

Windows 应用软件的安装程序

廖雷 (南京电力高等专科学校)

摘要:安装程序有利于 Windows 应用软件的安装。本文在分析了 Visual Basic 3.0 专业版提供的安装程序的样本程序的基础上,总结出了定制安装程序的一般方法,它具有通用性好、实现方便、支持汉字的特点。

Windows 下的应用软件可能包括多个文件,如可执行文件、数据文件、动态连接库、位图文件、图标文件等等,因此安装应用软件时最好采用安装程序,此安装程序的外观和功能类似于 Microsoft 公司的 Word、Excel、Visual Basic、Visual C++ 等工具软件的安装程序。利用此安装程序,用户可以决定应用软件安装到哪个目录,而安装程序会在安装过程中提示用户插入某个特定编号的磁盘,用状态条反映当前的安装进度,最后生成相应的程序组和程序项。若在安装过程中出现错误,则会给出错误信息,必要时,中止安装进程。

生成安装程序主要有两种方法,一种是用 Visual Basic 3.0 下的 Setup Wizard,它是 VB 程序组中的一个程序项,利用它只需在几个对话框中回答几个有关问题即可生成安装程序。但它只能对用 VB 书写的软件生成安装程序,加上缺乏部分安装功能和在安装过程中的提示信息均为英文的弱点,故有一定的

局限性。另一种方法是用高级语言书写安装程序,其中一种选择是用 C/C++,但在 Windows 下用 C/C++ 开发程序难度较大,会使许多人退而却步。另一种选择是用被大家普遍接受的 Visual Basic,当然编写安装程序这一工作决定了开发者需对 Windows 底层有较深入的了解,因此,为了给用户提供方便,在 Visual Basic 3.0 专业版的 Setupkit 子目录下,以 VB 源代码方式给出了编写安装程序所需的函数和安装程序的样本程序,用户可以在此基础上,定制自己的安装程序。本文试图对 Setupkit 中提供的函数及其样本程序作出分析,从而总结出定制安装程序的一般方法。

在 Setupkit 中有一个名为 Setup.exe 的 Windows 可执行程序,该程序执行时,寻找一个名为 Setup.lst 的正文描述文件,该文件一般具有如下内容:

```
APPSETUP.EXE
SETUPKIT.DLL
VBRUN300.DLL
```

第一行的 APPSETUP.EXE 说明用户定制的安装程序名为 APPSETUP.EXE,第二行的 SETUPKIT.DLL 是 SETUP.EXE 所需的动态连接库,第三行的 VBRUN300.DLL 是用 Visual Basic 3.0 书写的可执行程序 APPSETUP.EXE 所需的动态连接库。

用户定制的 APPSETUP 程序是通过 Setupkit 提供的用 VB 书写的工程文件 SETUP1.MAK 进行改写得到的,SETUP1

由以下几个窗体和对话框组成: Setup1: 主窗口, PathDlg: 决定安装路径的对话框, StatusDlg: 以状态条显示当前安装进程的对话框, MessageDlg: 显示正在创建程序组和程序项的对话框, 主窗口出现时, 产生了 Form-Load 事件, 该事件过程的流程图如下:

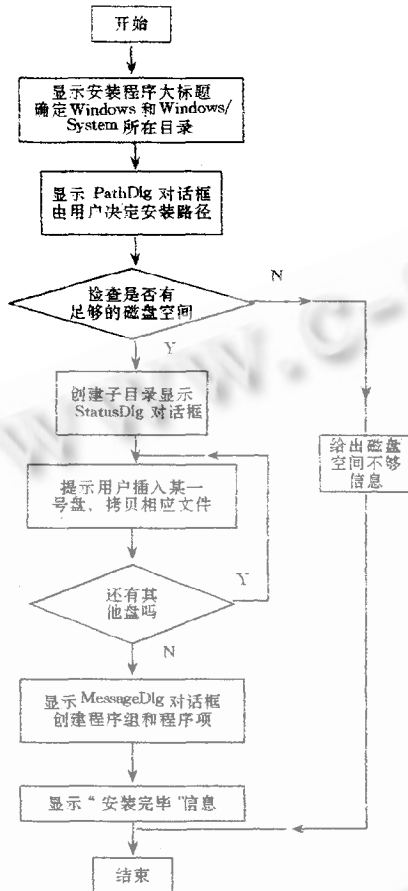


图 1

弄清上述过程后, 用户就可以根据需要进行定制自己的安装程序, 具体工作如下:

1. 改变 SETUP1.FRM 中的某些常量, 具体有:

APPNAME: 在安装过程中显示的大标题。

APPDIR: 用户安装应用程序的缺省目录。

WINSYSNEEDED: 用户文件在 WINDOWS 和 [WINDOWS\SYSTEM] 目录中所需的空间, 如动态连接库一般放在 [WINDOWS\SYSTEM] 子目录下。

OTHERNEEDED: 用户在非系统目录中所需的空间。

笔者发现, 检测磁盘空间是否足够大和以状态条显示安装进度均需利用上述两个参数。

2. 根据实际需要改变调用 PromptForNextDisk 函数时的实

在参数, 该函数的一般形式是:

```
Function PromptForNextDisk ( DiskNum%, FileToLookFor $ ) As Integer
```

该函数通过一消息框要求用户插入某一号盘片, DiskNum% 指出所插的盘号, 而检测插入盘是否为某一号盘的方法是检测该盘上是否有 FileToLookFor \$ 对应的文件。

3. 调用 CopyFile 复制文件, 其一般形式是:

```
Function CopyFile (ByVal SourcePath $, ByVal DestPath $, ByVal SourceFileName $, ByVal DestFileName $) As Integer
```

该函数可以对用 COMPRESS.EXE 压缩的程序解压缩。

4. 有些定制的安装程序, 需要有一些选择功能, SETUP1.MAK 中无此功能, 此时, 用户必须自行定义 OptionDlg 对话框, 再加入特定的代码, 完成相应的功能。

5. 将原先的英文提示信息改为中文。

从上可以看出, VB3.0 专业版中的 Setupkit 仅提供了一个样板程序, 用户需根据需要对之进行修改和扩充, 这就建立在理解和掌握 Setupkit 所提供的一些通用函数基础上, 现将其中函数罗列如下:

```
Sub CenterForm(X As Form)
```

置指定窗口于屏幕中央。

```
Function CreatePath(ByVal DestPath $) As Integer
```

产生指定的路径, 该路径可以有级深度。

```
Sub CreateProgManGroup(X As Form, ProgGroupName $, ProgGroupPath $)
```

通过 DDE 协议向程序管理器发出请求, 建立一个新的程序组。

```
Sub CreateProgManItem(X As Form, CommandLineInfo $, IconTitle $)
```

通过 DDE 协议向程序管理器发出请求, 建立一个新的程序项。

```
Function FileExists(FilePthName $) As Integer
```

检测一个特定文件是否存在。

```
Function GetDiskSpaceFree(Drive $) As Long
```

检测某一磁盘的可用空间。

```
Function GetFileSize(FileName $) As Long
```

检测文件大小。Function GetWindowDir() As String

确定 WINDOWS 所在目录。

```
Function GetWindowSysDir() As String
```

确定 [WINDOWS\SYSTEM] 所在目录。

```
Function IsValidPath(Path $, ByVal DefaultDrive $) As Integer
```

检查用户所给的路径是否合法。

```
Sub UpdateStatus(FileBytes) As Long
```

以状态条方式反映当前安装进度。

采用上述方法定制的安装程序, 具有通用性好、实现方便、支持汉字处理等特点, 使用的特点, 使用效果令人满意。