

VB 中 RichTextBox 控件的编程技术

杜树杰 (青岛海洋大学计算中心 266003)

摘要:如何在应用程序运行时有效地控制文本的编辑格式、设置灵活的排版功能,是用户经常遇到的问题。本文就 RichTextBox 控件自身具有的在编辑、排版方面的概念,结合其提供的属性和方法,从编程角度介绍了解决上述问题的一些技术环节。

关键字: RichTextBox 控件 OLEObject 悬挂式缩进

从 VB4.0 开始提供了 RichTextBox 控件,它具有比普通 TextBox 控件更高级的用户输入输出功能和控制文本编辑的格式特征。使用户可以在运行时有效地进行文字格式、排版等方面的多样化控制。它还可以直接读写指定文件,支持 OLE 对象的嵌入,当然也具有 TextBox 控件提供的数据库绑定功能以及各种属性、事件和方法,同时取消了 TextBox 控件 64K 字符能力的限制。下面从几个方面介绍 RichTextBox 的编程功能。

一、对 OLEObjects 集合的支持

RichTextBox 对每个嵌入的对象都表示为一个 OLEObject 对象,这允许你文档中创建的控件可以包含其他控件文档。我们可以通过 ADD 方法给一个 OLEObjects 集合增加一个 OLEObject 对象,其句法是 object.Add index, key, sourcedoc, class。其中 index 为整数, key 为字符串,它们均可视为 object 的索引标识, sourcedoc 是嵌入到 OLEObject 对象的文件名, class 是嵌入对象的 OLE 类名。如下面的代码在 RichTextBox 控件中增加一个 Excel 表格,并设置 KEY 标识为“PCPrice”: RichTextBox1.OLEObjects.add, "PCPrice", , "Excel.Sheet.5"。在运行时,还可用 Clear 和 Remove 方法,动态地向该控件中充填或删除嵌入对象。

对于 OLEObject 对象,还可以通过 Class 属性返回嵌入对象的类名,通过 DisplayType 属性返回或设置嵌入对象为图标还是内容,也可以打开 OLE 对象进行编辑,并通过 DoVerb 方法执行指定的 OLEObject 对象的谓词操作,而这些谓词列表可以通过 ObjectVerbs(Index)和 ObjectVerbsCount 属性获得。

二、RichTextBox 中字符的查找和选择

1. Object.Find(string, start, end, option)。其中 option 规定匹配方式,而省略 start 和 end 表示搜索整个 RichTextBox 文件。例如 AtPos = RichTextBox1.Find

(Text1.Text, , , rtfWholeWord OR rtfMatchCase)以整字匹配方式搜索全文,并区分大小写。

2. Object.Span charset, direction, negate。如果 negate 值为 False,所选字符是包含在 charset 中的字符,选择停止于首次发现不在 charset 中的字符位置。Direction 表示搜索方向。例如 RichTextBox1.Span "0123456789", True, True 表示搜索时一旦遇到数字符号便停止。

三、RichTextBox 控件与文件的关联方式

1. 以 RTF 格式或 ASCII 文本格式打开或保存文件。可以使用该控件的方法(LoadFile 和 SaveFile)直接读写文件,当用 LoadFile 方法加载一个文件时,加载文件的内容将取代 RichTextBox 控件的全部内容。这将使 Text 和 RTFText 属性的值改变。以下实例用 LoadFile 方法装入一个文本文件。

```
Private Sub LoadButton_Click()  
    CommonDialog1.Filter = "ASCII Text Format files  
| *.txt"  
    CommonDialog1.ShowOpen  
    RichTextBox1.LoadFile CommonDialog1.FileName,  
    rtfText  
End Sub
```

2. 利用该控件的属性。比如 SelRTF 和 TextRTF, 联合 Visual Basic 的文件输入输出语句读写文件。例如:
Open "textread.rtf" For Input As 1

```
RichTextBox1.TextRTF = Input$(LOF(1), 1)
```

可以将 textread.rtf 的内容加载到 RichTextBox1 中。又如:

```
Open "textcreate.rtf" For Output As 1  
Print # 1, RichTextBox1.SelRTF
```

将 RichTextBox 中选择的内容保存到一个 RTF 文件中。

3. 将 .rtf 文件的内容装入 RichTextBox 控件,只需

将该文件(比如从 Windows 95 “资源管理器”中)的内容直接拖入该控件,或者将其他应用程序(比如 Microsoft Word)中使用文件的突出显示部分直接拖入该控件。

4. 在设计阶段设置 FileName 属性,将 .rtf 或 .txt 文件的内容装入该控件。

5. 将 RichTextBox 控件绑定到 Data 控件,可以处理 Microsoft Access 数据库的 Memo 字段,或者其他数据库中大型的文本字段(比如 SQL 服务器中的 TEXT 数据类型)。

6. 在运行时,用上面提到的 OLEObject 集合的 Add 方法,将嵌入对象如位图、图标、应用程序图标,加到该控件中。在设计或运行时,可以拖放任何嵌入对象到控件中,包括文档,比如 Microsoft Excel 的电子表格和 Microsoft Word 文档等。

四、RichTextBox 的格式处理

1. 用 SelFontName、SelFontSize 和 SelFontColor 等设置字体属性

要改变 RichTextBox 控件中的字体风格,可以通过 SelFontName、SelBold、SelItalic、SelFontSize、SelUnderline、SelFontColor 等属性实现。下面的示例显示来自 CommonDialog 控件的一个字体对话框,用于指定 RichTextBox 控件中选定文本的字体属性。

```
Private Sub Command1_Click ()  
    CommonDialog1.Flags = cdlCFBoth or cdlCFEffects  
    CommonDialog1.ShowFont  
    With RichTextBox1  
        .SelFontName = CommonDialog1.FontName  
        .SelFontSize = CommonDialog1.FontSize  
        .SelColor = CommonDialog1.Color  
        .SelBold = CommonDialog1.FontBold  
        .SelItalic = CommonDialog1.FontItalic  
        .SelStrikethru = CommonDialog1.FontStrikethru  
        .SelUnderline = CommonDialog1.FontUnderline  
    End With  
End Sub
```

2. 格式化段落的缩进、悬挂式缩进和项目符号

RichTextBox 控件的另一个特性是它可以创建有缩进、悬挂式缩进和项目符号的段落。标准的缩进使所选的文本段落的左边界从 RichTextBox 框的左边界处向右移,这类缩进是利用 SelIndent 属性来完成的;SelRightIndent 属性还可以使所选的文本段落的右边界向左移;SelHangingIndex 属性可指定文本段落中第一行的后续行的左边再缩进的距离,又称为悬挂式缩进;还可以通过 Bul-

letIndent 属性实现带项目符号的悬挂式缩进样式,此时要注意段落的 SelBullet 属性必须设置为 True。显然 SelIndent、SelHangingIndent、BulletIndent 属性在设计时是不能设定的,只能在编程过程中改变其值。下表中的每段分别代表 SelIndex、SelHangingIndent 和 BulletIndent 属性设置后的结果。

该行设置 SelIndent 属性为 1 厘米

该行设置 SelIndent 属性为 0.5 厘米,同时设置 SelHangingIndent 属性为 1 厘米

※ 该行设置 SelIndent 属性为 0 厘米,同时设置 BulletIndex 属性为 1 厘米,要求 SelBullet 属性设置为 True

需要强调的是 RichTextBox 控件的容器的 ScaleMode 属性决定了 SelIndent、SelHangingIndent、BulletIndent 和其他属性值的单位。例如,如果 RichTextBox 控件的容器是 Form 对象,则将该 Form 对象的 ScaleMode 属性由 7(厘米)改为 1(Twip)时, SelIndent 属性必须同时由 0.5(厘米)改为 283(Twip)。这是因为一厘米 = 567 Twips。

3. 其他的格式设计事项

设置 RichTextBox 框的 SelAlignment 属性为 0、1、2,可以改变文本段落中的对齐方式。实现选择段落沿 RichTextBox 框的左中右对齐。

在 RichTextBox 控件中还有一个有用的属性 SelCharOffset。它的作用是使选中的文字从字体中间位置开始相对上移(SelCharOffset 取正整数)或下移(SelCharOffset 取负整数)若干个 Twips,形成类似于 WPS 中上下标号的效果。当然该属性提供了更多的编程控制效果。

RichTextBox 控件主要是对 RTF 文件实施排版控制的,也可用于编写阅读特别大的文本文件的应用程序,以及在程序中实现功能较为完备的文本编辑器。因此,诸如 ScrollBars、Change 事件(RichTextBox 控件中多使用 SelChange 事件)、文本打印(SelPrint 方法)等文本编辑器中通常具有的属性、事件和方法这里就不再多说了,读者可在编程实践中体味 RichTextBox 控件的强大功能。

参考文献

- [1] Microsoft 公司. Microsoft Visual Basic 5.0 ActiveX Controls Reference. Microsoft Press
- [2] Microsoft 公司. Microsoft Visual Basic 5.0 Online Reference. Microsoft Press

(来稿时间:1998年9月)