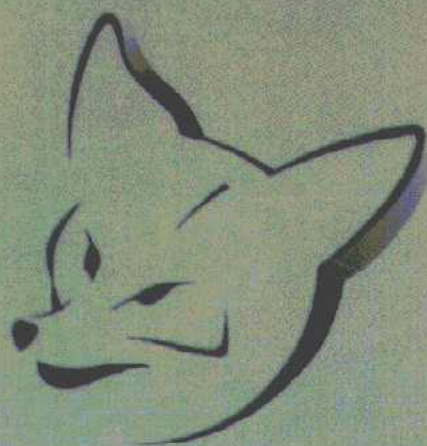


# Visual FoxPro 6.0

## 中文版 专家指南



夏伟文 编著  
熊文亮 审校



清华大学出版社

<http://www.tup.tsinghua.edu.cn>

# Visual FoxPro 6.0 中文版

## 专 家 指 南

夏伟文 编著  
熊文亮 审校

清华大学出版社

(京)新登字 158 号

## 内 容 提 要

本书结合作者新实践,针对 FoxPro 的早期版本,简明扼要地介绍了 Visual FoxPro 6.0 所具有的独特功能。对已经熟悉 FoxPro 早期版本的读者尽快掌握 Visual FoxPro 之精髓具有一定的指导意义。

书 名: Visual FoxPro 6.0 中文版专家指南

作 者: 夏伟文 编著

出版者: 清华大学出版社(北京清华大学学研大厦,邮编 100084)

<http://www.tup.tsinghua.edu.cn>

印刷者: 北京昌平环球印刷厂

发行者: 新华书店总店北京发行所

开 本: 787×1092 1/16 印张: 9.25 字数: 224 千字

版 次: 2001 年 1 月第 1 版 2001 年 1 月第 1 次印刷

书 号: ISBN 7-302-03878-3/TP·2335

印 数: 0001~4000

定 价: 16.00 元

# 前 言

目前计算机技术领域内较为流行的 FoxPro 数据库管理系统已为众多的使用者所接受,并得到各界的关注,因其广泛的应用天地使得 FoxPro 能得到飞速的发展。在不断推出的性能优异更新换代的产品中,Visual FoxPro 就是当前广泛使用、功能较齐全的一个,尤其在 1998 年期间出现的 Visual FoxPro 6.0 中文版(简称为 VFP 6.0)是适应网络技术迅速发展,基于 Internet 平台的,对于网上发布及其查询功能更为适用的一种新的优秀数据库系统。与过去的 FoxPro 相比较,VFP 6.0 除了功能上的增强、简化改进外,较为突出的是增加了不少新的向导(Wizard),例如连接向导和应用程序向导,这无疑带来了更高层次的技术更新和发展。

本书作者为了帮助更多的使用者少走弯路,迅速掌握和使用 VFP 6.0 这个较新的数据库系统,结合自己历年来工作实践的体会,编写了本书。书中除了简明扼要地介绍 Visual FoxPro 6.0 的一些特有性能外,更多地通过实际操作的叙述以帮助读者尽快地熟练使用 Visual FoxPro 6.0 及熟悉其应用环境。其中还就作者本身在实际应用的一些开发使用经验作了阐述,以期对读者起到抛砖引玉的作用,希望有更多的使用者在自我开发的基础上达到新的境地。

本书的结构编排尽量深入浅出,考虑查询阅读方便,可以使不同需要的读者能有的放矢尽快查阅到所急需的内容。相信使用者通过本书的阅读能迅速掌握使用和开发 Visual FoxPro 6.0,这也是作者和出版者的愿望。

在本书的编写过程中,中国人民大学吴微和赵彦云教授、中国电子报社张凤栖主任编辑等给予了热情的帮助和大力的支持,正是他们的认真审读纠正了本书内容中的一些不妥之处。本书最后由中科院微软培训中心的熊文亮先生审校,保证了本书内容的准确。本书的编写过程中也参阅了不少其他专家的有关资料和书籍,使作者本人受益匪浅,在此不一一列举并诚致谢意!由于写作时间紧迫,作者的水平有限,书中难免会有不足之处,敬请专家和读者不吝赐教。

编 者  
(于美国洛杉矶)  
1999 年 12 月

# 第 1 章 Visual FoxPro 概述

## 1.1 Visual FoxPro 发展简史

Visual FoxPro 这一软件技术上的起源可以追溯到 1975 年。一位名叫 C. W. Ratliff 的工程师,当时受雇于 Martin-Marietta 公司,在位于美国加利福尼亚州 Pasadena 市的喷气推进实验室从事一个航空航天项目的研究。Ratliff 先生在业余尝试编写一个类似于英语的解释执行并能与用户交互的数据管理程序。1976 年,他开始在一部 IMSAI 计算机上测试他编好的程序,在这其后的几年中,Ratliff 先生将程序命名为 Vulcan 并卖出 100 份拷贝。直到 1980 年,Ratliff 先生遇到另外三个人 George Tate, Hal Rawluk 和 Hal Lashlee。这三个人负责将产品推向市场,Ratliff 本人则继续从事产品开发工作。于是他们组建了一个名为 Ashton-Tate 的公司并将该产品重新命名为 dBASE II 作为公司主要产品。

dBASE II 最初运行于 CPM 操作系统上,后来随着 DOS 系统代替 CPM 系统,dBASE II 又进入了 IBM PC 市场,成为 PC 机的数据库标准软件包。从 80 年代初到 80 年代中期,dBASE II 一度占领 PC 机数据库市场 70% 的份额。

1987 年 2 月,由大学教授 Dave Fulton 领导的 Fox 软件公司推出与 dBASE III 兼容并在速度上略胜一筹的 FoxBASE 1.0。同年 7 月,Fox 软件公司又推出了 FoxBASE 2.0。

也就是在这一时期,随着 PC 机在中国的迅速普及,dBASE III 和 FoxBASE 2.0 也迅速占据了国内 MIS 开发工具的市场,而 xBASE 语言也成为事实上的标准。与此同时,Ashton-Tate 公司在 1988 年推出的 dBASE IV 由于未能包含用户需要增加的功能且速度比不上 FoxBASE,从而未能在市场上取得成功,这也为后来 Borland 公司最终收购 Ashton-Tate 公司埋下伏笔。

与此相反,Fox 软件公司在稍后推出的 FoxPro 1.0 则进一步确立了其在关系数据库市场上的优势地位。FoxPro 的优点有以下三个方面:

第一是扩充了命令和函数。FoxPro 比 dBASE IV 多出 140 条左右的语句,还扩充了 90 余条 FoxBASE 和 dBASE IV 都没有的命令和函数。

第二是速度上的优势。FoxPro 1.0 比 dBASE III 快 16 倍,比 FoxBASE 快 2 倍。

第三是增加了基于窗口的下拉菜单和对话框。

1991 年,Fox 软件公司推出了另外一个极具震撼力的产品 FoxPro 2.0。该产品包括 Rushmore 数据查询技术,并使一个表可以拥有几百条记录。为了进一步与 Ashton-Tate 公司争夺 xBASE 数据库市场,Fox 软件公司宣布其下一个版本同时支持 4 种平台: DOS, Windows, Macintosh 和 UNIX。FoxPro 2.5 正是由于同时支持这 4 种平台,因而在市场竞争中取得了巨大的优势。

1992 年发生的另外一件重大事件是 Microsoft 购买了 Fox 软件公司。在其后的两年中,是 FoxPro 开发者生意最为兴旺的时期。1994 年 5 月微软公司在中国推出的支持 Windows 和 DOS 平台的 FoxPro 2.5b 中文版,首先提出了 DBCS(双字节中文内核)的概念。然而,由

于国内用户更希望看到全中文界面的 FoxPro,所以微软公司在 1996 年推出 Visual FoxPro 3.0(简称为 VFP 3.0)中文版时引入了全中文的界面,并对文档和例子进行了全面的汉化。

本书将介绍的是微软公司在 1998 年推出的 Visual FoxPro 6.0(简称为 VFP 6.0)中文版。Visual FoxPro 6.0 与 3.0 相比主要有以下几方面的改进:增强的项目及数据库管理、改进的调试工具、增强的查询及视图设计、增加的表单功能及其更简便的设计。此外,还增加了多个新的向导(Wizard),例如连接向导和应用程序向导。Visual FoxPro 3.0 与 FoxPro 2.5b 相比改进就更多了,例如引入了其他 Microsoft 产品中很普遍的向导(Wizard),方便了用户建立各种报表、查询、表格和标签等。另外,VFP 3.0 中还引入了面向对象的编程和数据库容器的概念。本书将在第 2 章中讨论 Visual FoxPro 6.0 技术的更新和发展。

## 1.2 Visual FoxPro 6.0 的安装

### 1. Visual FoxPro 6.0 中文版的软硬件要求

#### (1) 硬件要求

VFP 6.0 要求运行于 CPU 为 486 66MHz 以上的 IBM PC 机上,另外还需要一个鼠标和至少 16M 的内存。VFP 6.0 的最大安装需要 90MB 的硬盘空间。

#### (2) 软件要求

VFP 6.0 要求运行于 Windows 95(中文版)或 Windows NT 4.0(中文版)及其以上版本之上。由于 Windows NT 对系统的开销会更大,所以对内存的要求量也更高。如果用户需要运行“升迁向导”,则还需下列软件:

在服务器端要安装 Microsoft SQL Server 4.x 或更高的版本,在客户端要安装 Visual FoxPro 及 ODBC 组件。如果用户使用 Oracle 数据库,则需在服务器端安装 Oracle Server 6.0,在客户端安装 Oracle 的 ODBC 驱动程序。

### 2. 从 CD-ROM 上安装 Visual FoxPro

首先将 VFP 6.0 的光盘插入 CD-ROM 驱动器,然后从“开始”菜单中选择“运行”。假设 CD-ROM 的盘符为 d,则键入“d:\Setup”并按回车。然后将出现 Visual FoxPro 安装程序的主画面。由于是全中文界面,用户只要按照提示一步步完成就可以了。

用户若选择“最小安装”,只需要 15MB 硬盘空间;若选择“完全安装”,则需要 85MB 硬盘空间;若要安装包括联机文档在内的所有文件,则要 192MB 硬盘空间。

## 第 2 章 Visual Foxpro 6.0 的新增功能

### 2.1 Access 和 Assign 方法程序

利用这两种用户自定义的方法程序可在查询或试图更改属性的值时执行所需的代码。这样您就能够为类或对象创建公共接口,将其界面与执行的操作分离开来,也能够方便地检查属性值的有效性,还可以轻而易举地保护 ActiveX 控件子类的属性。

在查询属性值时,将执行 Access 方法程序中的代码。通常可以通过在一个对象的引用中使用属性、将属性值保存到一个变量中或使用问号(?)显示属性的值。

在试图更改属性值时,将执行 Assign 方法程序中的代码。通常可以通过使用 STORE 或 = 命令为属性赋予一个新值。只有在运行时刻查询或更改属性值时,才执行 Access 和 Assign 方法程序。在设计时刻查询或更改属性值时,不会执行 Access 和 Assign 方法程序。

由于赋予的属性值将传递到 Assign 方法程序,因此在 Assign 方法程序中必须包含 PARAMETERS 或 LPARAMETERS 语句,以便接收该值。

### 2.2 Active Document

Active Document 是一种基于 Windows 的,嵌入在浏览器中的非 HTML 应用程序,它提供了在浏览器界面中访问该应用程序功能的途径。VFP 6.0 的 Active Document 应用程序与任何其他的 VFP 6.0 应用程序一样,都可以运行表单、报表和标签,将类实例化,运行代码及操作数据,而且 Active Document 应用程序还具有自身的优势,即能包容在像 Internet Explorer 这样的 Active Document 容器中。

如下为 Active Document 提供的一些功能:

1. Active document 总在现场激活。
2. 通过 Active Document 菜单和工具栏命令,可以用到 Active Document 宿主程序的一些功能。
3. 使用 Internet Explorer 查看时,Active Document 可以实现同其他网页的无缝集成。
4. 对于将纯 VFP 6.0 客户应用程序迁移到使用基于 HTML 客户界面的 Active Platform 应用程序,Active Document 迈出了革命性的一步。

### 2.3 组件管理库

“组件管理库”是软件对象(例如类库、表单、按钮等)类型的容器,也包含新的 VFP 6.0 类。可使用“组件管理库”将组件分成对象、项目、应用程序或其他类。这种可视分组方式可进行动态定制,这样可以使用、复制或重新组织“组件管理库”中几种不同分类的组件。在“组件管理库”的任何地方,只要放置了对指定数据项的引用,都可以对其进行访问。对于单独的

项目,在不同的目录或文件夹中可以有多多个引用。例如,一个按钮可能会出现在一个或几个“组件管理库”的项目目录中(用文件夹代表),也可能在“工具”目录中出现,因为在“工具”目录中包含了所使用的所有按钮。

## 2.4 代码范围分析器应用程序

代码范围应用程序可记录文件中的哪些代码行被执行了。而分析应用程序则提供了哪些行被真正执行、一行代码被执行的次数、所耗费的时间等信息。代码范围模式和分析模式使开发者能找出应用程序中存在问题的区域,特别是能够找到被跳过的代码以及性能瓶颈。VFP 6.0 的“代码范围分析器”提供了可以使用并进行定制的代码范围引擎,以及一个多窗口应用程序,用于分析程序和项目。

## 2.5 对 GIF 和 JPEG 图形的支持

经过改进的 VFP 6.0 能够支持 GIF (Graphics Interchange Format) 和 JPEG (Joint Photographic Electronic Group) 图形文件格式,以进一步加强对 Internet 互操作性的支持。VFP 6.0 中有三个领域支持图形:语言,控件和对象,以及界面。

### 2.5.1 VFP 6.0 语言

下列命令和函数已得到改进,现支持新的图形文件格式。

1. GETPICT() 通过在“命令”窗口使用 GETPICT() 函数,可以显示“打开图片”对话框,该对话框已得到改进,可快速定位所有 VFP 6.0 支持的图形文件。选中“预览”复选框可显示当前选中的图形文件,而在早期版本的 VFP 6.0 中,在选择新的图形文件时,需要选取“预览”按钮。此外,“图片”区的面积也增加了。

2. CLEAR RESOURCES 使用 VFP 6.0 的 CLEAR RESOURCES 命令可清除所有缓存的图形文件,包括.gif 和.jpg 文件。

### 2.5.2 VFP 6.0 控件和对象

下表列出了可将其属性指定为图形文件 VFP 6.0 的控件和对象。除早期 VFP 6.0 版本所支持的.bmp 和.dib 图形文件外,还可指定.gif、.jpg、光标、动画光标和图标等图形文件。

| 控件或对象              | 属 性             |
|--------------------|-----------------|
| CheckBox 控件(复选框控件) | DisabledPicture |
|                    | DownPicture     |
|                    | Picture         |
| Command Button 容器  | DisabledPicture |
|                    | DownPicture     |
|                    | Picture         |

续表

| 控件或对象                    | 属 性             |
|--------------------------|-----------------|
| Container 对象             | Picture         |
| Control 对象               | Picture         |
| Custom 对象                | Picture         |
| Form 对象                  | Picture         |
| Image 控件(图象控件)           | Picture         |
| OptionButton 控件(选项按钮组控件) | DisabledPicture |
|                          | DownPicture     |
|                          | Picture         |
| Page 对象                  | Picture         |
| Screen 对象                | Picture         |

### 2.5.3 VFP 6.0 界面

使用几种 VFP 6.0 设计器的“打开”对话框可以指定图形文件。下列设计器的“打开”对话框得到改进,可包含新的图形文件格式。

#### 1. 表单设计器和类设计器

在“属性”窗口中双击一个支持图形文件的属性、或选取属性的对话框按钮,可显示“打开”对话框。

#### 2. 项目管理器

在项目管理器的“全部”和“其他”选项卡中,可以向一个项目添加图形文件。选取“全部”或“其他”选项卡后,选取“其他文件”项,然后选取“添加”,将显示“添加”对话框,在此可向项目中添加图形文件。

#### 3. 报表设计器

“报表控件”工具栏包含“图片/ActiveX 绑定控件”按钮。单击该按钮,并将鼠标光标拖到报表设计器的一个带区,将显示“报表图片”对话框。选取“文件”对话框按钮将显示“打开”对话框。

## 2.6 HTML 帮助

HTML Help 是 Microsoft 所提供的用于创建适应 Internet 时代要求的帮助文件的解决方案。Visual Studio 中自带的 HTML Help Workshop,可用于为您的 VFP 6.0 应用程序创建和发布 HTML Help 文件。

## 2.7 新的和改进的语言元素

VFP 6.0 语言已经得到了改进,以简化编程工作。而且,将 VFP 6.0 API 库 Foxtools.fll 中的许多有关文件名操作的函数也添加到了 VFP 6.0 中。

## 2.8 OLE 拖放

VFP 6.0 现在支持 OLE 拖放,这是一个强大而有用的工具,允许您在其他支持 OLE 拖放的应用程序(例如 Visual Basic、Windows Explorer、Microsoft Excel 和 Word)之间移动数据。在一个已发布的 VFP 6.0 应用程序中,您可以在该应用程序的控件之间移动数据,或者在控件和其他支持 OLE 拖放的 Windows 应用程序之间移动数据。

## 2.9 项目管理器挂接程序

在以前版本的 VFP 6.0 中,访问项目的唯一途径是像处理 .dbf 文件一样,处理 .pjx 文件或窗口标题为基础的“项目管理器”窗口。为了能以编程方式访问“项目管理器”,采用了 Project 对象关联模型。该模型由项目、Project 对象和相应的 ProjectHook 对象组成。当通过执行 CREATE PROJECT、MODIFY PROJECT、BUILD APP 等命令打开一个项目时,都将实例化相应的 Project 对象。Project 对象是打开项目和 VFP 6.0 IDE 之间的媒介,并且允许开发者直接与项目交互。Project 对象也将实例化一个 ProjectHook 对象,用于通知发生的事件。

## 2.10 自动服务程序的改进

VFP 6.0 提供了经过改进而更可靠的自动服务程序(Automation Server),这些自动服务程序能更好地与 Microsoft Transaction Server、Microsoft Visual Basic 和 Active Server Pages 这样的产品和技术协作。

## 2.11 新增和改进了的向导和生成器

除了对已有的几个向导进行了改进之外,VFP 6.0 还带有新的向导和生成器,帮助您生成应用程序、创建数据库、在 Web 上发布数据、建立对象模型,以及创建您自己的向导。

### 1. 新的应用程序向导

VFP 6.0 的应用程序向导提供对改进了的“应用程序框架”和新的“应用程序生成器”的支持。从“组件管理库”中、或从 VFP 6.0 的“工具”菜单中单击“向导”,然后单击“应用程序”,即可以运行“应用程序向导”。为了向后兼容,在“向导选取”对话框中有 VFP 5.0 的“应用程序向导(5.0)”。

### 2. 新的连接向导

连接向导包括“代码生成向导”和“反向工程向导”。使用这些向导可以轻松实现 VFP 6.0 类库和 Microsoft Visual Modeler 模型之间的转换。

### 3. 新的数据库向导

VFP 6.0 的数据库向导使用模板创建数据库和表。也可以使用向导创建索引、以及新数据库中表与表之间的关系。

#### 4. 新的示例向导

VFP 6.0 示例向导提供了一些创建自己的向导的简单步骤。其输出为根据指定的数据源中的记录创建的 HTML 文件。

#### 5. 新的 Web 发布向导

VFP 6.0 的 Web 发布向导可以根据指定的数据源中的记录创建一个 HTML 文件。

#### 6. 改进的文档向导

VFP 6.0 的文档向导提供一个选项,以使用“代码分析器”来创建文档。

#### 7. 改进的表单向导

更新了的 VFP 6.0 表单向导为保存在数据库中的特定字段提供输入掩码、格式和一个字段映射类。该向导还包含更多的表单样式选项,包括可滚动表单。

#### 8. 改进的图形向导

VFP 6.0 的图形向导可以使用 Microsoft Graph 从 VFP 6.0 的表来创建图形。更新了的向导支持 Microsoft Office 97 的 Graph 8.0 组件,包括数据表的自动服务和 Series by Row/Col 选项。

#### 9. 改进的导入向导

更新了的 VFP 6.0 导入向导支持 Office 97 和 Microsoft Excel 的多数据表操作,并提供向数据库中导入表的选项。

#### 10. 改进的标签向导

VFP 6.0 的标签向导现在可以更好地控制标签字体,并能直接访问“添加标签向导”。

#### 11. 改进的邮件合并向导

VFP 6.0 邮件合并向导可为 Microsoft Word 合并文档创建数据源,或创建一个其他字处理器能使用的文本文件。更新了的向导支持 Office 97 中的 Microsoft Word 8.0 组件和真正的 VBA 自动服务(具有 Application 对象和集合支持的)。

#### 12. 改进的数据透视表向导

VFP 6.0 数据透视表向导可以帮助创建交互的工作表,用来总结和分析一个表中的两个或更多字段中的数据。更新了的向导支持 Office 97 中的 Microsoft Word 8.0 组件,可以选择在 Excel 中直接保存一个数据透视表、或将其作为一个对象添加到表单中。

#### 13. 改进的报表向导

VFP 6.0 的报表向导包含高级的分组和总结功能,因此可以在单个向导中定制报表。其中也有更多的报表样式可供选择。

#### 14. 改进的远程视图向导

VFP 6.0 的视图向导提供了对 System 表的访问,这样可以使用支持这些表的 ODBC 驱动程序功能。

## 2.12 对 2000 年日期的支持

VFP 6.0 中的一些语言得到了改进,以在应用程序代码中支持 2000 年日期。

## 第3章 升级到 Visual FoxPro 6.0

由于硬件配置的局限,大量国内 FoxPro 用户使用的还是 FoxPro 2.6,应用程序也是基于 2.6 版开发的。随着硬件的更新换代及其性价比的不断提高,为软件平台的升级提供了可能。那么,将 FoxPro 2.6 的应用程序升级到 FoxPro 6.0 是否有必要,又如何办到呢?这是本章将要讨论的内容。

### 3.1 决定升级到 Visual FoxPro 6.0 的判定法则

在决定升级到 Visual FoxPro 6.0 之前,FoxPro 2.6 应用程序的开发者应该问一下自己,自己的应用程序是否在 FoxPro 2.6 中运行得很好?是否需要增加应用程序的功能以及由此带来的时间和费用上的开销有多少?也许在认真思考这些问题之后,你或许宁愿将你的应用程序运行在 FoxPro 2.6 当中。但是一想到 FoxPro 6.0 新的功能与特性,如面向对象编程、数据库增强功能、强大的开发调试环境以及改进的系统性能,许多人还是抵抗不了它的诱惑而将自己的应用程序升级到 Visual FoxPro 6.0 当中。下面我们将就 FoxPro 2.6 的应用程序升级到 Visual FoxPro 6.0 时所应用的方法和原则展开讨论。

#### 3.1.1 升级前的准备工作

##### 1. Visual FoxPro 6.0 与 FoxPro 2.6 的差别对升级带来的影响

###### (1) 操作系统平台的差别

FoxPro 2.6 可运行于 16 位的 Windows 3.2 中文版(以下简称 WIN32)上,或者是西文版的 Windows 3.1 上。而 Visual FoxPro 6.0 只能运行于 32 位的 Windows 95 中文版(以下简称 WIN 95)或者 Windows NT 4.0 中文版(以下简称 WIN NT)上。如果开发者要升级应用程序到 Visual FoxPro 6.0,应用程序的使用者也必须使用 32 位的 WIN 95 或 WIN NT 平台。

###### (2) 代码风格的差别

大多数 FoxPro 2.x 的程序代码风格为传统的面向过程型,这些程序代码的大多数只需略加改动就可以在 Visual FoxPro 6.0 当中执行。

对于 Visual FoxPro 6.0 而言,面向对象的代码风格是面向对象型。因此在转换旧的代码风格到新的面向对象型的时候,开发者要有如下的思想准备:

① 开发者将把宝贵的时间和精力用于开发可重用代码库,而全部的回报将在以后的而不是当前的项目中体现。

② 开发者希望开掘出面向对象程序开发所具有的事件处理、通用代码语法等一系列优点。

③ 应用程序风格的差异。由于 Visual FoxPro 6.0 只能运行于 WIN 95 或 WIN NT,其

应用程序也只有区别于 Windows 3.1 的 Windows 95 的外观和感觉。与此相反, Visual FoxPro 2.6 的应用程序所具有的仍然是 Windows 3.x 的风格。应用程序的使用者是否能接受这种风格上的变化呢?

④ 转换功能上的差异。与 Visual FoxPro 2.6 相比,在 Visual FoxPro 6.0 当中,新增了一系列功能:Active X 控制,事件模型(Event Model),数据缓冲(Data Buffering),OLE 自动化服务器(OLE Automation Server),离线视图(Off-Line Views),外部联结(Outer Joins)等等。从 Visual FoxPro 2.6 升级到 Visual FoxPro 6.0 后,开发者该如何利用这些新的功能呢?开发者务必对此有所准备。

⑤ 专业术语上的差别。在 Visual FoxPro 6.0 当中,Visual FoxPro 2.6 的术语被重新做了定义。下表只是反映了其中重要的变化,详情请参阅 Visual FoxPro 6.0 帮助文件中的词汇表。

| FoxPro 2.6 的术语                  | 对应的 Visual FoxPro 6.0 术语                |
|---------------------------------|-----------------------------------------|
| @...GET 字段                      | 文本框(TextBox)控制                          |
| @...SAY 字段                      | 标签(Label)控制                             |
| 空(Blank),空字段(Empty Field),空变量   | NULL 值                                  |
| 浏览(Browse)                      | 表格控制                                    |
| 代码片段(Code Snippets)             | 属性(Properties),方法(Method)和事件(Event)代码   |
| 数据库(Database)                   | 表(Table)                                |
| 编辑区(Edit Region)                | 编辑框(Edit Box)控制                         |
| 通用(General) 字段                  | OLE 绑定型(OLE Bound)控制                    |
| 弹出(PopUp)                       | 组合框(Combo Box)控制                        |
| 增强的工具(Power Tools)              | 可视化设计器和工具栏(Visual Designer and Toolbar) |
| 按钮(PushButton)                  | 命令按钮(Command Button) 控制                 |
| 单选钮(Radio Button)               | 选项按钮(Option Button)                     |
| 展幕(Screen)                      | 表单(Form)                                |
| 展幕图片(Screen Picture)            | 图象(Image)控制                             |
| 展幕集(Screen Set)                 | 表单集(Form Set)                           |
| 系统内存变量(System Memory Variables) | 系统变量(System Variable)                   |
| 文本(Text)                        | 标签(label)控制                             |

2. 应用程序本身的复杂程度对升级的影响

在升级之前,开发者应对下列问题做到心中有数:

- (1) 应用程序是否需要完成跨平台操作,升级后是否要保留该特点。
- (2) 应用程序是否使用.FLL 文件?开发者是否能有完整的源码和编译环境来重新编译它们?

3.1.2 将 Visual FoxPro 2.6 应用程序升级到 Visual FoxPro 6.0 的三种方法

1. 重新编译法

这种方法所需的工作量最小,基本上不用转换。由于 Visual FoxPro 6.0 具有向下兼容特性,使得 FoxPro 2.6 的程序在 Visual FoxPro 6.0 当中重新编译后也很少出错。但是使用

这种方法 Visual FoxPro 6.0 的面向对象编程的优点就体现不出来了。因此,这只是一种权宜之计。

重新编译法通常采用下列步骤:

- (1) 在 Visual FoxPro 6.0 中创建一个新的项目(Project);
- (2) 将原有的主程序(Main.prg)加入项目;
- (3) 连编(Build)该项目(Project);
- (4) 将项目管理器(Project Manager)未加入的文件手工加入项目;
- (5) 如有需要,加入相关的位图(Bitmap);
- (6) 包含文件(#Includes File)等等;
- (7) 连编(Build)你的应用程序。

## 2. 转换修改法

Visual FoxPro 6.0 自带的转换器(Converter)可以把 Visual FoxPro 2.6 的屏幕集(Screen Set)迁移成向下兼容的 Visual FoxPro 6.0 中的表单集(Form Set)。在屏幕集转换完成后,开发者还需进一步修改使之成为能够实现事件控制等新功能的真正的表单。有关转换器的使用和屏幕到表单的转换,将在本章中进一步阐述。

## 3. 全部重写法

在 3.1.1 节我们提到,应用程序的复杂程度影响到升级的方法。如果 Visual FoxPro 2.6 的应用程序不很复杂,开发者不妨在 Visual FoxPro 6.0 中重新编写一遍。当然,你可以保留 Visual FoxPro 2.6 应用程序里原有的数据结构和事务处理规则。如果你希望充分利用面向对象编程的优点的话,那么按面向对象的程序设计的要求重新调整程序结构或许是不可能的。虽然这种方法费时费力,但这却是三种方法中最有效地发挥 Visual FoxPro 6.0 的功能、最灵活的一种。通过这种方法,你会发现 Visual FoxPro 6.0 新的功能同时会使你的程序代码变得简洁、清晰,你编写类库时投入的时间精力将在以后的程序开发中得到意想不到的回报。尽管如此,花一点时间和精力学习 Visual FoxPro 6.0 的面向对象编程的最佳时机是在新的应用程序的设计阶段。

# 3.2 Visual FoxPro 6.0 自带的转换器

Visual FoxPro 6.0 内置的 Converter 本身也是用 xBase 语言编写的。其源程序安装在 C:\VFP\TOOLS\CONVERT 子目录下,而编译好的 Converter.APP 默认地安装在 Visual FoxPro 的主目录下。Visual FoxPro 6.0 的系统变量 Converter 默认地指向了主目录下的 Converter.APP。如果开发者修改了 Converter 的源程序并重新编译之后,修改 Converter 这个变量可以更换系统原有的转换器。

当用户在 Visual FoxPro 6.0 中试图打开 FoxPro 2.6 的屏幕(Screen)式项目文件时,系统自动探测并调用 Converter.app 并出现了如图 3-1 所示的对话框。

在本对话框最上面的选项是为了方便同时在 Macintosh 和 PC 机上使用 FoxPro 的用户增加的功能。只有用户在转换 FoxPro 2.6 的项目式屏幕文件时才会出现。无须跨两种机型操作的用户可忽略此项。

本对话框的第二个是为了将原文件做备份,并保存在复选框以右边文本框所指路径下

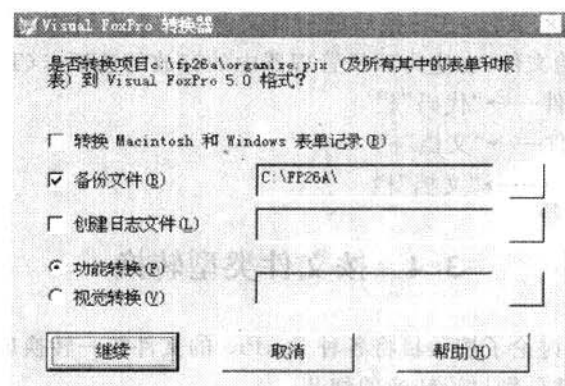


图 3-1 FoxPro 转换器

以 OLD 为名的子目录当中。但是原有的文件目录结构将被破坏,所有文件都将直接保存在 OLD 目录下。因此,读者最好在备份过的 FoxPro 2.6 文件目录上进行转换而保留原有的目录结构。

本对话框的第三个复选框是为了保存操作日志文件到指定的路径。

### 3.3 按项目统一转换

对于大多数应用程序而言,按项目(Project)转换 FoxPro 2.6 的文件是一件省力省心的事。Convert 会一次将项目中的所有文件包括屏幕、报表等自动转换完毕,同时,这也是将屏幕集转换成表单的唯一方法。

本对话框最下面的两个单选框提供了转换 FoxPro 2.6 文件时所用到的两种方法。

第一种方法“功能性转换”。这种转换方法是为了让用户更充分地利用 VFP 6.0 的新的事件模型等特性而设计的。在这种转换方式下,FoxPro 2.6 原有屏幕文件中的代码片段(Snippet Code)将映射到 VFP 6.0 表单文件中的对象属性、事件和方法程序中。但转换完成后,开发者需要进一步对表单中的方法、属性以及表单集的 WindowsType 做相应的调整。有关细节将在以后的章节中讨论。

第二种方法“可视性转换”。与第一种转换方式相反,这种转换方式不会提供直接的 FoxPro 2.6 READ 兼容性。

所有的代码片段(如 OPEN、VALID 等等)仅仅复制为不可编译的 .PRG 文件。用户只需将原有过程代码剪下并粘贴到新的对象属性中。这种方法不需进一步调整表单属性。但同时也无法充分利用 VFP 6.0 新增的特性。

功能性或者是可视性方法对所有的屏幕文件统一转换。你不能对某些屏幕做可视性转换而对其他方法做功能性转换。

如果你选择了“功能性转换”,Converter 将自动完成下列步骤:

- (1) 将原有的 FoxPro 2.6 的文件复制到新建的 OLD 子目录下。
- (2) 将 FoxPro 2.6 的项目文件中所包含的屏幕、报表和标签(Label)文件统一转换成 VFP 6.0 的格式。

(3) 对于不同种类的文件,在转换后的项目文件里放在不同的位置上。下面简要阐述其对应关系。左边为原始文件,右边为项目管理器中的存放的选项卡(Tab)名。

.PRG 和.SPR 文件——→“代码”栏

.SCX 和.SCT 文件——→“文档”栏

.FRX 和.LBX 文件——→“文档”栏

### 3.4 按文件类型转换

在 3.3 节中,我们讨论了按项目将各种 FoxPro 的文件统一转换成 VFP 6.0 格式。下面我们就各种类型的文件分别讨论转换的细节。

#### 3.4.1 将屏幕文件转换成表单

如果用户选择“可视化转换”,则须将原有的代码片段手工添加至表单中相关的方法和程序中,有关细节请参阅联机文档(Online Document)和帮助文件的有关内容。

##### 1. 转换 FoxPro 2.6 屏幕文件的步骤

(1) 当用户在 VFP 6.0 当中使用下列命令打开一个 FoxPro 2.6 屏幕时,如图 3-1 所示的转换器自动出现。

MODIFY COMMAND<FoxPro 2.6 屏幕文件名>

假设用户此时选择“功能性转换”,则转换器自动将原有的屏幕文件(.SCX 和.SCT)(扩展名)更名为.S2X 和.S2T,同时生成转换后的表单和表单集。

(2) 在转换屏幕文件,原有的屏幕事件代码被映射到 VFP 6.0 的表单属性和事件中。映射关系如下表所示。表中 Form 为表单,右栏中 FormSet 为表单集。

| FoxPro 2.6 的屏幕事件        | VFP 6.0 的表单事件                  |
|-------------------------|--------------------------------|
| Setup(初始化代码)-#Section 1 | Formset. Load                  |
| Setup(初始化代码)-#Section 2 | Form. Load                     |
| Read Active             | Form. Active 和 Formset. Active |
| Read Deactive           | Formset. Deactive              |
| Read Show               | Formset. ReadShow              |
| Read When               | Formset. ReadWhen              |
| Read Valid              | Formset. ReadValid             |
| Cleanup                 | Formset. Unload 和过程式函数的 SPR 文件 |
| Read Cycle              | Formset. ReadCycle             |
| Read Lock               | Formset. ReadLock              |
| Read Mouse              | Formset. ReadMouse             |
| Read Save               | Formset. ReadSave              |
| Read Timeout            | Formset. ReadTimeout           |

(3) 设置 WindowType 属性。由于用户选择了功能转换,表单的 WindowType 属性被设为 2 或 3,并且表单集保持了 FoxPro2.6 屏幕集的 READ,从而保证兼容性。这是可视性转换所无法做到的。

(4) 为每一个进行转换的屏幕生成一个以 .SPR 为后缀的文件。这种 SPR 文件与 FoxPro 2.6 的 .SPR 文件不同,它包含一个 DO FORM 命令以启动转换后的 .SCX 文件。除此之外还指定一个可选的包含文件。

(5) 设置其他有关的表单及控制属性。

## 2. 转换过程中对应关系

在转换过程中,FoxPro 2.6 原有的屏幕集(Screen SET)、屏幕(Screen)、READ 级(LEVEL)、READ 屏幕模(Screen MODLE)和 READ MODEL、屏幕模式(Screen MODLE)分别对应于下列的 VFP 6.0 表单特性:表单集、表单集中的表单,面页,WindowType 属性=2 和 WindowType 属性=3。

下表列出了 FoxPro 2.6 的屏幕集代码片段与 VFP 6.0 中的表单属性和事件的对应关系。表中的 FORMSET 为屏幕集。

| FoxPro 2.6 的属性代码                  | VFP 6.0 的表单属性和事件             |
|-----------------------------------|------------------------------|
| 打开文件(Open Files)                  | 关闭文件(Close Files)            |
| 窗口名(Window Name)                  | 窗口标题(Window Title)           |
| 宽度(Width)                         | 高度(Height)                   |
| 左边界(Left)                         | 顶部(Top)                      |
| 中间位置(Center)                      | 边界(border)                   |
| 可移动(Movable)                      | 关闭(Close)                    |
| 图标(Icon)                          | 半高标题棒(Half Height Title Bar) |
| 颜色(Color)                         | 墙纸(Wallpaper)                |
| 最小化(Minimize)                     | 字体(Font)                     |
| 环境(Environment)                   | FormSet. AutoOpenDE          |
| FormSet. AutoCloseDE              | FormSet. 窗口名. Name           |
| FormSet. 窗口名. Caption             | FormSet. 窗口名. Width          |
| FormSet. 窗口名. Height              | FormSet. 窗口名. Left           |
| FormSet. 窗口名. Top                 | FormSet. 窗口名. Center         |
| FormSet. 窗口名. border              | FormSet. 窗口名. Movable        |
| FormSet. 窗口名. Movable             | FormSet. 窗口名. Close          |
| FormSet. 窗口名. Half Height Caption | FormSet. 窗口名. Color          |
| FormSet. 窗口名. Wallpaper           | FormSet. 窗口名. Minimize       |
| FormSet. 窗口名. Font                | FormSet. 窗口名. Environment    |

## 3. 在转换屏幕文件过程需要注意的问题

当用户使用 VFP 6.0 的转换器自动转换完屏幕文件后,并不意味着转换工作的结束。用户还要检查以下几点:

(1) 在 FoxPro 2.6 的 SETUP 代码中,你可以用下列语句以避免创建窗口。

```
IF <条件表达式>
RETURN
ENDIF
```

而在 VFP 6.0 中,你必须用 RETURN .F. 来达到相同的效果。

```
IF <条件表达式>
```