

Visual foxpro 程序中文档与应用程序关联的实现

周灿林 (山东工业大学数理系 250061)

摘要:本文介绍 visual foxpro 程序中文档与应用程序关联的实现原理,给出了具体实例,从而揭示了 windows 95 文件管理器中关联概念的实质。

关键词: Registry Visual foxpro 注册表编辑器 应用程序

1. 问题提出

如果你对 windows 95 文件管理器使用很熟的话,一定知道:当打开文件管理器后,用鼠标点中某个文档文件,然后双击鼠标左键,可能会启动浏览该文档文件的应用程序。也可能会给出一个警告信息:无法运行程序,请选择文件菜单中的关联命令来创建一个关联。windows 95 文件管理器是如何实现关联的?它在实现关联时到底做了些什么?如果要在自己编制的程序中实现类似功能该如何做?问题由此提出。

2. windows 95 注册表

在 windows 95 中有 Registry(注册表),它是 windows 95 操作系统及其应用程序的配置和性能等信息的存储室。Registry 由 user.dat 和 system.dat 两个文件组成,当 windows 95 启动时,regedit 读取它们并形成一个简单的 registry,这样通过把 registry 分为用户数据和系统数据,使系统有足够的灵活性支持不同硬件的不同用户。要浏览 Registry,可运行 regedit.exe,此时显示对话框。它的左边是 registry 的各种关键字,右边是选中关键字的键值。Registry 的关键字是树状结构。其中 HKEY_CLASSES_ROOT 为文件扩展名与应用程序的关联注册信息。

3. 注册表编辑器的使用

regedit.exe 就是注册表编辑器程序,利用它用户可以查看和编辑注册表。

我们可利用注册表编辑器实现 bmp 文档与 acdsee 32(一个常用的看图应用程序)的关联,按下列主键路径展开相关子键"HKEY_CLASSES_ROOT\ .BMP",可看到键值为"默认 未设置键值",可用鼠标点中"默认",选"编辑"选项中的"修改",这时弹出一新界面,可输入要关联的应用程序命名"acdc_bmp".按下列主键路径展开相关子键"HKEY_CLASSES_ROOT\ acdc_bmp\ shell\ open\ command",可看到键值为"默认" "c:\ acdsee32

\ acdsee32.exe"/v"。这表明 bmp 文件与 acdsee 32 程序已建立关联。windows 95 文件管理器实现关联,实际上也就是往注册表中写入以上信息。当然关联建好后,关联的应用程序必须存在,才能打开 bmp 图象文件。

4. visual foxpro 程序中文档与应用程序关联的实现

在 visual foxpro 程序中实现文档与应用程序的关联,实际上就是以程序的方式完成刚才用注册表编辑器或文件管理器所完成的对注册表所做的修改。

vfp 程序可以通过调用 windows api 函数访问 registry 的任何部分。下面简单介绍这些函数的使用方法。

(1)RegCreateKey 函数用于创建或打开一指定的关键字,返回其句柄,Visual Foxpro 下对其引用前应首先作如下方式声明:

```
DECLARE Integer RegCreateKey IN Win32API ;
    Integer nHKey, String @cSubKey, Integer @nResult
```

(2)RegSetValueEx。函数用于设置关键字的值,Visual Foxpro 下对其引用前应首先作如下方式声明:

```
DECLARE Integer RegSetValueEx IN Win32API ;
    Integer nHKey, String cValueName, Integer nReserved, ;
```

```
Integer nType, String cBuffer, Integer nBufferSize
```

利用以上两个函数,可以设置 bmp 文档与 acdsee 32 应用程序的关联。

其他象 txt 文档与 notepad.exe 应用程序的关联等类似情况的处理,处理方法相似,限于篇幅在此不再说明。

附程序代码(本程序在联想 p5/166pci、windows 95、visual foxpro 5.0、安装了 acdsee 32 应用程序的机器上调试通过)如下:

```
#DEFINE REG_SZ 1 && Data string
#DEFINE REG_BINARY 3 && Binary data in any
```

```

form.
# DEFINE REG_ DWORD 4 && A 32 - bit number.
# DEFINE HKEY _ CLASSES _ ROOT - 2147483648
&& BITSET(0,31)
# DEFINE HKEY _ CURRENT _ USER - 2147483647
&& BITSET(0,31) + 1
# DEFINE HKEY _ LOCAL _ MACHINE - 2147483646
&& BITSET(0,31) + 2
# DEFINE HKEY _ USERS - 2147483645 && BITSET
(0,31) + 3
    DECLARE Integer RegCreateKey IN Win32API ;
        Integer nHKey, String @cSubKey, Integer @nRe-
result
    DECLARE Integer RegCloseKey IN Win32API ;
        Integer nHKey
    DECLARE Integer RegSetValueEx IN Win32API ;

```

```

Integer hKey, String lpszValueName, Integer
dwReserved, ;
        Integer fdwType, String lpbData, Integer cbData
nResult = 0
cSubKey = ". bmp"
nerrorcode1 = RegCreateKey ( HKEY _ CLASSES _
ROOT , @cSubKey, @nResult)
cvalue = ""
cvalue = "acdc _ bmp" + chr(0)
cvaluesize = len(cvalue)
IF nerrorcode1 = ERROR _ SUCCESS
        nerrorcode2 = RegSetValueEx(nResult, cvalue,
0, REG _ SZ, m. cvalue, m. cvaluesize)
ENDIF
= RegCloseKey(nResult)

```

(来稿时间:1999年5月)