

在 VB 下开发 Windows Help 的方法

郑沛 (北京航空航天大学计算机系 100083)

作为 Windows 操作系统的一个突出优点,其应用程序强大的联机帮助系统使广大用户受益匪浅。评价应用软件的质量,一个主要的标准就是看软件的 Help 系统是否完善。利用 Visual Basic,可以方便地建立 Windows 下应用软件的 Help 系统,主要分为以下 5 步:

(1)由 Help 作者规划软件的 Help 系统,确定 Help 将提供的信息内容。

(2)由 Help 作者利用编辑器编写 Help 标题文件,并加入特定的编码信息。这个文件是 Help 系统的主体,文件中还包括一些控制代码,它们将决定用户如何在 Help 中移动。

(3)由程序员根据 Help 标题文件建立 Help 工程文件(*.HPJ),并利用 VB 专业版的 Help Compiler 进行编译,生成 Help 文件(*.HLP)。

(4)由程序员利用 VB 开发访问 Windows Help 的应用程序,调出上述 Help 文件(*.HLP),显示用户指定的标题。

(5)由 Help 作者和程序员对该 Help 系统进行调试和测试。

其中第 3 步“编写 Help 标题文件”和第 4 步“建立 Help 文件”是本文讨论的重点,其他 3 步就不详细介绍了,有兴趣的读者可阅读参考文献[2]。

1. 用 Microsoft Word 建立 Help 标题文件(*.RTF)

Help 作者根据用户的实际情况,拟定 Help 的结构和具体内容。实际上,Help 是由一些标题(Title)组成的,某些标题之间可以交叉引用。Help 作者必须首先列出所有标题组和单个标题,写出标题的内容,确定相关标题的交叉引用关系。在此基础上,可利用 Microsoft Word 建立标题文件。一般标题文件类型为 Rich Text Format,即 RTF,它通过使用控制代码来实现 Windows Help 包含的所有功能:

- * “#”号,定义唯一标识一个标题的上下文字串;
- * 美元符“\$”脚注,定义标题的题目,它将显示在书签(BookMark)及历史列表(History)中。
- * 字母“K”脚注,定义一个标题的关键字;

* 加号“+”脚注,定义浏览顺序号;

* 双下划线,表示标题间的交叉引用,选择其上字串可跳到相关标题;

* 隐藏正文(Word 中是虚下划线),它位于双下划线字串后面,指定交叉引用得标题上下文字串:

例如:Help Topic 1HELPTOPIC-ONE

其中,双下划线的“Help Topic 1”将显示在屏幕上,其后虚下划线上的“HELPTOPIC-ONE”是隐藏正文,并不显示。Help 运行时,用户点选“Help Topic 1”后,Help 将跳到以“HELPTOPIC-ONE”为上下文字串的标题中。

又如:在正文中,

\$ KHelp Topic 1

作为脚注符插入了“#”、“\$”和“K”,在脚注区中:

HELPTOPIC-ONE

\$ Topic One

k Topic One, Some keywords here

其中,“#”定义了一个标识本标题的上下文字串“HELPTOPIC-ONE”,“\$”定义了标题的题目“Topic One”,“K”为这个标题定义了两个关键字。

此外,标题之间要用硬分页隔开,标题文件中也可插入 BMP 图像文件。

2. 用 Help Compiler 建立 Help 文件(*.HLP)

Help Compiler 3.1 是 VB 3.0 专业版提供的命令行编译器,使用时首先要建立 Help 工程文件(*.HPJ),它由若干小节组成,必要的小节包括:

* [Files]节,列出标题文件名(*.RTF);

* [Options]节,包括指定 Help 系统的标题,编译系统的警告级别,是否压缩 Help 源文件等。例如:

[Options]

Title = Example Help 'Help 窗口标题

Contents = CONTENTS 'Help 窗口中 Contents 显示标题

Compress = false '不压缩 Help 文件

Warning = 3 '显示所有警告

* [Map]节,若使 Help 系统支持上下文相关标题,即支持“F1”键帮助功能,可为上下文字串指定一个数字。在下例中,两个上下文串 HELPTOPIC-ONE 和 HELPTOPIC-TWO 被赋予了数值 256 和 257。

[Map]

HELPTOPIC-ONE 0x100

HELPTOPIC-TWO 0x101

* [Config]节,指定 Windows Help 打开 Help 文件时执行的宏,例如:

[Config]

BrowseButtons()

为 Help 窗口增加了浏览按钮“<”和“>”。

* [Windows]节,指定 Help 窗口的属性,如位置,大小等。例如:

[Windows]

main = "Example Help", (0, 0, 1023, 1023),,,
(100, 100, 100)

定义了 Help 二级窗口标题,窗口坐标及背景颜色。

建立工程文件后,就可编译生成 Help 文件。Help Compiler 的用法如下:

HC31 Filename.HPJ

如果编译器发出警告或错误信息,可根据提示加以改正。

3. 用 Microsoft Visual Basic 实现应用程序的联机帮助功能

我们已经得到了 Help 文件(*.HLP)。为了测试 Help 文件,可以在 Windows (3.1)下运行 WinHelp.EXE,再用“File”菜单的“Open”命令打开这个文件,进行测试。用 VB 的目的,就是使上述过程自动化,也就是说,当用户在软件窗口中按“F1”键,或者选择了“Help”菜单中的某一项,或者按下了窗口中的 Help 按钮,程序自动打开相应 Help 文件的相关标题。在 Visual Basic 3.0 中,实现上述过程的方法包括:

(1)利用通用对话框(Common Dialog Boxes)。当通用对话框的 Action 属性值为 6 时,可执行 WinHelp.EXE,并根据其他属性的设置显示指定的标题,这些属性是:

①HelpFile: 该属性指定 Help 文件名。

②HelpCommand: 该属性指定 Help 帮助类型,即帮助信息命令。这些命令是全局常量,存于 Constant.TXT 中。如果 HelpCommand 设置为 HELP-INDEX,将显示

Help 索引(Index),如果 Help 没有索引将显示主标题 Contents。例如:

'show VB help index or contents

dlgExample1.HelpFile = "VB.HLP" ' Specify the
Help file to open.

dlgExample1.HelpCommand = HELP-INDEX ' When WinHelp.EXE is executed, Help Index will be displayed.

dlgExample1.Action = 6 ' Execute WinHelp.EXE.

上例中,通用对话框 dlgExample1 将作为一个 Help 窗口被打开,并且显示 VB Help 的 Contents。可以把这段代码写入某个控制的事件过程—例如命令按钮(Command Button)的 Click 事件中,或者 Help 菜单的 Click 事件中,从而实现相应的 Help 功能。

③HelpKey: 如果 HelpCommand 属性设置为 HELP-KEY,即根据关键字(Keyword)查找帮助信息,则 Help-Key 属性应设置为要找的关键字。例如:

'show VB help for keyword

dlgExample1.HelpFile = "VB.HLP" ' Specify the
Help file to open.

dlgExample1.HelpCommand = HELP-KEY ' When WinHelp.EXE is executed, Help for a specified keyword will be displayed.

dlgExample1.HelpKey = "MouseDown" ' Specify the keyword.

dlgExample1.Action = 6 ' Execute WinHelp.EXE

(4) HelpContext: 如果 HelpCommand 属性设置为 HELP-CONTEXT,即显示特定的上下文(Context),则必须为 HelpContext 指定该上下文标志(RTF 文件中“#”后的字串)。例如:

'show help for Context

dlgExample1.HelpFile = "MY.HLP" ' Specify the
Help file to open.

dlgExample1.HelpCommand = HELP-CONTEXT ' When WinHelp.EXE is executed, Help for a specified ContextID will be displayed.

dlgExample1.HelpContext = HELPTOPIC-ONE

dlgExample1.Action = 6 ' Execute WinHelp.EXE.

通用对话框 dlgExample1 指定“My.HLP”作为 Help 文件,并显示以“HELPTOPIC-ONE”为上下文标志的标题,“HELPTOPIC-ONE”定义在标题文件的脚注“#”后,

工程文件的[Map]段为这些标志赋了值。

(2)利用控制的 HelpContextID 属性。很多控制如窗体(Form)、多文档界面窗体(MDI Form)、菜单、文本框(Text Box)、列表框(List Box)、命令按钮(Command Button)等,都具有 HelpContextID 属性。该属性指定一个上下文字符串,当用户把焦点移到控制上按“F1”键时,Windows Help 将显示 Help 文件的指定上下文标题。设置这一属性之前,首先要为应用程序设定 HelpFile 属性,即 App.HelpFile = HelpfileName。例如:

```
Sub Form-Load()  
App.HelpFile = "my.HLP"  
txtExample1.HelpContextID = HELPTOPIC-ONE  
End Sub
```

其中,窗体中的文本框 txtExample1 的上下文相关标题被指定为“HELPTOPIC-ONE”,这个标志定义在标题

文件中,在工程文件的[Map]节中,它与一个数字相对应。

本文主要讨论用 VB 开发 Windows 下的 Help 的过程,实际上只是利用了 VB 控制的有关属性,执行 WinHelp.EXE,显示指定标题。当然也可以考虑在 VB 中直接调用 WinHelp.EXE,效果基本一样。笔者写了一个演示这个过程的例子,包括标题文件、工程文件和 VB 窗体文件,它覆盖了本文所涉及的所有方法,限于篇幅,就不列出了,有兴趣的读者可与笔者联系。

参考文献

- [1] 林丽等《精通 Visual Basic 3.0 for Windows》人民邮电出版社。
- [2] 李振格编译《Borland C++ 程序员手册》北京航空航天大学出版社。

(来稿时间:1996 年 9 月)