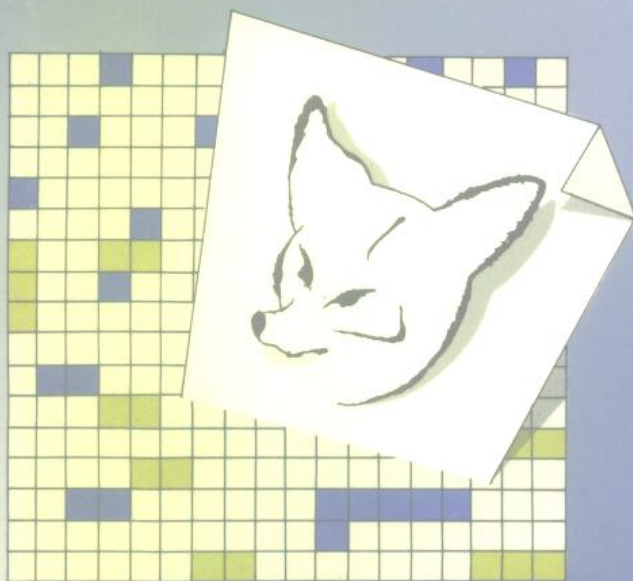


FoxPro 2.6 for Windows

程序设计

周诚彪 俞一峻 编著



复旦大学出版社

FoxPro 2.6 for Windows 程序设计

周诚彪 俞一峻 编著

复旦大学出版社

195503

FoxPro 2.6 for Windows 程序设计

周诚彪 俞一峻 编著

复旦大学出版社出版

(上海国权路 579 号)

新华书店上海发行所发行 复旦大学印刷厂印刷

开本 787×1092 1/16 印张 18.5 字数 459,000

1996 年 5 月第 1 版 1996 年 5 月第 1 次印刷

印数 1—5,000

ISBN7-309-01659-9/T·143

定价: 25.00 元

前 言

近十年来,PC 机性能持续高速地发展,PC 平台的应用从无到有,从简单到复杂,呈现出一片繁荣旺盛的态势。作为现代应用软件开发的关键支撑软件,PC 平台的数据库产品日益受到众多数据库厂商和用户的关注。

FoxPro 是 Microsoft 公司开发的新一代数据库产品。作为 PC 平台运行最快的数据库管理系统,FoxPro 具有的两个特性非常引人注目:其一,FoxPro 保持了对 dBASE, FoxBASE 等 XBASE 产品的兼容性,使得大量的软件投资得以保护。这点对于我国过去用 XBASE 开发的许多应用尤为重要;其二,由于提供了功能强大的应用开发工具(特别是各种可视化工具和向导工具)和 SQL 风格的数据库管理功能,FoxPro 正体现出了一种优秀的数据库应用开发环境的特征。可以预期,在未来的几年,FoxPro 仍将是 PC 平台上极其重要的数据库产品。为了让广大读者更好地学习和掌握 FoxPro 程序设计功能,我们着手编写了这本书。

由于 FoxPro 软件融合了数据库管理、GUI 界面开发工具和程序设计语言等多种内容及多平台支持的产品特征,如何在有限的篇幅中把最有效的内容介绍给读者是我们选材的出发点。我们在编写中突出了以下的特点:第一,本书只讲述 FoxPro for Windows 版本的内容。这主要考虑到 Windows 平台的广泛流行与 FoxPro 应用良好的跨平台移植特性;第二,FoxPro 是个不断更新的数据产品,为保持对原先版本的兼容性,FoxPro 有趋于复杂的倾向,在本书中,我们的重点着眼于最新的 2.6 版本的语言和工具特征;第三,FoxPro 的程序设计可以分为两种模式:用手工书写代码的语言模式和采用各种工具开发的工具模式。语言编程是最基础也是最根本的开发手段;而工具开发有其高效和易操作性的优点,我们在本书中努力讲清如何结合两种模式的长处来进行应用设计。具体的内容偏重于 FoxPro 的语言描述,工具的讲解强调其功能描述而忽略其操作的细节。

FoxPro 的程序设计是一门极富魅力和技巧的创造性活动,为此在本书编写时我们没有采用教条式的编写风格。我们的目的不是教给读者所谓的程序设计“秘诀”,更大的期望是让读者学会亲自实践的能力。由于好的实例在讲解时具有很好的示范性,本书的许多内容都通过一个具体的应用开发例子而展开。本书的编写采用了分课时和单元的结构,教学内容相对固定的课时结构更有利于读者循序渐进地学习。

在本书的编写过程中我们得到了许多人士的帮助,夏玲明女士和王蔚女士完成了大量的文稿工作,谨向她们致以衷心的感谢。需要特别地感谢孙未未编辑,没有他热情的鼓励和帮助,本书难以成稿;谢铭培教授审阅了全书,并且提出了很多宝贵的意见,在此一并致谢。

由于时间仓促,编者水平有限,本书可能存在着疏漏和不足之处,恳请各位专家和读者不吝指教。

编 者

1995 年 9 月

目 录

前 言	1
-----	---

第一单元 程序框架的构造

第 1 课 窗 口	2
1.1 什么是窗口	2
1.2 可视化编程快速入门：“Hello World”	4
1.2.1 打开屏幕设计窗口	5
1.2.2 定制界面窗口及显示内容	6
1.2.3 产生 SPR 文件	8
1.3 解剖 HELLO.SPR	9
1.4 窗口的定义	10
小 结	14
练 习	14
第 2 课 条形菜单系统的组成	15
2.1 条形菜单系统的组成	15
2.2 程序实例	17
2.2.1 启动菜单生成器	18
2.2.2 定制 AIDEMO 的菜单系统	18
2.2.3 生成 MPR 文件	21
2.3 透视 DEMO.MPR	21
2.4 运行 AIDEMO 应用程序	22
2.5 菜单的语言定义	24
2.5.1 菜单条的创建	24
2.5.2 菜单面板的创建	25
2.5.3 弹出菜单的创建	27
2.5.4 弹出选项的创建	28
2.5.5 菜单事件	29
小 结	30
练 习	30
第 3 课 输出对象	32
3.1 增加数据成分	32
3.2 扩展 AIDEMO	34
3.3 运行新的 AIDEMO 程序	37
3.4 程序模块调用关系	39
3.5 用于输出的 SAY 命令和按钮命令	41

3.5.1 有格式输出的 SAY 命令	41
3.5.2 创建 Push Button 按钮的 GET 命令	44
小 结	46
练 习	47
第 4 课 输入对象	49
4.1 实例要求	49
4.1.1 修改 InfoOut 输出屏幕	49
4.1.2 设计 InfoIn 输入屏幕	51
4.2 追求更好的格式控制	55
4.3 函数/过程名字的含义	58
4.4 编辑区对象的定义	59
4.4.1 实现编辑区对象的@...GET 命令	59
4.4.2 SQL INSERT 命令	63
小 结	65
练 习	65

第二单元 语 言

第 5 课 数据对象	68
5.1 数据属性	68
5.1.1 数据对象的标识	68
5.1.2 数据类型	69
5.1.3 数据值和数据宽度	74
5.2 动态的数据类型	75
5.3 数据对象的形式和生存期	75
5.4 内存变量的基本操作	75
5.4.1 创建内存变量	76
5.4.2 释放内存变量	77
5.4.3 查看内存变量	78
5.5 数 组	79
小 结	80
练 习	80
第 6 课 表达式	81
6.1 操作数	81
6.2 运算符	82
6.2.1 字符型运算符	82
6.2.2 日期型运算符	82
6.2.3 逻辑运算符和关系运算符	83
6.2.4 数值运算符	85
6.3 运算符优先级	86
6.4 函 数	87
6.4.1 数值和数学函数	88
6.4.2 字符函数	89
6.4.3 日期函数	93

6.5 内部函数应用实例	95
小 结	97
练 习	97
第7课 结构化控制语句	98
7.1 编写程序的准备	98
7.2 控制结构概况	100
7.3 分支语句	100
7.3.1 IF 语句	100
7.3.2 CASE 语句	101
7.4 循环语句	104
7.4.1 FOR 循环语句	104
7.4.2 DO WHILE 循环语句	107
7.5 两个特殊的控制	109
7.6 控制语句的嵌套结构	110
7.7 逐步求精的程序开发原则	111
小 结	114
练 习	115
第8课 过程和用户自定义函数	116
8.1 过 程	116
8.1.1 创建过程模块	117
8.1.2 过程的参数传递	121
8.2 用户自定义函数	122
8.2.1 函数的创建	123
8.2.2 函数的引用	124
8.3 函数还是过程	125
8.4 变量的全局/私有属性	127
8.4.1 申明全局变量	127
8.4.2 申明私有变量	127
8.5 数组的参数传递	129
8.6 模块间的数据通信	130
小 结	132
练 习	133

第三单元 数据库基础

第9课 数据库的创建和打开	136
9.1 什么是数据库	136
9.2 创建数据库	138
9.2.1 数据库规划	138
9.2.2 确定主关键字	139
9.2.3 确定字段类型	140
9.3 CREATE TABLE 命令	141
9.4 CREATE CURSOR	143

9.5 打开数据库	144
9.6 关闭数据库	146
9.7 工作区的选择	147
9.8 查看数据库全局信息	148
小 结	150
练 习	150
第 10 课 数据库内容的操作	152
10.1 记录的定位操作	152
10.1.1 记录的定位与记录指针	152
10.1.2 EOF()和 BOF()函数	154
10.2 增加记录	155
10.3 删除记录	157
10.4 恢复命令	158
10.5 记录的修改	159
10.6 逻辑视区:数据库的投影和选择	161
10.7 选择记录范围	163
10.8 空 字 段	164
10.9 完善 AIDEMO 的数据处理	166
10.9.1 更多的数据库表格	166
10.9.2 GRADE 表格的输入屏设计	167
小 结	171
练 习	171
第 11 课 索引和排序	173
11.1 什么是索引	173
11.2 FoxPro 的索引文件类型	175
11.3 索引文件的创建	176
11.4 选择索引表达式	178
11.5 选择性索引	180
11.6 索引的打开、关闭和更新	181
11.7 选定主索引	183
11.8 删除索引	185
11.9 基于索引的查找命令	185
11.9.1 SEEK 命令	185
11.9.2 FIND 命令	186
11.9.3 FOUND() 函数	187
11.9.4 SEEK()函数	187
11.10 使用 SORT 排序命令	188
小 结	190
习 题	191
第 12 课 BROWSE 命令的程序设计	192
12.1 BROWSE 命令概况	192

12.2 定制 BROWSE 窗口	194
12.2.1 几何外观：大小和位置	194
12.2.2 列宽和列名汉化	195
12.2.3 字体的选择	197
12.3 BROWSE 字段的格式化	197
12.3.1 导出字段(Calculated Fields)	198
12.4 BROWSE 的有效性验证	200
12.5 BROWSE 窗口的标题和名字	204
12.6 窗口分区和列表选择	206
12.7 用 BROWSE 显示多表内容	207
小 结	208
练 习	209

第四单元 高级技术

第 13 课 SQL 查询	212
13.1 SQL SELECT 命令概述	212
13.2 增加 SQL SELECT 所需的数据	214
13.3 FROM 子句指定查询的原始表格	214
13.4 SELECT 指定输出字段	215
13.5 WHERE 子句实现联接条件和选择	218
13.5.1 WHERE 实现元组选择	218
13.5.2 WHERE 实现表格间的联接操作	219
13.5.3 WHERE 子句的谓词搜索	221
13.6 GROUP BY 子句和列函数	224
13.7 HAVING 子句实现分组选择	226
13.8 ORDER BY 子句实现有序输出	227
13.9 用于合并运算的 UNION 子句	228
13.10 输出重定向	228
13.11 什么是 RQBE	230
小 结	230
练 习	231
第 14 课 Rushmore 技术	233
14.1 Rushmore 技术概述	233
14.2 数据准备和测试手段	235
14.3 可利用 Rushmore 技术的命令集	237
14.4 正确使用 Rushmore 条件	238
14.5 基本优化表达式	240
14.6 基本表达式的组合	241
14.7 复杂表达式的组合	243
14.8 更好地使用 Rushmore	244
14.8.1 保持适当的索引数量	244

14.8.2 在索引表达式中避免使用 NOT(1)和 FOR	244
14.8.3 保持 SET DELETED OFF	245
14.8.4 用 SET ORDER TO 命令删去所有主排序	245
14.8.5 巧用字符串比较运算“=”	246
14.8.6 有序输出	246
14.9 关闭 Rushmore	246
小 结	247
练 习	247
第 15 课 通用字段、OLE 技术和报表设计	249
15.1 时髦的 AIDEMO	249
15.2 用 OLE 实现通用字段的输入	254
15.2.1 嵌入方式	254
15.2.2 链接方式	256
15.3 报表设计	257
15.3.1 报表打印和预览	258
15.3.2 利用报表书写器设计报表	259
15.4 完善 MMEDIA.PRG 程序	262
15.5 REPORT 和 READ 命令	264
15.5.1 REPORT 命令	264
15.5.2 READ 命令	265
小 结	267
练 习	268
第 16 课 向导工具	269
16.1 向导工具的向导	269
16.1.1 向导工具的启动	269
16.1.2 向导工具的屏幕导航按钮	269
16.2 数据库向导	270
16.2.1 启动数据库向导	270
16.2.2 用数据库向导样本建立个人藏书数据库	271
16.3 屏幕向导	274
16.3.1 启动屏幕向导	275
16.3.2 使用屏幕向导	275
16.3.3 执行屏幕	277
16.3.4 了解生成出来的屏幕	278
小 结	284
练 习	285

第一单元 程序框架的构造

程序用框架是现代应用程序的基础。

我们以 AIDEMO 的实例来介绍如何综合利用屏幕生成器和菜单生成器来快速建立程序框架。

另外,我们还从语言角度对涉及界面对象的创建定义命令作了解释,希望通过这种方式来更好地提高程序框架构造能力。

第1课 窗 口

窗口是 Microsoft Windows 应用程序最显著的外观特性,它也是构造 Windows 应用界面的各种组成元素中最主要的一个。应用程序的开发人员利用窗口,辅以其他必要的界面组成元素:菜单、命令按钮、输入/输出控制对象等,为应用程序的最终用户——使用这个程序的个人或集体——提供了一个与程序相互对话、相互交流的舞台。

利用一件新事物的前提,首先是熟悉这件东西。在本课的开始,读者将会学到从用户视角看到的全部窗口(Window)特征。MS Windows 这个 GUI 操作环境之所以风行世界,一个重要的原因是它为每个应用程序定义了一种称之为图形用户界面(GUI)的一致界面风格。因此,对于所有想进入 Windows 应用程序设计(包括 FoxPro For Windows)的读者,第一步就是要了解这种风格、遵循这种风格,从而保证自己开发的程序能融入 Windows 世界。

理解了窗口的用户外观,接着应该开始窗口的设计和开发工作。在本专题,读者将会深刻地体会到 FoxPro 在创建应用程序界面时的强大功能。希望就此读者能进入一条充满自信和喜悦的 FoxPro 学习之路。

实例是最好的设计指导。本课提供一个简单而基本的实例,让读者增强对程序设计的理解。但同时语言是最基本的编程手段。我们希望通过工具演示和语言编程相对照的方式,使读者一方面加深对工具的理解,另一方面学会程序设计的最新手段和方法。

1.1 什么是窗口

当读者在 DOS 的命令行打入 WIN 后,就可用鼠标打开相应的 FoxPro For Windows 的图符(Icon)。在看到 FoxPro For Windows 的主窗口之前,肯定已经看到了很多不同大小、具有不同显示内容的长方形区域,这就是窗口。图 1-1 是一个典型的窗口外观图示。请读者注意,虽然我们正在讲解 FoxPro For Windows(下面简称 FoxPro)应用程序的界面设计,但我们故意把屏幕上其它的长方形区域(而不仅仅是 FoxPro 这个主要矩形区域)作为窗口的外观示例,这是为了进一步强调:窗口是 Windows 环境的概念,用 FoxPro 开发 Windows 下面的应用程序应该遵循这个概念的约定。

图 1-1 显示了窗口的两大组成部分:框架和客户区。框架是位于矩形区域上下、左右和周边的一些小装饰组成的集合。FoxPro 允许程序员根据个人爱好方便地显示(选择)或隐去(舍去)这些装饰。按照术语的说法就叫做定制窗口(Customized Window);而矩形区域中间较大的一块就是客户区。在这里可以显示任何用户感兴趣的输出信息,犹如程序开发人员提供给用户的一个演出舞台。



图 1-1 窗口的外观

具体地说,窗口是由下面的一些部件组合而成。

标 题

标题水平地位于窗口顶端,主要用来显示一个称之为标题(Title)的简短字符串,以反映窗口的功能;标题区的颜色变化还能反映出窗口的活跃状况;另外,当窗口是可移动时,在标题区按下鼠标(Mouse)左键并拖动,还能让用户随意移动窗口。这是窗口框架不可缺少的部件。

控制菜单

也称为窗口系统菜单框。它用来显示这个窗口允许的动作列表。通常有还原、移动、大小、最小化、最大化和关闭选项。Windows 支持一个重要的缺省动作,就是双击此框,即可以关闭该窗口。

最小化按钮

在这个按钮上面按下鼠标左键,窗口就会缩成一个叫图符的图案。它对应系统菜单框中的最小化选项(Min)。

最大化按钮

在这个按钮上面按下鼠标左键,窗口会填满父窗口的客户区,即它的最大的形状。它对应系统菜单框中的最大化选项(Max)。

边 框

边框(Border)位于窗口这个矩形区域的最外面。FoxPro 允许有 System、Panel、Double、Single 和无边框等多类边框风格,它们的差异大致在边框的厚薄上面。当窗口能重定大小时,用户在边框上按下鼠标左键并拖动,能够随意改变窗口的大小。

状态框和菜单

状态框和菜单经常跟窗口联系在一起,但 FoxPro 不把它作为窗口的框架组件。下一课将会看到菜单的介绍内容。程序员对窗口的设计,很大程度上说就是对以上几个部件属性的裁剪设计。在本课的实例中能看到 FoxPro 提供裁剪的种种手段。

客 户 区

客户区是窗口这个大矩形中的一个内部矩形。与早期 dBASE 或 FoxBASE 系统开发的程序大量占据整个显示屏幕的情形相比较,窗口的概念给了程序员输出信息时有更大的灵活性。因为只须遵循定义窗口、激活窗口、输出显示(用 say 或? 等 FoxPro 命令)这种方式,程序员就可以在不同的窗口上显示不同的内容!这正是窗口客户区的特殊功能。尽管没有类似框架部件的属性,客户区还是隐含着重要的性质:大小。呈现给用户的输出实质上是在这个大小的区域中完成的!特别由于窗口框架部件的大小都相对固定,所以窗口的大小直接影响着客户区的大小。因此,当定义窗口的大小时,实质上是在定义客户区的大小。

1.2 可视化编程快速入门:“Hello World”

当了解窗口的外观特性后,再来学习有关窗口程序设计的课题,就变得相对简单。在早期,由于缺乏必要的工具支持,开发程序的图形用户界面(GUI)往往是一件耗时又费劲的工作。但自从 Microsoft 公司推出商业化的 Visual Basic(VB)产品后,可视化的程序设计风格很快使程序员从繁重的编码中解放出来。这种工具使得程序员可以更加专注于界面部件的位置、大小、色彩等要素在美学上的思考。FoxPro For Windows 一个很重要的贡献就是也提供了类似 VB 的可视化界面设计工具。这一点对于所有的 FoxPro 程序员都是一件幸运的事情。下面我们将沿着一个很基本的实例流程,领略一下可视化工具设计的强大功能。

我们将要设计的第一个实例是许多关于 GUI 设计时的入门例子:“Hello World”。我们的要求是:产生如图 1-2 所示的窗口,里面显示一条欢迎信息,用户可以用鼠标点击该窗口而退出应用程序。

选取图 1-2 实例的目的是为了讲解可视化界面工具设计时的一些细节,所以对实例的要求只是功能性的,里面既不涉及复杂的控制流程,也没什么数据元素。这样的好处是使我们的讲解可以只针对界面方面的问题展开,从而加强实例的针对性。但在另一方面,这个简单的实例确实是一个完整的程序,所以当读者遵循下面的演示过程最终完成这个程序的开发时,会有一种很愉快的成就感。这也是本书从界面设计入手的思想出发点。

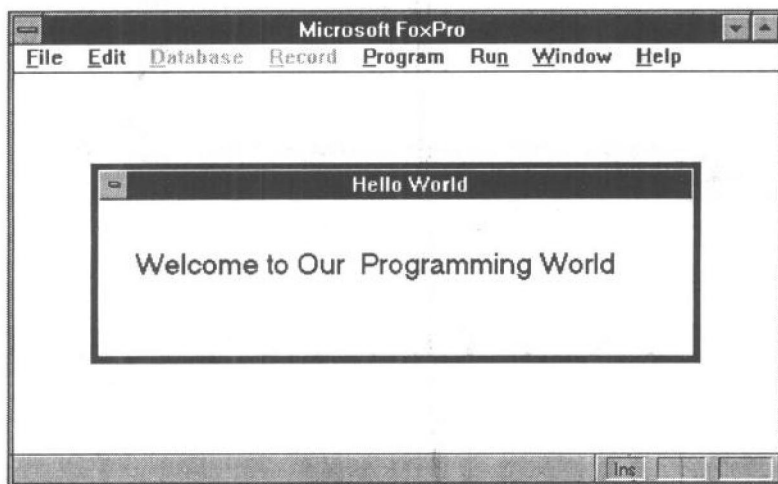


图 1-2 “Hello World”应用程序窗口

实例的最主要部分是设计一个如图 1-2 所示的窗口，然后在其客户区中显示一条欢迎信息。可以按照传统的程序设计模式：完全用 FoxPro 提供的 PRG 语言，编写出一个符合要求的应用程序。但在这里，为了强调 FoxPro 可视化编程的优越性。我们首先演示了工具化设计的模式，在本专题的后半部分，可以看到两种模式的一些对照评价。

界面设计时最主要的工具是屏幕生成器(Screen Builder)。遵循以下三个步骤，能得到一个符合图 1-2 要求的应用程序：

- (1) 打开屏幕设计窗口。
- (2) 利用工具定制界面的窗口以及里面的显示内容。
- (3) 产生 SPR 文件

由于本书不是 FoxPro 的交互使用说明，所以我们不对操作步骤作细致的描述，读者可以参见 FoxPro 的用户手册。

1.2.1 打开屏幕设计窗口

当在命令窗口打入如下命令后：

```
CREATE SCREEN scx\hello
```

FoxPro 就会弹出 Screen Builder 工具；或者打开 File 菜单，选择 New 操作，然后在对话框中选择 Screen 的分支，用 Save as 把它换成 hello 这个名字也能达到相同目的。FoxPro 的主窗口将出现如图 1-3 所示的屏幕设计窗口，窗口中显示了一个缺省的矩形区域(将要产生的应用程序窗口客户区)。

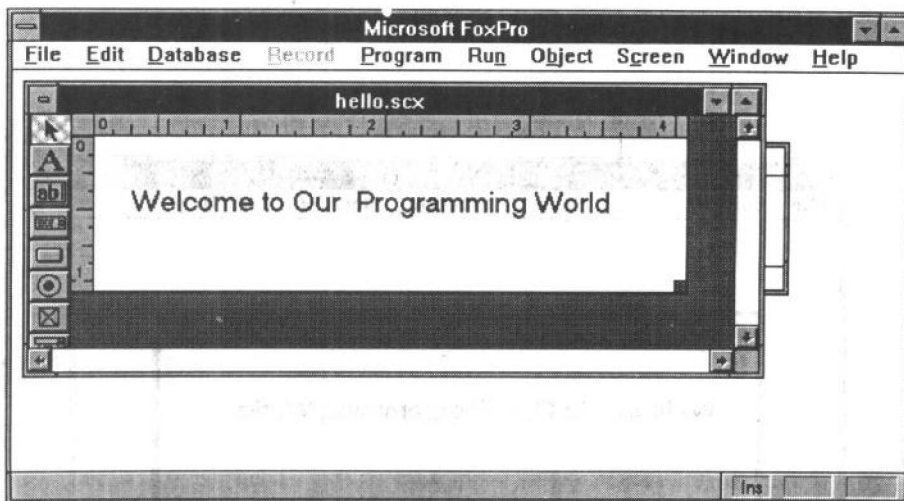


图 1-3 屏幕生成器主窗口

1.2.2 定制界面窗口及显示内容

完成第一步后打开 Screen 菜单,选择里面的布局选项(Layout),这时弹出的 Screen Layout 对话框(图 1-4)就能让程序员完成所需的窗口设计:

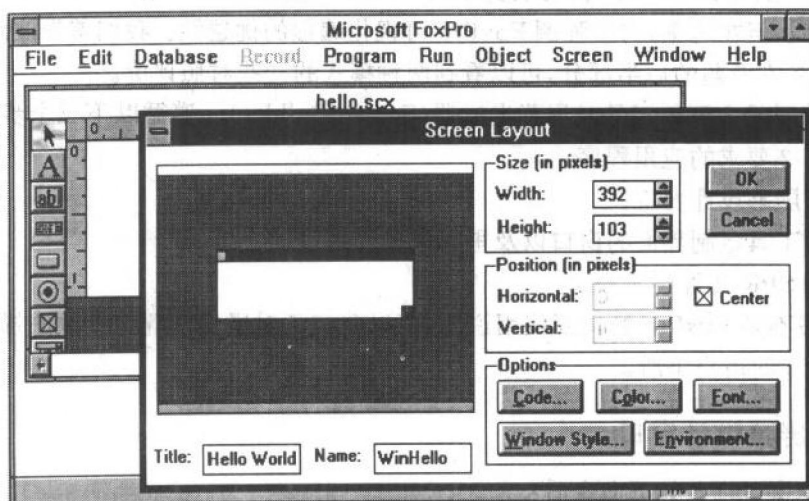


图 1-4 屏幕布局对话框

- (1) 在 Title 的文本区中,输入“Hello World”,它完成了窗口标题的设置。
- (2) 在 Name 的文本区中,输入 WinHello,这是给左窗口取了一个独一无二的名字标识(ID)
- (3) 可以用鼠标拉动角柄;也可以把对话框取消直接拉动设计屏幕上的角柄,再弹出此对话框;或者改变 Size 大小框中的宽(Width)、高(Height)的数码器内容。总之,不

管使用何种方式,均能随意自如地更改窗口的大小,而且这种改变会立即反映在屏幕上,这就是可视化编程的一个特点。随着我们编程经验的增加,可以深深地体会这种可视功能带来的优越性。

- (4) 可以用与上面相似的方式拖动窗口来改变窗口的位置(不能拖动设计屏上的矩形区域),也能表现出可视化的特点。更为简便的方法,是选择位置框(Position)中的中央选项(Center),可以看到窗口能够迅速移到屏幕的中央。
- (5) 按一下选项框(Options)中的窗口风格(Window Style)按钮,在风格(Style)对话框中可以进一步设定窗口的一些属性。如选择 Movable、Close、Minimize 选项,分别设定了产生的窗口具有移动属性、可以被关闭、能缩成最小图符。在此,我们只选择了移动属性。在边框(Border)的单选框(Radio Box)中选择 System,如图 1-5 所示。
- (6) 在图 1-4 中按一下选项框(Options)中的代码(Code)按钮,在代码对话框中需要设置一些 READ 级的命令子句,如图 1-6 所示。这里只把 Valid 表达式设为真。
- (7) 当完成了窗口的各项定制后,选择 OK 按钮来释放所有的对话框。然后,在屏幕设计窗口的左边工具栏里按一下字符按钮图符,在窗口客户区的适当位置输入“欢迎进入 FoxPro 程序设计世界”,如图 1-3 所示。至此,我们已用屏幕生成器初步完成了应用程序的设计工作。

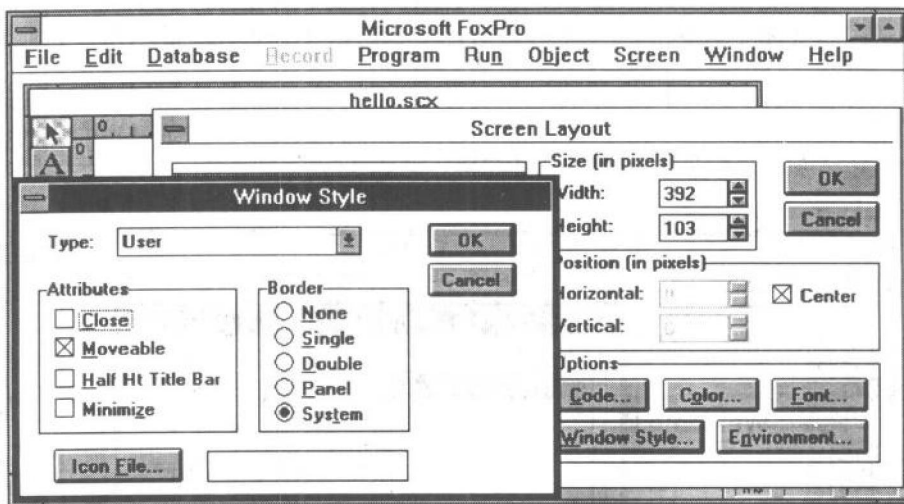


图 1-5 定制窗口风格