro2.0** 版本。**FoxPro2.0** 于 1991 年中期推出，它需要一台 IBM 或兼容的 PC 机，带 512K 内存(推荐使用 640K)。为了运行带有帮助文件和实用程序的完整程序，您至少应具备 4M 硬盘空间。由于 **FoxPro** 在运行中将建立临时文件，因此，最好留有比最小要求值更大的空间。

彩色监视器的效果当然很好，您可以选用各种彩色图表以使显示更有利于视觉。但是，与专为彩色监视器上运行而设计的软件包不同，**FoxPro** 不但可以使用彩色监视器，而且在单色监视器上的显示不存在任何问题。此外，鼠标器有助于您的工作，但也不是必需的。本书既介绍了鼠标器指令，也介绍了键盘指令。

本书未提供有关安装方面的具体指导和如何在网络上使用 **FoxPro**。

0.3 本书的编排说明

在各个章节中，您将遇到循序渐进的过程。由于有些过程建立在以往过程所完成工作的基础上，所以您最好从第一部分入手，逐渐深入，并通过具体例子来实习。在读完第一部分后，您可以继续学习第二部分，掌握更高级的编程技巧；当然，您也可以跳至第三部分，学习如何使用 **FoxPro** 进行程序设计。

本书的第一部分(第一至第六章)介绍了 **FoxPro** 的特性，这是一般用户所必须掌握的。程序中不必了解的特性放在后续部分或附录中，不会妨碍您尽快学会使用本程序。例如：有些菜单系统的特性是不必需的，因此放在第二部分中，这将有助于您尽快了解 **FoxPro**。

在学习了本书第一部分后，您便可以着手使用 **FoxPro**。您将能利用所学的知识应付各

种简单数据库应用程序，这几乎包括了所有实际工作中可能遇到的问题。

本书还将帮助您成为 FoxPro 的高级用户，书中第二部分向您提供了菜单系统的全部功能。例如：指导您如何建立键盘宏指令，即对任何功能键或者几乎所有由 Ctrl、Alt、Shift 与其它键的组合键赋值。该部分还将指导您如何建立和使用关系型数据库。

本书的第三部分以 FoxPro 为具体实例，介绍了计算机编程的基本原理(诸如控制流和传递参数)。在最后几章，您将编写一个完整的邮政列表程序，其中包括窗口、鼠标器支持，以及弹出式菜单。FoxPro 的功能非常强大，无论您以往是否编写过程序，在读完本书之后，您一定能够编写出高质量的程序。实际上，本书虽未涉及 FoxPro 的全部先进编程特性，但却是一本适合于中等水平读者的详细而有价值的指导书，它将指导您使用最主要的工具：屏幕生成器和菜单生成器。

您也可以跳过第二部分，在完成第一部分之后直接进入第三部分。您可以将第一部分的所有程序作为练习。在学过编程(第三部分)后，再返回第二部分，学习其它 FoxPro 命令，提高编程能力。

FoxPro 语言丰富，任何人要全部掌握它都很困难。它包括了所有 dBASE 语言的内容和其它许多内容。即使是经验丰富的程序员，在浏览参考手册后也会惊喜地发现许多新的命令或函数，它们比以往的命令和函数更加方便。与此同时，使用 FoxPro 接口及基本命令也非常简单。

0.4 本书中的约定

命令用尖括号(小于和大于号)括起来的字及括号本身在实际使用命令时应由实际值替代。例如：如果一条命令包括 <file name>，则必须输入实际的文件名。

0.5 有关数据库的背景信息

六十年代，计算机科学家开始研究如何将以前用书面形式保存的文件设计成新的数据形式，以便用计算机更有效地管理文件。就一般而言，他们可以用新的术语精确地描述出以往人们用书面文件表达的内容。他们所开发的某些全新数据处理方法在没有计算机的时代根本无法实现。

0.6 简单数据库

通常，书面编排的数据可以用同样编排方法由一个简单的数据库更完美地表示出来。

典型的老式办公文件由一些相同的表格构成，在每个栏目中填入相应数据。例如：将一份名单写在一组索引卡片上，每张卡片都标有在何处填写姓名、地址、城市、州、邮政编码以及电话号码。虽然，大多数数据要写在更大的表格上，放进文件夹，但基本原则是一样的：首先打印一份标准表格，指示数据类型，然后在每张表的各个栏目中填入实际数据。

简单的数据库正是按照这一方法编排。您所要了解的是引入计算机之后，如何用新的

术语来描述它们：

- 字段：每个需要填充的栏目称为一个字段。在上面叙述的索引卡片数据库中，姓名为一个字段，地址为一个字段，等等。
- 记录：在一张表格上出现的所有数据称作一个记录。在我们的例子中，一个人的姓名、地址和电话号码构成一个记录。
- 文件：一组相似的记录放在一起称为一个文件。在例子中，装有全部姓名和地址的索引卡片盒构成一个文件。当然，只要您还有另外的盒子，您还可以拥有其它的文件，其中保存其它的名单和地址。

无论何时，只要人们谈到数据库，就会反复提到上述三个术语。图 I-1 表示了如何使用数据库术语描述老式的索引卡片文件。

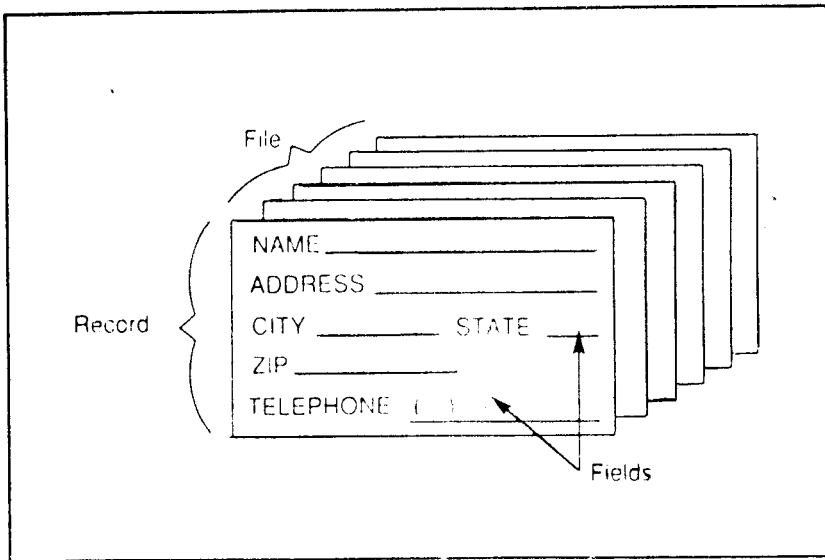


图 I-1 数据库术语

将这种数据编排成一张便笺是很方便的。当然，计算机可以用两种方式显示数据，表格形式可以显示每个人的表格，便笺形式则是一列姓名、一列地址、等等，这样每个人的数据可以列在一行上。当数据按便笺方式编排时，每一列代表一个字段，每一行代表一个记录，便笺则代表整个文件。

0.7 关系型数据库

计算机化数据的某些优点是明显的。计算机可以连续查找您所需的记录，例如：查找某人的电话号码。它可以打印不同表格中的数据，避免从不同表格中拷贝数据的麻烦。例如：当您想给人打电话时，便可以打印一张名单和电话号码，当您想发信时，便可以打印一张标签。它还可以从文件中选择某些记录，例如：仅列出住在纽约的人的名字和电话号码，或者只打印住在新泽西的人的邮政标签。它可以使您按不同顺序使用数据。例如：您可以毫不费力地按邮政编码顺序打印一份邮政标签，按字母顺序打印一份名单和电话号

码。

除了计算机化数据的明显优点之外，计算机科学家还发现了计算机的潜在优势。他们发现计算机可以有效地避免重复。

靠纸张办公的年代，一家大型公司具有许许多多的表格，其中包括每一位雇员的姓名和住址。工资部门使用的表格包含每一位雇员的姓名、住址和工资额；福利部门使用的表格包括每位雇员的姓名、住址和所享受的福利；人才开发部门使用的表格包括每位雇员的姓名、住址以及所参加的培训课程，等等。

很明显，这种重复是低效率的。同样的姓名、住址以及其它诸如社会保险号码、电话号码等基本信息必须一遍又一遍地填写。如果一位雇员搬了家，那么各个部门都需要更改许多表格上的住址。

在老式的办公室里，重复是难以避免的，但很快计算机科学家便找到了一条途径，使得姓名和住址一经录入，各个部门便都可以使用这些姓名、住址以及工资、福利、培训课程等特殊数据。

以前，将所有数据结合在一起的方法十分复杂而且不易使用。虽有按层次编排的数据库和用指针编排成网络的数据库，但这两种数据库都很复杂，只有经过数据库理论培训的程序员才能理解和使用它们，公司必须有一个程序员小组专门从事信息收集和产生报告的工作。

1970年 E.F.Codd 发明了一种新的方法用以将数据结合在一起，这就是所谓的关系型数据库。其思想是将数据划分为独立的文件，而这些文件可借用公共的关键字段联系起来。

在范例中，公司以其雇员的编号为一个关键字段。每位雇员的雇员编号、姓名、地址以及其它基本数据构成一个文件，雇员编号和工资构成另一个文件，雇员编号与福利待遇又构成一个文件等等。诸如姓名和住址等基本信息可能仅出现一次。各个部门都有一个程序用来将雇员编号与所需的每个人的工资或福利等基本数据联系起来。

借助于关系型数据库联系数据的主要工作由数据库管理程序来完成。不需程序员建立指针来联系一个记录与另一个记录。只需输入一个文件中正确的雇员编号和工资，数据库管理系统便可以把这个记录与另一个文件中姓名和住址的记录联系起来。

Codd 用数学集合理论分析了关系型数据库，并发明了使数据重复率最低的数据划分方法。由于其既有效又直观易用，因此关系型数据库很快便成为处理复杂数据的极好方法。

在简单数据库中，例如：只由姓名和住址构成的数据库，数据库和文件没有区别；一个简单的数据库只涉及一个文件。而关系型数据库所涉及的文件不只一个。在已定义的术语(字段、记录和文件)之外还需增加一个术语——数据库(database)。一个数据库是由一个应用程序中所用的全部数据构成，无论其是在一个文件中还是在多个文件中。

0.8 DBASE 与 FOXPRO

八十年代，随着 IBM-PC 及其兼容机的广泛使用，dBASE II 程序迅速成为主导的数据库管理程序。简单程序之所以普及，一般是由于它们易于使用，但它们仅限于管理简单的数据库，例如：由姓名和住址构成的简单表格。而 dBASE II 则是既能处理简单数据库又可

以处理关系型数据库的最流行的程序。

dBASE 的交互式命令使用户可以建立、编辑和打印数据库报表。dBASE 还包括一种易学易用的程序设计语言。程序员可以用它为用户建立应用菜单驱动程序。不想为学习 dBASE 而烦恼的用户可以请程序员为他们建立应用程序，此后，只需在菜单上进行选择便可进行数据编辑或者精确地输出所需的报表；用 dBASE 完成此类编程比用其它计算机语言更快、更简单，因为 dBASE 是为完成这类任务而专门设计的。

由于功能较强，dBASE 是目前微机上最流行的数据库管理系统，在其鼎盛时期，约有 80~85% 的 IBM-PC 兼容微机数据库应用程序使用 dBASE。dBASE 已成为一种工业标准。即使是对那些与数据库无关的程序，人们也希望使用 dBASE 数据文件。

继 dBASE II 之后的 dBASE III 和 dBASE III + 功能得到增强，但它们仍存在局限性。

但许多用户认为掌握各种版本的 dBASE 非常困难。dBASE 的早期版本仅为用户提供了点提示符，用户必须记住命令才能对数据库进行操作。必须输入正确的命令才能编辑数据或生成报表。即便是最简单的数据库，也必须掌握许多命令；如果命令在拼写或在语法上有细微差错，dBASE 将显示错误信息并拒绝工作。

严格地说，dBASE 比其它程序向用户提供的信息要少：就连复杂的 DOS 都能显示当前驱动器提示符，而 dBASE 则仅仅显示一个信息量最小的点提示符。新的 dBASE 版本增加了一个“辅助” (Assistant) 模式，可以使用菜单完成一些基本的工作，但其能力是有限的。

用户们提出了各种各样的问题。有的抱怨 dBASE 语言的某些限制，比如没有建立数组的能力，但目前最普遍的问题是 dBASE 不带编译器。

大多数程序设计语言是编译型的，应用程序按照类似英语的语言书写，然后编译器将其转换成计算机可用的机器语言。程序员向用户提供的应用程序通常是编译过的程序，它可以直接在计算机上运行。

有一些较原始的程序设计语言(如老版本的 BASIC)是解释型的。程序也是由类似英语的语言书写。运行中，解释器在某一时刻读入一条语句并解释成机器码。与编译型程序相比，解释型程序有两个主要的缺点。首先，它运行很慢：每条语句在运行前都要经过解释，这需占用时间。第二，应用程序必须与解释程序一起提交给用户，否则程序就无法运行。

dBASE 程序员也抱怨这些问题。人们认为在大型数据文件上运行程序速度太慢。有些人认为用 dBASE 编写的程序必须由 dBASE 来运行是一件很糟糕的事情。没有 dBASE 的用户要雇用一位程序员为其编写数据库管理系统，他们当然不希望再花同样多的钱去拷贝一份 dBASE。换句话说，虽然一些程序员使用 dBASE 编写了商品化的程序(诸如小型公司记帐程序)，但是由于用户为了运行这些程序必须具有 dBASE，因此，这些程序不可能真正得到大量用户。

面对这些问题，一些与 Aston-Tate 无关的公司提出了编译型和伪编译型的 dBASE III 和 dBASE III + 语言——从总体上讲，它们的功能得到了加强，但有时与原语言有些不兼容。

从事该项工作的公司之一就是 Fox Software 公司，它开发了 FoxBase、FoxBase+ 和 dBASE 编译开发系统，其包括一个伪编译器。伪编译器不能把程序转换成计算机可直接运行的机器码，而是做部分转换并建立比原程序更接近机器码的代码，其运行速度比用解释程序要快，但它仍然不能运行，必须同时提交一个同步运行模型以便在运行中随时进行解

释。

FoxBASE 曾被普遍认为是最好的 dBASE 编译器,因为其程序运行速度快而且与 dBASE 高度兼容。用 dBASE 编写的程序可以用 FoxBase 编译,而其它编译器则必须在编译之前对程序进行一些修改。

各种 dBASE 版本不断出现。编译器使原始 dBASE 语言功能增强,同时,也带来一些差别。随着更多增强型版本的出现,dBASE 的标准变得越来越模糊了。Aston-Tate 公司已不再能定义 dBASE 的标准,就连 dBASE IV 本身也未按标准来设计。目前,程序员们一般用 xBase 来表示各种数据库管理程序设计语言。

当功能明显优于 FoxBase+ 的 FoxPro 首次推出时,迅速受到了众多数据库用户和程序员的好评。在目前的 FoxPro 第二版中包括了 Fox 公司的专利 Rushmore 技术,对其测试表明:与其它产品相比,它能以成百倍的速度进行查询。无疑这将对用户运行巨型数据库有所帮助。目前, FoxPro 已经成为新的数据库管理系统的标准。

FoxPro 是围绕着类似 Macintosh 的强大用户界面来设计的,它是基于窗口和弹出式菜单(既可以用键盘又可以用鼠标器操作)以及特有的命令窗口而设计的,您可以在窗口输入在 dBASE 点提示符之后输入的过程命令。作为最显著的特点,该界面能够使从菜单系统中选择的命令在命令窗口中自动生成等效命令。这使用户易于熟悉程序,并且逐渐学到更多的知识。

对于程序员来说,借助 FoxPro 的菜单和屏幕生成器,通过无线按钮、按钮和其它先进的接口特性可以方便地建立弹出式菜单和屏幕。即使是一位中等水平的程序员,也可以借助鼠标器建立令人满意的用户界面。高级程序员则希望利用 FoxPro 的底层文件处理功能(它类似于 C 语言的文件处理功能),并使用它的应用程序生成器建立不受操作系统控制的应用程序和 EXE 文件。

第一部分 使用 FoxPro

第一章 熟悉 FoxPro

使用 **FOXPRO** 时，您所需做的一切工作均是围绕着弹出式菜单与窗口进行的。初学者利用该接口可对其数据进行简单的菜单驱动操作。高级程序员可利用同一接口编写和编译其程序文件。

本章指导您如何使用 **FOXPRO** 接口，并在您继续阅读本书其它部分之前，熟悉 **FoxPro** 的基本概念。

如您所见，**FoxPro** 既可使用键盘又可使用鼠标器。当使用键盘时，您可以多种途径完成同一工作。这使您能自由选择您认为最方便的方法。

由于大多数操作可以用多种办法实现，因此在每次的讨论中不可能将其一一列举。以后章节中还将提供您利用 **FoxPro** 的详细说明书，使您能从菜单和对话框中做出选择，说明书将不会详述进行选择的全部方法，因为这将占用过长的时间和过大的篇幅。

因此，充分利用本章非常重要。本章将详述完成 **FoxPro** 操作的全部方法，而这些内容在您使用本书时将经常使用。在您阅读时体会一下各种方法。在您结束本章时，如果您能熟悉一种选择菜单的方法、一种使用对话框的方法、及一种使用窗口的方法，将有助于您使用本书的其它章节。

1.1 FoxPro 菜单结构介绍

FoxPro 带有便于使用的接口，使您能借助命令窗口从弹出式菜单中选择项目，您可将 **dBASE** 早期版本的命令键入窗口。在从菜单中进行选择方面，**FoxPro** 的功能是独特的，程序显示出命令窗的命令等价式。

本节将指导您从 **FoxPro** 菜单系统中进行选择，并略述 **FoxPro** 的菜单结构。当然，首先您需启动程序。若您若未安装 **FoxPro**，则可根据附录 A 的安装说明进行安装，然后继续阅读。

① 确认您已进入 **FOXPRO2** 目录，该目录是在安装程序时建立的。

② 在 **DOS** 下，键入 **FOXPRO**，按 **Enter**。您将看到初始化 **FoxPro** 屏幕，如图 1.1 所示。

主菜单在屏幕顶部。大部分版面被 **FoxPro** 字母及其下的版权提示所占据。命令窗靠近屏幕底部。注意：光标在命令窗，您可键入过程命令。不幸的是本幅屏幕图中看不到光标。如果您想了解怎样找到光标，文本将给您具体指导。

屏幕顶端的高亮度菜单被称为“菜单区”。写在菜单区的每个字被称为“菜单项”，