

FOXPRO™ 2.5

DEVELOPER'S HANDBOOK

开发人员手册

Foreword by Dr. David Fulton, Architect, Fox Management, Microsoft Corporation

2ND EDITION

FEATURING
FOXPRO™ 2.5 for
DOS and
FOXPRO™ 2.5 for
WINDOWS™

CONTAINS TIPS FOR
CROSS-PLATFORM
DEVELOPMENT


RANDOM HOUSE
ELECTRONIC PUBLISHING

希望

DAVID M. KALMAN

海洋出版社

FoxPro 2.5 Developer's Handbook

FoxPro 2.5 开发人员手册

David M. Kalman 著

彭 彭 木 杉等 译

东 岳 审校

海 洋 出 版 社

1995 年 · 北京

(京)新登字 087 号

内 容 提 要

本书是一本开发人员手册,对运用 FoxPro 2.5 进行数据库管理系统开发作了全面系统的介绍。全书共十章和一个附录,内容有备注与字符串处理、数据库/索引文件的定义与设计、BROWSE 技术、报表生成器、Rushmore 技术与 SQL、FoxPro API、屏幕与菜单管理、DDE 与 OLE。

本书内容丰富,集技巧性与实用性于一体,是初学者的良师,有经验编程人员的益友。本书可作为大中专院校学生、科研人员学习使用 FoxPro 的教材和参考用书。

需要本书的读者可直接与北京海淀 8721 信箱书刊部联系。邮政编码:100080,电话:2562329。

版 权 声 明

Copyright © 1993 David M. Kalman

This translation published by arrangement with Random House, Inc.

本书英文版名为《FoxPro 2.5 Developer's Handbook》,由 Random House 出版公司出版。版权归 Random House 所有。本书中文版由 Random House 授权出版。未经出版者书面许可,本书的任何部分均不得以任何形式或任何手段复制或传播。

FoxPro 2.5 开发人员手册

David M. Kalman 著

彭 彭 木 杉 等译

东 岳 审校

责任编辑:钱晓彬

*

海洋出版社出版发行(北京市复兴门外大街1号)

北京市地矿局印刷厂印刷

开本:787×1092 1/16 印张:36.625 字数:855 千字

1995 年 1 月 第一版 1995 年 1 月第一次印刷

印数:1—5000

*

ISBN 7-5027-4102-X/TP·254 定价:62.00 元

前 言

当前数据库技术已在日常生活中获得了广泛的应用,它是计算机科学与工程领域的重要研究内容,是计算机科学与工程技术人员应掌握的必不可少的知识与技能。

FoxPro 作为新一代数据库管理系统的杰出代表,是美国 Fox 软件公司于 1989 年推出的,最初版本为 1.0,其后版本不断更新,功能不断增加。自 Fox 软件公司并入 Microsoft 公司后,于 1992 年 3 月推出了全新的适合 DOS 和 Windows 两个平台的 FoxPro 2.5 版。FoxPro 作为全新的关系型数据库管理系统,具有强大的性能,无与伦比的速度,完整而丰富的工具,极其友好的图形用户界面,简单的数据存取方式,完整的 XBASE 语言,良好的兼容性,独一无二的跨平台特性及真正的可编译性,使系统成为目前最快、最完美的数据库系统,并逐渐成为新的 DBMS 工业标准。如同 dBASE 逐渐被 FoxBASE 所取代那样,FoxPro 将成为用户微机数据库管理系统的首选对象。

本书是一本开发人员手册,对运用 FoxPro 2.5 进行数据库管理系统开发作了全面系统的介绍。

本书由彭彭和木杉翻译,东岳负责审校,参与本书翻译工作的还有张东灵、王彬、章强、傅国勇、刘小东、林得利、张小莉、刘伟和章林。

译 者

1995 年 1 月

目 录

第一章 概 述	(1)
1.1 什么是 FoxPro 2.5	(1)
1.2 历史展望	(2)
1.3 本书包括哪些内容	(3)
1.4 本书对你合适吗	(4)
1.5 了解 FoxPro	(5)
1.6 数据驱动应用程序生成和工具	(6)
1.7 分布应用程序和 .EXE 程序	(15)
1.8 Rushmore 技术	(17)
1.9 SQL 与 RQBE	(18)
1.10 FoxPro 2.5 索引格式	(19)
1.11 模板语言命令	(19)
1.12 FoxPro 2.5 的主要优点与新的特征	(22)
1.13 FoxPro for Windows 中的字体和字型	(27)
1.14 FoxPro for Windows 中的屏幕定位	(32)
1.15 新颜色选项	(32)
1.16 新系统内存变量	(34)
1.17 FoxPro for Windows 中的动态数据交换	(35)
1.18 对象连接与嵌入 (OLE)	(35)
1.19 刚刚起步	(36)
第二章 掌握备注和字符串	(37)
2.1 容量	(37)
2.2 字符串处理性能	(37)
2.3 字符串处理的特征	(38)
2.4 EDIT 窗口颜色	(45)
2.5 激活和关闭一个文本编辑区	(48)
2.6 对备注字段进行索引	(62)
2.7 不用退出保存备注	(64)
2.8 移动备注	(64)
2.9 备注字段的压缩	(66)
2.10 给文本中的表达式赋值	(68)
2.11 在 MS-DOS 应用程序中显示图形	(73)
第三章 数据库/索引文件的定义和设计	(76)
3.1 FoxPro 索引文件格式	(76)
3.2 FoxPro 2.x 新的索引特征	(77)

3.3	同时操作复合索引和压缩索引	(79)
3.4	索引的一般策略	(80)
3.5	典型的主关键字	(82)
3.6	FoxPro 2.5 的数据类型	(85)
3.7	备注文件 (.FPT) 基础	(92)
3.8	数据库的设计: 理论和应用	(98)
3.9	对典型非标准化的思考	(101)
第四章	掌握 BROWSE	(104)
4.1	BROWSE 的演变	(104)
4.2	定义 BROWSE 颜色	(106)
4.3	选择 BROWSE 的字段	(110)
4.4	控制单个字段的显示形式	(115)
4.5	用模式和样板把字段格式化	(115)
4.6	定义一个定制列标题	(117)
4.7	字段级事件触发器和有条件输入	(118)
4.8	BROWSE...FORMAT; BROWSE...FIELDS 的一种选择	(120)
4.9	行级 BROWSE 确认和触发器	(120)
4.10	限制 BROWSE 范围: FOR 和 KEY	(133)
4.11	用 BROWSE 建立良好的关系	(136)
4.12	单个 BROWSE 中的一对多	(140)
4.13	格式化 BROWSE 窗口	(142)
4.14	分区选择	(143)
4.15	修改 BROWSE 窗口的 FONT 和 STYLE	(144)
4.16	窗口浏览	(146)
4.17	BROWSE 标题	(153)
4.18	BROWSE 和源文件	(154)
4.19	不需要 PREFERENCE	(157)
4.20	维护 BROWSE 窗口	(159)
4.21	“防弹的” BROWSE	(162)
4.22	多用户 BROWSE	(163)
4.23	BROWSE 的限制	(168)
4.24	BROWSE 综述	(174)
第五章	掌握报表生成器	(175)
5.1	拷贝报表	(175)
5.2	在报表生成器中操作对象	(175)
5.3	通过例子学习——A/R 报表	(176)
5.4	使用报表变量	(188)
5.5	在报表中使用 UDF	(193)
5.6	在报表中使用 UDFs 的策略	(195)

5.7 有 SET SKIP 的一对多报表	(196)
5.8 理解打印驱动程序	(201)
5.9 GENPD.APP 剖析	(206)
5.10 打印机驱动程序剖析	(208)
5.11 用用户过程来显示报表的状态	(212)
5.12 本章小结	(216)
第六章 Rushmore 技术和 SQL	(217)
6.1 介绍	(217)
6.2 把 XBASE 的专门术语转换为相关的术语	(217)
6.3 Rushmore 技术	(218)
6.4 Rushmore 的限制	(222)
6.5 把 Rushmore 设置为断开方式	(222)
6.6 优化 Rushmore	(223)
6.7 测试我们的假设	(225)
6.8 FoxPro 2.5 的 SQL 命令介绍	(230)
6.9 用 SQL 选择数据	(241)
6.10 高级 SQL 查询	(251)
6.11 优化 SQL	(267)
6.12 SQL 概述	(268)
第七章 掌握 FoxPro 2.5 API	(269)
7.1 工具集	(269)
7.2 本章约定	(270)
7.3 API 基础: FoxPro API 库的分析	(270)
7.4 库例程	(272)
7.5 事件编程	(276)
7.6 识别 FoxPro 的键码	(278)
7.7 库实例	(279)
7.8 API 函数及其性能	(290)
7.9 在 FoxPro 2.5 库构造工具集中的新内容	(291)
7.10 命令表达式例程	(293)
7.11 调试 API 例程	(299)
7.12 低级文件的输入和输出	(302)
7.13 备注字段的输入/输出	(304)
7.14 内存管理和字符串处理	(306)
7.15 内存变量和数组的管理	(309)
7.16 菜单管理	(311)
7.17 返回函数	(316)
7.18 用户接口控制	(317)
7.19 用动态连接库扩展 FoxPro for Windows	(349)

7.20	RegFn () 和 CallFn () 合理使用	(354)
第八章	菜单管理	(355)
8.1	引言	(355)
8.2	菜单历史	(355)
8.3	FoxPro 2.5 系统菜单	(358)
8.4	新的命令和函数选择项	(366)
8.5	管理菜单生成器	(384)
8.6	菜单生成器剖析	(401)
8.7	用户菜单小结	(418)
8.8	小结	(421)
第九章	屏幕管理	(423)
9.1	屏幕结构概述	(423)
9.2	屏幕对象和事件驱动应用程序	(424)
9.3	新的屏幕对象	(425)
9.4	READ 命令的进一步探讨	(454)
9.5	掌握屏幕生成器	(467)
9.6	建立 SCREEN1 屏幕集	(481)
9.7	小结	(504)
第十章	DDE 和 OLE 对 FoxPro for Windows 的增强	(535)
10.1	引言	(535)
10.2	动态数据交换基础	(535)
10.3	对象连接与嵌入 (OLE)	(549)
10.4	本章小结	(555)
附录 A	数据结构和实例中的数据	(556)

第一章 概 述

1.1 什么是 FoxPro 2.5

FoxPro 2.5 是 Microsoft 公司自 1992 年中兼并了 Fox 软件公司的第一个 FoxPro 版本。Fox 软件公司 1989 年推出 FoxPro 第一版,并在此之前推出了 FoxBASE⁺。FoxPro 2.5 由一系列适用于 MS-DOS、Microsoft Windows、UNIX 和 Macintosh 的 FoxPro 软件包组成,所提供的跨平台特征将允许我们在一个平台上建立应用程序,并在其他平台上运行。

运行于 MS-DOS 下, FoxPro 2.5 更像 FoxPro 2.0, 它支持面向字符窗口 (C.O.W) 界面和伪图形屏幕对象,如下推按钮、复选框、无线按钮 (单选钮) 和弹出菜单等。因为 FoxPro 2.0 和早期的 FoxPro 版本是按照类图形界面设计的,所以 FoxPro 应用程序可以顺利地转成真正的图形用户界面 GUI。而在 Microsoft Windows 下,与 MS-DOS 相对应的那些 FoxPro 屏幕对象,具有直观的 Windows 界面。例如,在 FoxPro for MS-DOS 中,下推按钮是一个由尖括号括起的文本符号,而 FoxPro for Windows 不支持 Windows 的某些其他特殊功能,如动态数据交换 (DDE) 和对象连接与嵌入 (OLE)。

什么是 FoxPro? 人们常常称之为数据库管理系统 (DBMS),但是这个称谓并不能充分描述 FoxPro 的所有功能。FoxPro 提供下列功能:

- 交互数据库工具如 RQBE、BROWSE 和其他菜单驱动数据管理功能。
- 可视应用工具包括屏幕生成器、菜单生成器和报表生成器。
- 具有类 SQL 扩展的完整 XBASE 编程语言,可以兼容 Borland 公司的 dBASE IV 和 Computer Associates 公司的 dBase。
- 可视应用调试工具包括调试窗口、跟踪窗口和 5 个集成编辑器。
- 一个开放的打印机驱动系统,以便在 MS-DOS 下打印 FoxPro 报表。
- 一个提供重新定义菜单、鼠标等功能的窗口用户接口。
- 全图形和字型支持的 FoxPro for Windows。
- 有用的桌面附件,包括编辑器、计算器、日历/日记,ASCII 码表以及在 FoxPro for MS-DOS 中的大型记录器。
- 扩展 FoxPro 2.5 的 C 语言应用编程接口 (API) 的外部例程 (External Routine)。
- 集跟踪源文件,建立应用程序及运行于一体的应用项目管理器。
- 一个 .EXE 编译器,可以生成不进入 FoxPro 环境而能直接运行的 FoxPro 应用程序。
- 帮助最终用户生成简单应用程序的应用程序生成器 (FoxApp)。

FoxPro 2.5 向其开发者和最终用户提供了完整的工作环境。使用 FoxPro 2.5,开发者可以设计、调试和管理用户的应用程序。最终用户可以运行开发好的应用程序,解决疑难问题,设定报表格式,和建立数据库等。FoxPro 2.5 甚至可以将电子表格和文字处理软件集成在一起。例如,我们可以把 Lotus 1-2-3 或 WordPerfect 5.1 放在 FoxPro 菜单条中。在 FoxPro for MS-DOS 的菜单生成器中,我们只需简单地键入类似 RUN FOXSWAP WP51

的一条命令,即可以设定菜单条,当选择该菜单条,即可运行 WordPerfect 5.1 软件。Foxswap 是一个在运行指定程序时释放 FoxPro 大部分内存的程序。在 FoxPro for Windows 中,我们可以利用动态数据交换 (DDE) 和对象连接与嵌入 (OLE) 技术启动 Windows 的其他应用程序。FoxPro for Windows 可以担当 OLE 的客户机,也可以担当 DDE 的服务器和客户机。

除以上功能以外, FoxPro 2.5 还可以解决最大达 2MB 的大型数据库。而且不同于其他数据库管理系统。FoxPro 2.5 还可以管理 MEMO 字段中大量的自由格式的、可变长度的文本和二进制数据 (如位图图像或声音)。FoxPro for Windows 具有一个新的字段类型 General, 可以允许我们嵌入和连接 OLE 的对象, 包括图像、电子表格、文本文件和声音等。

FoxPro for MS-DOS 也出现了真正的 386 保护模式版式 (不增加额外成本), 可以充分利用计算机的有效扩展内存。这样, 我们就可以建立高度复杂的应用程序, 甚至可以利用更大的容量 (例如 65 000 个内存变量, 2MB 字符串, 65 000 个数组元素, 225 个工作区, 以及 65 000 字节记录)。不同于 FoxPro 2.0, FoxPro for MS-DOS 的扩充版本与 DOS 保护模式接口 (DPMI) 内存管理器兼容。这样, 我们就可以在 Microsoft Windows 下安全地运行 FoxPro for MS-DOS。

FoxPro for Windows 具有和 FoxPro for MS-DOS 扩充版一样的功能。

FoxPro 2.5 当然不只是数据库管理系统。如果我们称之为集数据库应用开发、运行及分布系统于一体的交叉平台要更准确些。你怎么应用就怎么称呼它吧。

1.2 历史展望

如果刚刚接触 FoxPro, 那么我们先简单回顾一下。最初的 FoxPro 诞生于 1989 年, 是取代 FoxBASE+ 作为 Fox 软件公司的主导产品的, FoxBASE+ 与 Ashton-Tate 公司的 dBASE III PLUS 严格兼容 (许多性能和语句扩展完全一致); 然而 Fox 公司的工程师们还是着手开发 FoxPro, 他们重新设计了用户接口, 从他们的 Macintosh DBMS 产品和 FoxBASE+/Mac 中借鉴了许多有用的东西。他们增设了新的报表生成器, 扩充了编程语言, 增加了与 Ashton-Tate 公司 1988 年推出的 dBASE IV 相应的功能。为了保持与 dBASE IV 的兼容性, Fox 软件包括了一个 dBASE IV 兼容接口 SET COMPATIBLE ON/OFF。

下面是 FoxPro 2.5 的历程:

1984 FoxBASE

1986 FoxBASE+

1988 FoxBASE+/Mac

1989 FoxPro 1.x

1991 FoxPro 2.0 for DOS

1993 FoxPro 2.5 for DOS 和 Microsoft Windows

Fox 软件公司最初推出了与 dBASE I 兼容的系统——FoxBASE。除了更好的功能以外, Fox 软件公司还移植了多用户性能和很久以前就支持 Ashton-Tate 的跨平台 (FoxBASE 是在 SCO XENIX 之下运行的)。Fox 软件公司其次推出的是 FoxBASE+, 增加了与 dBASE III 的兼容性, 扩充了性能使之领先于 dBASE。Fox 软件公司也对编程语言 (起源于 Nantucket 的 Clipper) 进行了重要的扩充, 包括 VALID 子句, 数组和弹出式菜单。不久 Fox 软件

公司就在 Macintosh 计算机上移植了 FoxBASE+/Mac, 这大大激励了 FoxPro 最初设计思想的产生。

由于提高了运算速度和增加了一些关键功能, Fox 软件公司引起了用户和开发者的注意。他们在 Fox 软件和 dBASE 二者择一时正举棋不定。Fox 软件公司也引起了 Ashton-Tate 公司的注意, 它声称 Fox 已对 dBASE 构成了侵权。1988 年 Ashton-Tate 公司对 Fox 软件公司和 Santa Cruz Operation (SCO) 公司提出了诉讼, 控告它们在 FoxBASE+ 中抄袭 dBASE 的结构和系列 (SCO 公司销售 Fox 产品的 XENIX 和 UNIX 版本)。1990 年 12 月 13 日, 美国洛杉矶地区法院判决解除 Ashton-Tate 公司的诉讼, 宣布 Ashton-Tate 公司的 dBASE 版权无效。法官 Terry Hatter, Jr 引证 Ashton-Tate 公司没有透露 dBASE 来源于一个叫做 JPLDIS 的编程语言。

不出所料, Ashton-Tate 公司马上向上级仲裁部门提出上诉, 但有趣的是, 在案件回到地区法院时, Hatter 法官决定改变他最初的判决。由于 Ashton-Tate 公司已经提出了上诉, 从法律程序来说, Hatter 对此案的裁决权已告终止, 由原诉讼所引起的一切事宜也都悬而未决。

1991 年 10 月 11 日, Borland International 兼并了 Ashton-Tate 公司。合并条款之一, 是 Borland 公司应美国司法部的要求, 结束与 Fox 软件公司的法律争端, 并允许其他公司使用 dBASE 而不会受到法律行动的威胁。接着, 1992 年 3 月 25 日, Microsoft Corporation 宣布, 它将兼并 Fox 软件公司, 并将使 FoxPro 成为世界上最大的软件公司的主导数据库系统。这个愿望于 1992 年 6 月实现了。Fox 软件公司的大部分开发队伍, 包括其总裁 David Fulton 博士, 都搬到 Microsoft 的 Washington 总部 Redmond。

随着时间的流逝, FoxPro 与 dBASE 语法的兼容性变得越来越不重要了。FoxPro 的特点是不仅需要不同的语法, 而且需要不同的方法。尽管人们在努力创建 ANSI 标准的 XBASE 语言, FoxPro 在非程序化文件存取、可视编程和事件驱动操作等方面还是与 dBASE 的兼容性背道而驰。不过 FoxPro 2.5 仍然提供了 dBASE IV 兼容模式, 但是采用这种模式就要舍弃 FoxPro 的许多最有成效的功能。

FoxPro 的开发者们计划推出 FoxPro 的客户/服务器版本, 以便与 SQL 服务器和 Oracle 服务器等关系数据库服务器连接, 但公司还没有宣布启动日期。Microsoft 的软件驱动程序将允许 FoxPro 的应用程序与大量的 back-end 数据库连接。

在客户/服务器结构中, 数据库处理和应用处理分为在局部处理器之间和在远程处理器之间, 还可以在局域网 (LAN) 或广域网 (WAN) 中实现。数据库处理 (索引和查找等等) 发生在 LAN 数据库服务器、小型计算机或主机 DBMS 之中。应用处理 (屏幕格式、报表、商务逻辑和用户接口元素) 则在工作站中运行。客户/服务器处理的优点包括集中式数据库管理, 安全性和具体事务处理。

1.3 本书包括哪些内容

《FoxPro 2.5 开发人员手册》为 MS-DOS 与 Microsoft Windows 单用户和多用户专业应用程序开发提供了技术、策略和方法。由于 Microsoft 公司承诺 FoxPro 2.5 与 FoxPro 未来的 Macintosh 和 UNIX 版本高度兼容, 因此会发现本书对这两个系统也很有用。

《FoxPro 2.5 开发人员手册》给出了大量的 FoxPro 函数、命令和工具的用法, 指导我

们顺利完成应用程序开发。本书详细研究了 FoxPro 的主要部分,并提供了它们的使用方法。本书给出了整个开发过程的逻辑顺序。它提供了一条避免误入歧途的捷径,使我们更快地编程,更容易的调试和更简便地维护应用程序。

此外,本书阐述了 FoxPro 2.5 的一些主要特征,包括应用项目管理,SQL SELECT 语法和 FoxPro 2.5 集成屏幕与菜单生成器。本书提供了高级编程的特殊技巧,如多窗口操作、增强的 BROWSE 控制、MEMO 字段和字符串处理、查表窗口、屏幕设计、多用户并行管理和应用程序安全性等。

本书也包括 FoxPro for Windows 的专有平台特征,其中有对象连接与嵌入 (OLE) 和动态数据交换 (DDE)。本书解释了怎样直接在 FoxPro 的新 General 字段中存储电子表格和图形,怎样最好地应用字体、字型 and 颜色等。

范例程序提供了清晰的、通用性好的代码。我们可以读懂它们,并经过简单地改动可以在不同的上下文里应用它们。这些代码都经过调试处理,满足专业标准。

本书也包括一些高深的专题,如 FoxPro for MS-DOS 打印机驱动程序和应用程序对象表(对象表包括项目、菜单和屏幕的全部规范。由于它们都是 FoxPro 数据库文件,所以开发人员可以直接存取它们以建立用户开发工具)。此外,《FoxPro 2.5 开发人员手册》还向我们展示了编制 FoxPro 2.5 扩展程序库 (API) 的基本知识。在那一部分,需要一个 C 编译器、一些高级编程经验。

尽管本书是针对 FoxPro for MS-DOS 和 FoxPro for Windows 的,但书中大部分的技术和策略也适合于 FoxPro for Macintosh 和 FoxPro for UNIX。

1.4 本书对你合适吗

阅读本书根本不需要有计算机专业的学历,但具有下列必备条件将有助于掌握本书的绝大部分内容。

(1) 在任何版本的 FoxPro 上的编程经验。如果已经在 FoxPro 1.x、2.0 或 2.5 上编写过程序,那就完全可以准备阅读《FoxPro 2.5 开发人员手册》。

(2) 在其他任何 XBASE 语言上的编程经验。如果在下列任何系统上编写过程序,那么通过《FoxPro 2.5 开发人员手册》学习 FoxPro 2.5 几乎没有什么困难。这些系统是: FoxBASE+ 或 FoxBASE+/Mac、dBASE I、dBASE II PLUS 或 dBASE IV; Computer Associates 公司的 CA-Clipper; WordTech System 公司的 dBase 或 Quicksilver。其他几个公司也开发了一些版本的 XBASE 语言。具有这些语言的任何一种的编程经验,都可以准备阅读下面的内容。

(3) 其他语言的一般编程经验。这些语言如 BASIC、C 或 Pascal。作为一个受过训练的编程人员,可以很容易地掌握 FoxPro 2.5 的编程概念。需要在 FoxPro 2.5 手册上学习 FoxPro 2.5 的命令和函数语法或细读 FoxPro 的联机帮助信息。

(4) 交互 FoxPro 应用。如果没有编写过程序,但明白交互 FoxPro 和一般数据库管理的工作原理,那么在手边放一本 FoxPro 手册,并经常通过热键 F1 提请帮助。注意,可以录入本书的许多范例程序,并在没有改动的情况下运行它们。如果不想录入,包括本书所有代码的程序盘价格适中,订单的详细内容见本书的背面。

1.5 了解 FoxPro

1.5.1 FoxPro 结构

在 FoxPro 的接口中,每个窗口都自备 I/O 设备驱动器,如数据输入、浏览和程序编辑。这些窗口可以重新设定,可以进行放大和缩小。MS-DOS 和 Windows 版本中,接口的一些性能也不一样。例如,在 MS-DOS 中,可以在屏幕右侧的底部也设置一个窗口,但 Windows 版本不支持这个功能。

FoxPro 的事件驱动接口允许我们使用鼠标、对话驱动工具来设计、调试和管理应用程序,这样就可以同时打开不同的窗口。例如,我们可以同时打开 10 个编辑窗口,5 个 BROWSE 窗口和 2 个屏幕生成器窗口。然后我们就可以用鼠标和键盘来选择它们。

FoxPro 也允许我们开发事件驱动应用程序,就像 FoxPro 开发环境一样,具有多输出窗口、下推按钮、无线按钮、弹出式菜单、文本框、对话框等。用户可以用鼠标(如果必要可以用键盘)操作应用程序。使用 FoxPro for Windows,可以在应用程序中加入图形、电子表格、文件、视像、声音等多媒体数据。当然,计算机要有声控板或图形板等必备的硬件。

为帮助我们以事件驱动方法开发窗口应用程序,FoxPro 2.5 使用了可视编程工具。例如,FoxPro 的屏幕生成器就通过在屏幕上设置对象来建立格式和对话区。有了每个屏幕和屏幕对象(如下推按钮或列表框),我们就可以把定义对象功能的 FoxPro 代码联系起来。我们可以通过 Screen Painter 全部定义和保留这些代码“片段”(就像 FoxPro 文件所称的那样)。这些代码片断贮存在 FoxPro 的 MEMO 字段中。FoxPro 的 Menu Builder 的工作原理与此一致。

使用这种方法,早期的 Fox Software,以及现在的 Microsoft 解决了怎样修改已生成的应用程序的传统代码生成器问题。应用典型的代码生成器,我们通过在屏幕上填空和获取对象的形式生成一个规范说明,系统分析这个规范说明并生成一个进程。规范说明输入系统生成源代码。随之而来的问题是,典型代码生成器的处理能力是有限的。编程语言本身比高水平的规范说明工具在性能和处理能力上要丰富得多。现在麻烦是,如果直接修改生成的源代码,那么在不重写修改的情况下根本就不能使用高水平的工具。

FoxPro 解决这一问题的方法是,在重新生成应用程序时管理好客户代码片断并保存修改的内容。FoxPro 2.5 中我们通过在 Screen Builder 中直接编辑代码片断表修改绝大部分应用程序。当重新建立一个屏幕,则最后的代码片断被保留下来。

有了集成工具箱,FoxPro 2.5 可以使我们建立面向窗口的编辑系统,尽量减少人工编辑;然而,如果宁愿使用自己的编辑工具和技术来开发应用程序,FoxPro 也不反对。可以使用自己的编辑器和代码生成器。高级工具所提供的所有功能在低水平的 FoxPro 2.5 命令和函数下依然适用。例如,利用 Screen Builder 使用下推按钮在屏幕位置 10, 15 处设定一个屏幕,可以在菜单中选择 Push Button,然后将其放在 10, 15 处的表格中。如宁愿用命令语言设定同一个对象,可以使用以下命令:

```
@ 10, 15 GET (memvar)
  PICTURE " @ * HN Push Me";
  SIZE 1, 10, 1;
  DEFAULT 0
```

FoxPro 2.5 让用户选择手写还是生成代码。在大部分应用程序中,可能两个都选。

1.5.2 外部例程 API

FoxPro 2.5 开创了开放结构的新天地。所谓开放结构也就是编程者可以修改和扩充 FoxPro 的程序。开放结构的一个特殊部分就是 FoxPro 的外部,例程应用编程接口 (External Routine API), 是一个管理 FoxPro 和外部程序间交互作用的函数集合。这些函数是可选 Library Construction Kit 的一部分,是每个开发者不可缺少的。外部例程 API 允许我们调用 C 语言编写的外部程序,就像它们是 FoxPro 内部函数一样调用。为 API 编写的一个或多个外部函数库可以在运行时被调用,其命令是 SET LIBRARY TO <library> ADDITIVE。

简单执行外部例程是许多数据库开发系统的基本特征。FoxPro 2.5 却不是这样。它提供了 FoxPro 内部设备存取。包括:

- 内存管理例程
- 文件输入/输出例程
- 备注字段例程
- 对话框
- 菜单 (系统和用户)
- 键盘、鼠标和空事件
- 视频输出例程
- 内存变量和数组
- FoxPro 窗口

此外,调用 API 可以执行 FoxPro 命令、求 FoxPro 表达式的值和进行浮点运算。事实上,外部例程 API 可以利用 FoxPro 2.5 创建高度集成的系统。对于有经验的编程人员,API 代表了实现高速操作、专用计算程序、日常事件处理等的无限可能性。它也代表了通讯、进程控制和设备处理等专用库的巨大机会。商用库的开发者也将很快开发 API 的潜能。

1.5.3 可重定义的菜单

开放结构的概念也要求高水平的 FoxPro 2.5 设备。例如,我们可以用 FoxPro 命令或使用 FoxPro Menu Builder 重新定义 FoxPro 系统菜单。我们可以增加或移动菜单条,建立新的弹出式菜单,和改变已有菜单选项的顺序或功能。作为一个开发人员,可以将熟悉的 FoxPro 2.5 环境传送给用户。作为一个最终用户,可以为日常 FoxPro 2.5 操作建立菜单选择。

1.6 数据驱动应用程序生成和工具

FoxPro 2.5 在标准数据库文件中存储了屏幕、菜单、标签和项目的定义。这意味着我们可以应用 FoxPro 编程语言或其他任何可读数据库文件的工具,直接控制这些应用对象。为使这些对象定义可存取,开发人员可以建立用户工具,以创建、维护和验证应用程序。

FoxPro 的对象定义的目的是:FoxPro 应用屏幕和菜单表产生应用程序代码 (标签和报告不能产生代码)。屏幕和菜单表中的数据输入到 FoxPro 的代码生成器中,分别应用样板文件 (GENSCRN.PRG 和 GENMENU.PRG) 处理输出结果。

例如,Screen Builder 应用扩展名为 .SCX 的数据库文件和扩展名为 .SCT 的备注文件

存储代码片断和定义接口对象及其属性。当我们存储一个新的屏幕，FoxPro 建立一个包含字段如 OBJTYPE、OBJCODE、NAME、VPOS、HPOS、COLORPAIR、VALID、FLOAT、CLOSE 等的 SCX 文件。每个屏幕对象都有自己的记录。

当我们产生代码时，FoxPro 运行 Screen Builder 的样板程序 GENSCRN.PRG。GENSCRN.PRG 利用 FoxPro 的文本连接功能，在 FoxPro 的命令下，将 SCX 和 SCT 文件中的数据 and 代码片断组合在一起。结果是 Screen Builder 中定义的属性和代码组合成一个 FoxPro 程序。

例如，我们可以应用 Screen Builder 建立一个名为 MAINSC 的屏幕，包括一个框和一个名为 mvar 的单个 GET。存储完定义以后，我们可以使用 USE 命令打开文件 MAINSC.SCX，并显示其内容。其中，我们发现 79 个字段和 3 个记录。第一个记录包含基本的屏幕设置数据。第二个记录包含框的定义，第三个包含 GET 的定义。下表仅显示了 79 个字段中的 10 个：

	记录 1	记录 2	记录 3
PLATFORM	DOS	DOS	DOS
UNIQUEID	_QCY1BSZ7V	_QCY1BT1GW	_QCY1BT8UQ
TIMESTAMP	440382022	440382024	440382030
OBJTYPE	1	7	15
OBJCODE	63	4	1
NAME	memo	memo	Memo (contains mvar)
VPOS	0.000	0.000	8.000
HPOS	0.000	0.000	7.000
HEIGHT	25.000	7.000	1.000
WIDTH	80.000	8.000	10.000

我们通过检验记录 2 框的定义来说明代码的生成。注意小写字母 memo 标记表示备注字段为空。OBJTYPE 字段为 7，表示是框；OBJCODE 为 4，表明为单线边界。VPOS 和 HPOS 都为 0，表示框起始于交点 (0,0)。同样，HEIGHT 是 7.000，WIDTH 为 8.000。PLATFORM 字段表示对象是用 FoxPro for MS-DOS 建立的，UNIQUEID 和 TIMESTAMP 字段被 FoxPro 的转换器用以帮助管理跨平台开发。

当我们为这个屏幕生成代码时，一个 GENSCRN.PRG 中名为 GENBOXES 的过程处理 OBJTYPE、OBJCODE、VPOS、HPOS、HEIGHT 和 WIDTH 字段并生成画该框的代码。

过程 GENBOXES 中，下面这段程序说明了 FoxPro 的文本合并命令 \ 和 \ 的用法。每一行中，由 \ 或 \ 开始，FoxPro 求 < < > > 内的表达式的值，然后把它们和其他字符文本一起放到输出文件中去。命令 \ 对输出内容增加一个回车和换行指令，命令 \ 继续在输出的同一行中增加内容，没有回车和换行。

```

\@ < <Vpos>>, < <Hpos>>, { <m.bottom>>}, < <m.right>>
DO CASE
CASE objcode = c_sgbox
    m.thisbox = c_single
    \ \ BOX " < <m.thisbox>> < <Fillchar>>"
CASE objcode = c_sgboxd

```

```

m.thisbox = c_double
\\ Box " < <m.thisbox>> < <Fillchar>>"
CASE objcode = c_sgboxp
m.thisbox = c_panel
\\ BOX " < <m.thisbox>> < <Fillchar>>"

```

本段程序中，由 C_ 开始的变量是在样本程序中由 #DEFINED 定义的实常量。

这段 GENSCRN 程序简单地替换了 MAINSC.SCX 中第 2 号记录中的数据，并生成如下下一行代码：

```
@ 0, 0 TO 6, 7
```

当 MAINSCR 传输到 FoxPro for Windows 之后，重新生成了产生框的 Windows 类语法的代码，如下：

```

@ 0.469, 0.438 To 6.532, 7.562;
PRN 1, 8;
STYLE " B"

```

GENSCRN.PRG 包含所能遇到的各种类型对象的专门的处理过程。一个单屏数据库文件（扩展名为 .SCX）可能包含多平台上生成的对象；然而，GENSCRN 只能处理当前平台相应的对象。当我们在不同平台间切换并企图修改屏幕时，FoxPro 的转换程序（TRANSPRT.PRG）要确定对象在另外平台上是否被修改过，并让我们选择对象的最后版本转换到当前平台。

Menu Builder 使用类似的代码生成方法，不过模板不同，名为 GENMENU.PRG。

图 1.1 给出了这种代码生成结构图。FoxPro 2.5 模板文件本身用为 FoxPro 2.5 编写的，所以我们可以根据用户开发目的的不同对模板进行剪裁，而不用学习其他编程语言。改变了模板，也就改变了所有随之生成的代码。

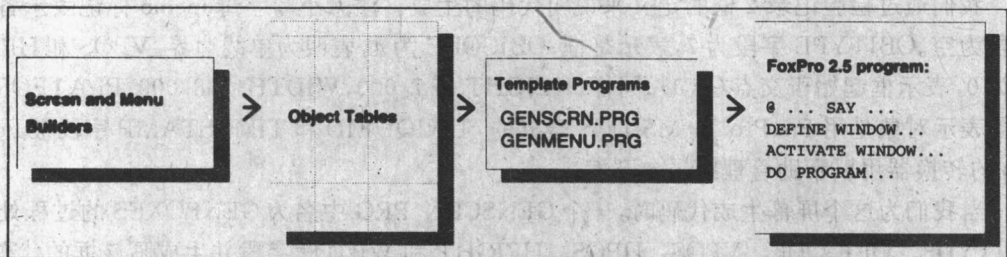


图 1.1 Screen Builder 的代码生成结构

这种数据驱动结构也引起了有趣的问题：应用程序的源代码在哪儿？源代码实际上就是对对象表中的数据。生成的源代码其实是中间代码，我们郑重强调不要改动它。如果改动了中间代码，那么当重新生成应用程序时，那些修改将被覆盖掉。我们可能为编程提示而读中间代码的。

▼警告：FoxPro 的屏幕文件（扩展名 .SCX）的结构随版本不同而改变。应该避免依赖这种结构的硬编码（hard-coding）程序。

1.6.1 菜单生成器

FoxPro 2.5 菜单生成器, 让我们用最少的编程最快地设计出用户菜单系统。当我们告诉 FoxPro 2.5 CREATE MENU (建立菜单), 屏幕上就会显示一个设计窗口, 我们可以输入菜单笺 (menu pad)、弹出式菜单、子菜单等。我们可以打开多菜单设计窗口, 其限制仅仅在于可用内存容量和文件处理。

为修改原系统菜单, 选择 FoxPro 的 Menu Builder 提供的菜单选项 Quick Menu, 它装载了缺省的 Systems Menu (系统菜单) 的定义。其结果显示在图 1.2a 和图 1.2b 中, 分别是 FoxPro for MS-DOS 和 FoxPro for Windows 的。我们可以修改这些定义, 增加新的填充模式和增加新的子菜单。我们也可以为每个菜单笺 (Menu pad) 的结果赋值。结果不是选择 Menu Pad 或 Menu Bar 之后发生的动作。可能的结果为另外一个菜单笺名。

当我们选择一个过程作为结果, FoxPro 提示我们输入代码, FoxPro 将这些代码和菜单定义一起存储在扩展名为 .MNX 的数据库文件中。如果要修改与菜单相关的代码, 只能通过 Menu Builder 来做。

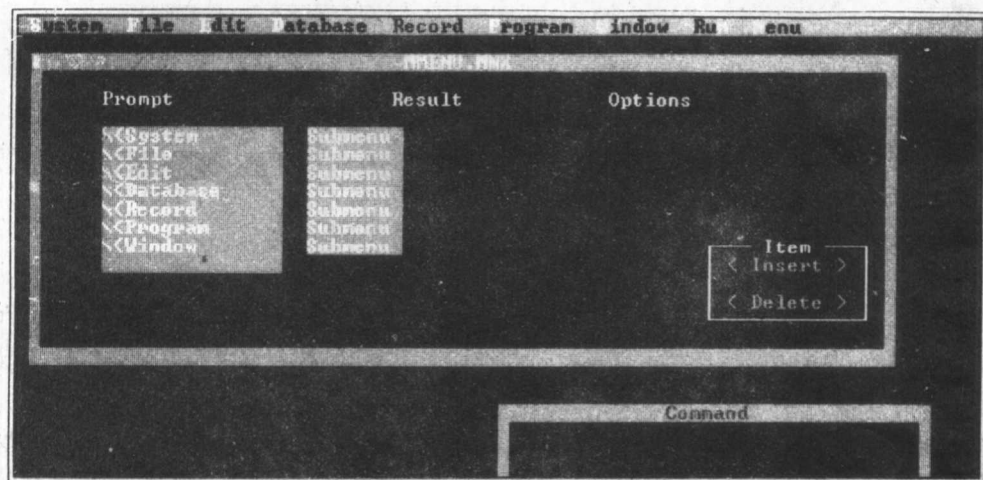


图 1.2a FoxPro for MS-DOS 系统菜单定义

这种形式的程序有时被称作事件驱动程序, 因为程序的运行只和菜单笺的选择有关, 事件就成为鼠标单击 (click) 或按钮选择。

Menu Builder 也允许我们输入 Setup 和 Cleanup 代码片断。我们可以使用 Setup 代码片断来准备环境, 打开数据库文件, 保存以前的显示菜单, 和定义内存变量。Cleanup 程序存储以前的环境设置, 关闭数据库, 和控制其他菜单, 用户可以自己选择。

Menu Builder 提供了几个其他功能, 使我们能够充分利用 FoxPro 自己的系统菜单。例如, 我们可以在不编程的情况下为许多 Menu pad 定义简化键。我们也可以定义一个菜单系统相对于当前菜单系统的位置关系。我们可以在当前菜单系统的右面再增加一个菜单系统, 或替代当前菜单系统。我们也可以在另一个菜单条之前或之后插入一个菜单条笺。