

利用 Windows API 实现 ODBC 的自动设置

李琳 (军事医学科学院情报研究所信息技术研究室 100850)

摘要: 本文针对利用 VB 提供的 Package & Deployment 向导创建的应用程序无法自动设置 ODBC 数据源的问题,提出了三种解决方案。并对利用 Windows API 修改 VB 自带安装程序 Setup1.exe,实现程序安装过程中自动设置 ODBC 数据源进行了详细介绍。

关键词: Visual Basic Windows API ODBC 安装程序

1 VB 自带打包向导存在的问题

利用 VB 打包向导制作的安装程序,一个最大的缺陷是无法在安装时自动设置 ODBC 数据源,只能在“ODBC 数据源管理器”中手工添加。这项操作对于比较熟悉计算机的用户来说并不难,但是对大多数应用程序的使用者来说,就比较复杂了,即使在操作手册中进行了详细的说明,仍会有许多用户不能接受。此外,越来越多的开发人员都采用客户机/服务器模型作为其应用程序的基础,这就必然要求在客户端添加一个 ODBC 数据源。

2 解决方法

笔者认为有三种方法可以实现这个操作,最好的解决方法是在安装应用程序的过程中自动添加 ODBC 数据源。

方法一: 放弃使用 Visual Basic 提供的 Package & Deployment 向导,转而利用一些专门用于制作安装程序的工具,如: Setup Factory 等。但这类工具一般需要购买,对于非商业化运作的软件开发项目来说也是一笔不小的开销。

方法二: 是最简单易行的,即通过直接修改注册表,在“ODBC 数据源管理器”中添加“用户 DSN”。打开“HKEY_CURRENT_USER \ Software \ ODBC \ ODBC.INI”键,添加一个“项”,并在这个项下添加所需的“字符串”,然后导出相应的注册表文件,并将该注册表文件随应用程序发行,在用户安装应用程序后运行该注册表文件(*.reg)即可。此种方法虽然对于开发人员来说比较简便,但建立的 ODBC 数据源所指定的服务器名

称被限定,对于名称不同的服务器,只能再手工修改,不利于程序发布。

方法三: 利用 Windows API 对 VB 的安装程序包进行修改,实现在安装应用程序时,根据指定服务器自动添加相应的 ODBC 数据源,从而,既简化了程序的安装操作,又节省了购置打包工具的开销。

3 利用 Windows API 修改 Setup1.exe

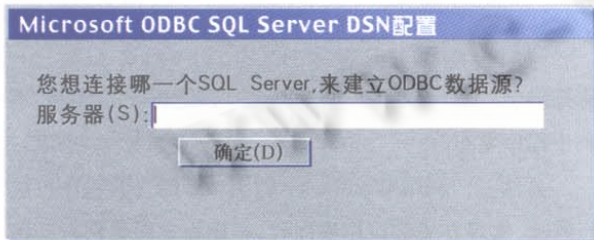
笔者通过在动态连接库(DLL)中调用 Windows API 过程,来实现与注册表的交互。在 Visual Basic 中使用 DLL 过程分为两步:先要声明,然后可以根据需要多次调用。在程序中主要用到三个 API 过程: RegCreateKeyEx、RegSetValueEx、RegCloseKey。下面是详尽的声明语句,当然在实际的程序中,除了要包括 API 所需的声明之外,还应包括常量和类型的声明,这里为了节省篇幅,就不再包括常量和类型的声明了。

```
Public Declare Sub RegCreateKeyEx Lib "advapi32.dll"
Alias "RegCreateKeyExA" (ByVal hKey As Long, ByVal
lpSubKey As String, ByVal Reserved As Long, ByVal lpClass
As String, ByVal dwOptions As Long, ByVal samDesired As
Long, lpSecurityAttributes As SECURITY_ATTRIBUTES,
phkResult As Long, lpdwDisposition As Long) Public Declare
Sub RegSetValueEx Lib "advapi32.dll" Alias
"RegSetValueExA" (ByVal hKey As Long, ByVal
lpValueName As String, ByVal Reserved As Long, ByVal
dwType As Long, ByVal lpData As Any, ByVal cbData As
```

Long)

```
Public Declare Sub RegCloseKey Lib "advapi32.dll"
(ByVal hKey As Long)
```

声明之后,就可以使用这些过程了。RegCreateKeyEx 用于生成指定键,如果该键已经存在于注册表中,则打开这个键。然后用 RegSetValueEx 设置子键的值。最后,用 RegCloseKey 过程释放指定键的句柄。对于客户端安装程序来说,需要用户在安装时确定数据服务器的名称,因此这里提供一个服务器名称的输入窗体 FormOdbcSet.FRM (如图),用于向程序传递建立 ODBC 数据源所需的参数,点击“确定”,开始建立 ODBC 数据源。当然,程序员可以根据实际情况增加建立 ODBC 数据源所需的参数设置。程序具体实现如下:



```
Private Sub cmdOK_Click()
```

```
On Error GoTo ErrorRoutineErr
Dim phkResult1, phkResult2 As Long
Dim SA As SECURITY_ATTRIBUTES
Dim subkey1, subkey2 As String
subkey1 = "Software \ ODBC \ ODBC.INI \ Odbc DB"
subkey2 = "Software \ ODBC \ ODBC.INI \ ODBC
Data Sources"
Dim strDescription, strDriver, strLastUser, strServer,
strOdbc As String
strDescription = "SQL Server ODBCDB"
strDriver = "C:\WINDOWS\SYSTEM\sqsr32.dll"
strLastUser = "UserName"
strServer = txtSQLserver.Text
strOdbc = "SQL Server"
'生成新键
RegCreateKeyEx HKEY_CURRENT_USER, subkey1,
0, "", _
REG_OPTION_NON_VOLATILE, _
KEY_ALL_ACCESS, SA, phkResult1, Create
'附值
RegSetValueEx phkResult1, "Description", 0, _
```

```
REG_SZ, strDescription, CLng(Len(strDescription) + 1)
RegSetValueEx phkResult1, "Driver", 0, _
REG_SZ, strDriver, CLng(Len(strDriver) + 1)
RegSetValueEx phkResult1, "LastUser", 0, _
REG_SZ, strLastUser, CLng(Len(strLastUser) + 1)
RegSetValueEx phkResult1, "Server", 0, _
REG_SZ, strServer, CLng(Len(strServer) + 1)
RegCloseKey phkResult1
'在 ODBC 列表中加入新建键值
RegCreateKeyEx HKEY_CURRENT_USER, subkey2,
0, "", _
REG_OPTION_NON_VOLATILE, _
KEY_ALL_ACCESS, SA, phkResult2, Create
RegSetValueEx phkResult2, "Odbc Shebei", 0, _
REG_SZ, strOdbc, CLng(Len(strOdbc) + 1)
RegCloseKey phkResult2
MsgBox "用户 DSN 建立成功! "
Unload Me
Exit Sub
ErrorRoutineErr:
MsgBox "ERROR #" & str$(Err) & " : " & Error & Chr(13)
_
& "Please try again."
End Sub
```

下面的任务,就是要把上面的程序添加到 VB 打包程序中。由 VB 打包向导创建的安装磁盘或硬盘文件夹中包含有把应用程序安装到用户系统中的所有需要的文件,其中必须包括 Setup.exe、Setup1.exe、Setup.lst。打包程序将文件 Setup.exe 和 Setup1.exe 从 Visual Basic 文件夹中复制到安装磁盘或硬盘文件夹,并创建了文件 Setup.lst。在文件 Setup.lst 中包含了应用程序所需文件的列表,以及一些通用信息。

用户运行 Setup.exe 来安装应用程序,而 Setup.exe 再通过执行列于 Setup.lst 中的主安装程序 Setup1.exe,实现复制和注册文件,以及创建启动图标等操作。因此,只需在 Setup1.exe 中加入对 ODBC 的设置,即可实现在安装过程中设置 ODBC 数据源的目的。Setup1.exe 的源程序 SETUP1.VBP 存放在 \VB98\Wizards\PDWizard\Setup1 文件夹中。打开工程 Setup1.vbp,选择主菜单“工程”的下拉菜单“添加窗体”,在弹出的“添加窗体”窗(下转第 47 页)

(上接第 44 页)

口中选择“现存”，添加前面所建的 FormOdbcSet.FRM 文件，并在 FrmSetup1.frm 文件的 Form_Load 事件中添加对该 FROM 的调用。SETUP1.VBP 工程正是通过 FrmSetup1.frm 的 Form_Load 事件，来实现程序安装过程中对各窗体的逐一调用。为了确保正确安装应用程序后再设置 ODBC 数据源，对窗体 FormOdbcSet 的调用语句应加在 Form_Load 的最后。最后，对工程 SETUP1.VBP 进行重新编译，将新创建的可执行文件 Setup1.exe 拷贝到 \VB98 \ Wizards \ PDWizard 目录下，替换原有的 Setup1.exe，即可。(注意确保对 Setup1.vbp 的最初版本进行保存。)

此时，再利用 Visual Basic 打包向导制作一个新的安装程序。在执行该安装过程的最后，会弹出一个窗口，提示用户输入数据服务器的名称，正确输入后，安装程序就

会自动在用户的“ODBC 数据源管理器”中添加一个用户 DSN，从而实现应用程序 ODBC 数据源的自动添加。

4 小 结

利用 Visual Basic 与 Windows API 相结合的方式，可以解决许多我们在编程过程中遇到的棘手问题；利用对工程 SETUP1.VBP 的修改，可以在 VB 应用程序安装过程中增加功能。本文正是将这两方面进行了良好的结合。

本文中程序在简体中文 Windows98、VB6.0 环境中调试通过。■

参考文献

- 1 《Visual Basic 6 高级开发指南》，Evangelos Petroustos, Kevin Hough, 电子工业出版社, 1999 年 7 月
- 2 《Microsoft Visual Basic 5.0 编程》，微软公司, 宇航出版社, 1998 年 10 月