

Visual Basic 应用程序自动适应多种屏幕设定技术

许刚 严寒冰 (浙江丝绸工学院 电子系 310033)

摘要:本文介绍了三种使 Visual Basic 程序能动态适应多种屏幕设定的方法。

关键词:屏幕设定 Visual Basic API 函数 Screen 属性

在编写一些通用软件时,如果程序员没有考虑用户的屏幕设定(屏幕分辨率和色彩数)问题,则在用户改变屏幕设定时,应用程序中的控件、图片等会改变在屏幕中的相对位置,一些高清晰度的图片还会有失真现象,从而造成软件用户界面的混乱。

解决用户的多种屏幕设定方法的基本思想是:在程序启动时对系统的当前屏幕设定模式进行检测,在程序中与屏幕有关的所有模块都以这个测试值为基准来调整应用程序的界面,从而达到使程序自动适应不同屏幕设定的目的。本文介绍的基于 API 函数、利用 Screen 的属性和依靠第三方控件的三种具体的解决方法是我们在使用 Visual Basic 5.0 开发多媒体教学软件时所采用的技术。

一、基于 Win32 API 函数的屏幕模式检测

应用 User32.dll 和 Gdi32.dll 两个动态连接库所提供的功能,准确地检测到用户系统的屏幕分辨率和色彩数设定,并据此跳转到相应的程序模块。

首先在工程的模块文件中添加下述声明:

```
Declare Function GetDC Lib "user32" (ByVal hwnd As Long) As Long
```

```
Declare Function GetDesktopWindow Lib "user32" () As Long
```

```
Declare Function GetDeviceCaps Lib "gdi32" -  
(ByVal hdc As Long, ByVal nIndex As Long) As Long
```

第二步,在程序启动窗体的 Form-load 事件中写入下述代码:

```
hdesktopwnd = GetDesktopWindow()  
hdccaps = GetDC(hdesktopwnd)  
displaywidth = GetDeviceCaps(hdccaps, 8)  
displayheight = GetDeviceCaps(hdccaps, 10)
```

```
displaybits = GetDeviceCaps(hdccaps, 12)
```

```
displayplanes = GetDeviceCaps(hdccaps, 14)
```

程序中的 displaywidth 和 displayheight 变量就是用户屏幕的分辨率设定值。屏幕色彩数则需要根据 displaybits 和 displayplanes 的值进行判断,表 1 给出了几种常用的屏幕色彩数的判断值:

表 1 常用屏幕色彩数的判定值

	2 色	16 色	256 色	16 位增强色	32 位真彩色
Displaybits	1	1	8	16	32
Displayplanes	1	4	1	1	1

二、利用 Visual Basic 中 Screen 的属性

Visual Basic 中的实体 Screen 有两种基本属性: Screen.Width 和 Screen.Height,利用它们可以使程序动态地适应用户系统的屏幕分辨率,但这种方法不能检测到用户系统的屏幕色彩数设定(解决方法是将程序中所有的图片都转换为 256 色),即只能针对屏幕分辨率进行灵活的编程。

Screen.Width 和 Screen.Height 并不是指用户屏幕的实际尺寸,而只是一个逻辑上的单位,与用户的 CRT 大小(14 寸、15 寸或 17 寸等)无关。表 2 给出了几种常用的屏幕分辨率下 Screen.Width 和 Screen.Height 的取值情况。

表 2 常用屏幕分辨率的屏幕属性值

	640 * 480	800 * 600	1024 * 768	1280 * 1024
Screen.Width	9600	12000	15360	19200
Screen.Height	7200	9000	11520	15360

下面用一个简单的示例程序来介绍利用 Screen 实体的 Width 和 Height 属性实现应用程序动态适应多种屏

幕分辨率的具体方法。

首先,在程序启动窗体的 Form-load 事件中写入下述代码:

screenflag = 0 'screenflag 是全局的 Long 型变量,用于存放屏幕设置标志

Dim screenh as Long, screenw as Long

'将变量声明为 Long 型,是因为 Long 型变量有益于提高 32 位程序的运行速度

screenh = Screen.Height

screenw = Screen.Width

'将程序中经常用到的控件属性放入变量有利于提高程序的运行速度

Dim sx as Currency, sy as Currency

'sx 和 sy 为控件位置改变的比例系数

Select Case screenh

Case 7200

If screenw = 10800 Then

MsgBox "分辨率为 720 * 480!", 48

screenflag = 1 '置标志位

sx = 10800/9600 '计算比例

sy = 7200/7200 '计算比例

End If

Case 9000

If screenw = 12000 Then

MsgBox "分辨率为 800 * 600!", 48

screenflag = 2 '置标志位

sx = 12000/9600 '计算比例

sy = 9000/7200 '计算比例

End If

Case 11520

If screenw = 15360 Then

MsgBox "分辨率为 1024 * 768!", 48

screenflag = 3 '置标志位

sx = 15360/9600 '计算比例

sy = 11520/7200 '计算比例

End If

Case 15360

If screenw = 19200 Then

MsgBox "分辨率为 1280 * 1024!", 48

screenflag = 4 '置标志位

sx = 19200/9600 '计算比例

sy = 15360/7200 '计算比例

End If

Case 18000

If screenw = 24000 Then

MsgBox "分辨率为 1600 * 1200!", 48

screenflag = 5 '置标志位

sx = 24000/9600 '计算比例

sy = 18000/7200 '计算比例

End If

'默认屏幕设定为 640 * 480(screenflag = 0)

End Select

然后,在其他窗体的 Form-load 事件中写入下述代码
If screenflag < > 0 Then '按比例调整控件的位置、大小和字体的模块

Controll1.top = Controll1.top * sy 'Controll1 为控件名,如 cmdhlp 等,下同

Controll1.left = Controll1.left * sx

.....

End If

Select Case screenflag '根据 screenflag 的值读入不同大小的图片

Case 1

Controll1.picture = loadpicture("xx1.jpg")

Case 2

Controll1.picture = loadpicture("xx2.jpg")

Case 3

Controll1.picture = loadpicture("xx3.jpg")

Case 4

Controll1.picture = loadpicture("xx4.jpg")

Case 5

Controll1.picture = loadpicture("xx5.jpg")

'screenflag = 0 时, Controll1 的 picture 属性为默认值,不做改变

End Select

三、依靠第三方控件

市场上有许多第三方软件商或个人编写的控件,其中的一些可以给用户所编写的应用程序提供增强性功能——如有专门用于播放 mp3 音效格式或 Gif 动画的,有用于数据分析、绘制立方体图表的……等等,其中有一种能使应用程序适应多种屏幕分辨率的“容器”型控件,编

(下转第 47 页)

(上接第 72 页)

程者只要将其他控件(如 Image、Command 等)放入其中,应用程序就可以动态地适应用户屏幕多种分辨率的设定了。

总结

比较上述的三种方法,第一种方法最适合用于编写较大规模的应用程序,因为这些程序往往对用户界面和内存容量要求较高,这样,使用该方法可以根据所测得的数据精确地动态调整用户界面,并根据用户屏幕的色彩数和分辨率动态地调入或卸载相应色彩数和分辨率的应用程序所使用的图片(如背景图片)等,其缺点是程序代码较为复杂,安装程序较大。第二种方法则适合用于编写那些对程序界面要求不高、安装程序大小受限的小型 and 袖珍型程序(如网上的共享软件等)。第三种方法的最大特点是能够大大提高编程的效率,但同时会影响应用

程序的运行速度和大小,另外,第三方控件的购买或注册费用也是相当可观的。

在实际的多媒体教学软件开发中,我们大都使用第二种方法进行编程——因为它代码简单,运行速度快。此外,用 Visual Basic 5.0 编写适应多种屏幕分辨率的应用程序还有许多其他的方法和技巧,如将各个窗体的“StartUpPosition”属性设为 2 - CenterScreen;用“Stretch”属性设为 True 的 Image 控件来存放图片或自定义 *.ocx 控件等,都不失为好的解决方法。

参考文献

- [1] Noel Jerke(美)等著,袁兆山、张健、袁晓辉等译,《Visual Basic 4 API 程序设计》,机械工业出版社,1997.1
- [2] Lee Adams(美)著,施小龙、葛玉宝、邓明辉译,《Visual Basic 动画编程》,学苑出版社,1994.6

(来稿时间:1999 年 2 月)