

Windows 下中西文输入法的自动切换

中国人民银行许昌分行 张富民 崔华锋 张遂昌



问题的提出

使用过 Visual ForPro 的用户都知道,在向其表单中添加记录时,随着当前输入字段的滚动,所采用的输入法能够自动切换。当新的输入字段的数据类型为字符型时,输入法自动切换为中文输入法,而如果是其他类型时又切换为西文输入法。

在某信息系统的开发中,我们仿效 VFP 的做法成功地在该系统的输入模块中实现了中西文输入法的自动切换,从而避免了用户使用过程中频繁的通过手工切换之烦恼,提高了工作效率。鉴于这类问题具有一定的普遍性,在此特将我们在解决这一问题时所采用的方法作一介绍,希望能对大家有所帮助。

注册表与输入法

注册表(Registry)是 Windows 9X/NT 操作系统的信息存储中心,存放着包括计算机硬件配置,已安装软件的设置信息,当前用户的系统环境设置以及某些特定类型文件与对其进行访问和操作的应用程序之间的联系等重要信息。Windows 操作系统的早期版本(Windows 3.X)中存放在初始化文件(*.INI)中的许多信息现在都存放在注册表中。

注册表采用了类似于磁盘文件目录结构的树型结构形式,利用 Windows 所带的注册表编辑器(RegEdit.Exe)或其他工具软件(如 Wedge Software 公司的 WinHacker95, Symantec 公司的 Norton Registry Editor 等等),用户可以方便的对注册表进行全方位的管理和维护。

注册表中有两个地方存放有关于输入法的信息,下面分别予以介绍。

1. Keyboard Layouts 子键

该子键的位置是“我的电脑/HKEY_LOCAL_MACHINE/System/CurrentControlSet/Control/Keyboard Layouts”,主要是存放 Windows 的键盘存局(Keyboard Layouts)及键盘语言的有关设置信息。该子键下

面包括若干个用 8 位十六进制数值表示的子键,分别对应于一种键盘布局或键盘语言,包括中文输入法。常用的几种输入法所对应的子键的十六进制子键名如下表所示:

输入法	子键名
英语(美国)	00000409
全拼	E0010804
双拼	E0020804
郑码	E0030804
智能 ABC	E0040804
区位	E0050804
王码五笔	E0200804
微软拼音输入法	E00E0804
表形码	E00C0804

在每个子键名下面,包括有如下几项键值:

- IME File 指定输入法文件(*.IME)
- Layout File 指定键盘布局文件(*.KBD)
- Layout Text 指定键盘布局名称,该名称和我们用鼠标右键点击任务条右端的输入法指示器图标后选择“属性...”菜单项后弹出的“键盘 属性”对话框上方列表框中的“布局”名称一致。

2. preload 子键

该子键同时位于“我的电脑/HKEY_CURRENT_USER/keyboard layout /preload”和“我的电脑/HKEY_USERS/.Default/keyboard layout/preload”两个位置,对其中任何一个位置的修改都将同时改变另一个位置下的相应信息。preload 子键下面有一组子键名形如“1”,“2”,...的子键,具体数日视系统中所安装的输入法而定。每个子键对应一种输入法,其键值即为前面在 keyboard layouts 子键下定义的 8 位十六进制值。Windows 系统在启动时按照 preload 子键下各子键的顺序依次装入各种输入法,其中子键“1”所定义的输入法为默认输入法。

编程思想和示例程序

在了解了上面这些知识后,本文一开始所提出的问题就很容易解决了。在 Win32 API 中,有关输入法的函数有 ActivateKeyboardLayout(激活输入法),LoadKeyboardLayout(装入输入法),GetKeyboardLayoutName(获得输入法名称),GetKeyboardLayout(获得当前输入法)和 UnloadKeyboardLayout(卸载输入法)等,有关这些函数详细的使用说明可以参看相关的 Win32 SDK(Software Development Kit,软件开发工具包)。下面结合一个 VC++5.0 环境下的示例程序(图1)来具体演示一下该问题的解决方法。



图 1

①启动 VC, 建立一个名为 LayOut 的基于对话框(Dialog-based)应用程序。

②如图 1 设计对话框, 两个编辑框的 ID 分别为 ID_EDIT_NAME 和 ID_EDIT_EMAIL, 对话框中各控件的 Tab Order 设置为: 1: ID_EDIT_NAME, 2: ID_EDIT_EMAIL, 3: IDOK。

③在对话框类 CLayoutDlg 中添加两个数据成员, 用于记录用户所选择的输入法:

```
//LayOutDlg.h:header file
```

```
.....
class CLayoutDlg:public CDialog {
.....
private:
HKL hklEnglish;// 西文输入法句柄
HKL hklChinese;// 中文输入法句柄
};
```

④打开 CLayoutDlg::OnInitDialog() 函数, 在其中添加如下语句:

```
BOOL CLayoutDlg::OnInitDialog()
{
.....
// 把中、西文输入法分别初始化为
// “智能 ABC 输入法”
// 和 “英语(美国)”
hklChinese=LoadKeyboardLayout(
"E0040804",KLF_ACTIVATE);
hklEnglish=LoadKeyboardLayout
```

```
("00000409",KLF_ACTIVATE);
```

```
return TRUE;
```

```
}
```

⑤选择 “View/ClassWizard/MessageMaps”, 为编辑框 IDC_EDIT_NAME 的事件 EN_SETFOCUS 和 EN_KILLFOCUS 分别添加处理函数:

```
void CLayoutDlg::OnSetfocusEditName()
```

```
{
```

```
    // 激活中文输入法
```

```
        ActivateKeyboardLayout
```

```
(hklChinese,KLF_UNLOADPREVIOUS);
```

```
}
```

```
void CLayoutDlg::OnKillfocusEditName()
```

```
{
```

```
    // 保存当前中文输入法
```

```
    hklChinese=GetKeyboardLayout(0);
```

```
}
```

⑥同样, 为编辑框 IDC_EDIT_EMAIL 的事件 EN_SETFOCUS 和 EN_KILLFOCUS 分别添加处理函数:

```
void CLayoutDlg::OnSetfocusEditEmail()
```

```
{
```

```
    // 激活西文输入法
```

```
        ActivateKeyboardLayout
```

```
(hklEnglish,KLF_UNLOADPREVIOUS);
```

```
}
```

```
void CLayoutDlg::OnKillfocusEditEmail()
```

```
{
```

```
    // 保存当前西文输入法
```

```
    hklEnglish=GetKeyboardLayout(0);
```

```
}
```

⑦编译、连接, 生成可执行文件。

运行上面生成的应用程序, 初始输入焦点为 “姓名” 编辑框, 输入法已经自动设置为 “标准” 中文输入法, 而把输入焦点移动到 “E-MAIL” 编辑框时, 输入法会自动切换到 “英语(美国)” 输入法。如果用户在某个编辑框中改变了输入法, 程序会自动记录并在回到这个编辑框时恢复用户原来的输入法。例如, 用户把 “姓名” 编辑框的输入法切换为 “五笔” 输入法, 然后到 “E-MAIL” 编辑框中, 再回到 “姓名” 编辑框中时, 会发现程序自动又切换到 “五笔” 输入法了。在实际使用时, 我们甚至可以把用户在各数据域所使用的输入法存放在注册表中, 以供应用程序下次启动时直接设置为用户所期待的输入法。这实现起来并没有什么难度, 只是限于篇幅, 在此未予演示, 需要或有兴趣的朋友不妨一试。■