

文章编号: 1001-9081(2001)09-0091-03

用 MFC 开发 ActiveX 控件以激活 Web 页面

赵国庆, 刘家彬

(北京石油化工学院 计算机系, 北京 102600)

摘 要:通过实例介绍了怎样利用 VC++ 开发 ActiveX 控件, 然后编写脚本将其插入 Web 页面。从而展示了 Active X 控件在网络开发中的强大生命力。

关键词:MFC; VC++ ; ActiveX; Web

中图分类号: TP311.52 **文献标识码:** A

1 引言

ActiveX 是微软公司的 Internet/Intranet 整体战略的一个组成部分。该战略意在帮助 Web 开发者们开发 Internet/Intranet 工具; 为 Web 站点开发动态的、丰富的内容。

ActiveX 包括三个主要组件, 用户可以利用它们更好地控制自己站点的内容以及内容的布局。这三个组件是: ActiveX 文档、ActiveX 控件和 ActiveX 脚本。ActiveX 文档是一种不需要做任何修改和处理就可以在支持 ActiveX 的浏览器中浏览其内容的文档。目前, 直接支持 ActiveX 的浏览器只有 MSIE (Microsoft Internet Explorer) 3.0 及其以上版本; Netscape 可以通过一个插件间接支持 ActiveX。Microsoft Office 中的 WORD 和 Excel 文档是两个在支持 ActiveX 的浏览器中可以浏览其内容的例子。

ActiveX 控件可以在任何支持 ActiveX 的容器中使用; 在 Microsoft Word 97 和 Excel 97 文档中; 在 Microsoft Access 和 Visual FoxPro 数据库应用程序中; 在微软的 VB、VC 中等。

然而, ActiveX 控件最令人兴奋之处在于它是一个网络控件。这就意味着 ActiveX 控件可以插入 Web 页并成为 Web 页的直接组成部分。ActiveX 控件是激活 Internet/Intranet 网的最新途径, 它使以前的静态网重获新生。本文将通过一个实例——DANCE.OCX 来详细介绍怎样用 VC++ 开发 ActiveX 控件。

ActiveX 脚本使得 Web 开发人员可以方便地将 ActiveX 控件加入用于创建 Web 页面的 HTML 文档中。编写 ActiveX 脚本的目的就是允许用户控制 ActiveX 控件的行为。本文给出了一个利用 VBScript 对 ActiveX 控件编写的例子——DANCE.HTM, 以激活 Web 页面。

2 ActiveX 控件开发实例: DANCE.OCX

本例设计思路受到 Microsoft J++ 中提供的一个 Java applet 例子“Nervous Text”的启发而创建的 DANCE.OCX 控件, 它是一个基本的 ActiveX 控件。

DANCE.OCX 控件使使用该控件的用户提供一个文本串中的每一个字符上下左右跳动, 看上去就好像文本在跳舞一样, 因而得名。通过编写脚本, 用户可以改变文本串内容、文本前景色、文本背景色、文本默认字体和跳动频率。

在 VC++ 5.0 开发环境中, 可按以下步骤开发 DANCE.OCX 控件:

2.1 创建 DANCE.OCX 控件框架

开发 DANCE.OCX 控件的简单方法就是首先利用 VC++ 5.0 提供的 MFC ActiveX ControlWizard 创建 ActiveX 的控件框架, 然后在此基础上添加必要的功能。创建 DANCE.OCX 控件框架的步骤如下:

- 运行 VC++ 5.0, 从 File 菜单选择 New, 出现 New 对话框。

- 在 New 对话框中选择 Projects | MFC ActiveX ControlWizard; 在 Project name 框中键入 DANCE; 在 Platforms 框中选中 Win32 复选框, 按 OK。

- MFC ActiveX ControlWizard 需要两个步骤来创建一个完整的 ActiveX 控件框架: 第一步取其默认值; 第二步选择复选框 Activates when visible Available in "Insert Object" dialog; 由于 DANCE.OCX 控件要插入 Web 网页, 为了提高系统的性能, 采用异步方式下载 DANCE.OCX 控件, 为此按 Advanced 按钮, 选中 Loads properties asynchronously 复选框, 按 OK 返回第二步。

- 按 Finish 按钮完成 DANCE.OCX 控件框架的设置, 出现 New Project Information 对话框。

- 按 OK 创建 DANCE.OCX 控件框架, 返回 VC 开发环境。通过以上几个步骤, 创建了构成 DANCE.OCX 控件的所有必要的文件和模板, 如下:

类模板	文件	描述
CDanceApp	DANCE.H	DANCE.OCX 控件主文件
	DANCE.CPP	
CDanceCtrl	DANCECTL.H	实现 DANCE.OCX 控件的实际控件功能
	DANCECTL.CPP	
CDancePropPage	DANCEPPG.H	提供一个控件属性页模板
	DANCEPPG.CPP	

以及几个文件:

文 件	描 述
DANCE.OLD	定义 DANCE.OCX 控件类型信息的文本文件
DANCE.RC	标准资源文件
DANCECTL.BMP	控件位图

收稿日期: 2001-06-07 (修改稿)

作者简介: 赵国庆 (1964-), 男, 讲师, 硕士, 主要研究方向: 网络开发、数据库; 刘家彬 (1962-), 男, 副教授, 博士, 主要研究方向: 网络开发、CIT 技术。

现在, DANCE.OCX 控件框架已经建立, 可以选择 Build 来对此控件进行编译。VC++ 自动创建一个类型库 (DANCE.TLB) 并且在系统登记库中登记这个控件, 其他用户就可使用该控件了。

如果你在 VC++ 提供的一个应用程序 ActiveX Control Test Container 中测试 DANCE.OCX 控件的话, 在视觉上, 此控件只是一个简单的椭圆, 没有任何功能。下面给控件添加必要的功能。

2.2 给 DANCE.OCX 控件框架添加功能和属性

可以修改 CDanceCtrl 类中的代码来实现特定的控件功能和属性以及预定义属性页; 修改 CDancePropPage 类中的代码及其对话框资源来实现特定的控件自定义属性页。

· 修改 CDanceCtrl 类添加功能和属性以及预定义属性页

要给 DANCE.OCX 控件添加的功能就是使一个文本串中的每一个字符上下左右跳动, 必须人工编写代码。本例是通过自定义函数 DanceText() 实现的, 利用系统定时器每隔一定时间调用一次 DanceText() 函数。由于篇幅所限, 不列出此函数程序清单, 只给出实现思路。首先, 根据当前字体计算文本字符串的边界矩形; 其次, 产生随机数确定文本串中一个字符的显示位置, 将该字符画到一个内存位图中, 并利用 CDC::BitBlt() 函数粘贴到控件用户区上, 循环直到所有字符画完为止。

添加的属性有: 文本内容、跳动频率、前景色、背景色以及默认字体, 可利用 ClassWizard 来完成。进入 New|ClassWizard|Automation 页面, 为 DANCE.OCX 控件添加如下属性:

- Caption: 跳动的文本;
- Interval: 系统定时器定时间隔(毫秒);
- ForeColor: 文本前景色;
- BackColor: 文本背景色;
- DanceTextFont: 文本字体;

属性分为预定义属性和自定义属性, 本例只需为 Interval 和 DanceTextFont 这两个自定义属性添加方法。下表汇总了为 DANCE.OCX 控件添加的 7 个方法:

方法	说明
OnCreate()	在创建 DANCE.OCX 控件窗口时, 启动系统定时器
OnDestroy	在撤消 DANCE.OCX 控件窗口时, 停止系统定时器
OnTimer()	由 MFC 调用以响应 WM_TIMER 消息
OnIntervalChanged()	当 Interval 自定义属性被改变时, 重新设置系统定时器
GetDanceTextFont()	获取由 m_DanceFont 数据成员保存的用户字体
SetDanceTextFont()	将用户通过字体属性页设置的字体保存到 m_DanceFont 数据成员中
DanceText()	使文本串中的每一个字符上下左右跳动

以上 7 个方法, 都可根据其说明编写出相应的代码。

在此, 重点对 GetDanceTextFont() 和 SetDanceTextFont() 两个方法加以说明。MFC 的 CFontHolder 类封装了 Windows 字体对象和 ActiveX Font 接口的全部功能, 用户通过这个类对象提

供的接口可以非常方便的获得与字体有关的信息。而系统没有提供预定义的 CFontHolder 字体对象, 故必须自己定义。在 CDanceCtrl 类定义头文件 (DanceCtrl.H) 中说明一个保护型数据成员:

```
CFontHolder m_DanceTextFont;
```

下面修改 CDanceCtrl 类的实现文件 (DanceCtrl.CPP), 以获取或设置 m_DanceTextFont。

通过 GetDanceTextFont() 获取 m_DanceTextFont 的值, 通过 SetDanceTextFont() 修改 m_DanceTextFont 的值。具体实现是:

```
LPTONIDISP CDanceCtrl::GetDanceTextFont()
{
    return m_DanceTextFont.GetFontDispatch();
    //通过字体接口获得字体信息
}

static const FONTPDESC _fontdescDance = {
    sizeof(FONTPDESC), OLESTR("宋体"), FONTSIZE(24),
    FW_NORMAL, ANSI_CHARSET, FALSE, FALSE, FALSE
};

void CDanceCtrl::SetDanceTextFont(LPTONIDISP newValue)
{
    // 加入属性处理
    m_DanceTextFont.InitializeFont(&_fontdescDance, newValue);
    OnFontChanged(); //通报字体改变
    SetModifiedFlag(); //自动生成
}
```

由 CFontHolder 字体接口获取字体信息; 通过 _fontdescDance 结构间接修改字体信息, 而不能直接修改 m_DanceTextFont 数据成员。通过下面三步将 m_DanceTextFont 与 _fontdescDance 联系在一起:

(1) 在 CDanceCtrl 的构造函数中登记 m_DanceTextFont 数据成员, 使控件字体由 m_DanceTextFont 标识。

(2) 定义 FONTPDESC 结构变量 _fontdescDance:

```
static const FONTPDESC _fontdescDance = {
    sizeof(FONTPDESC), OLESTR("宋体"), FONTSIZE(24),
    FW_NORMAL, ANSI_CHARSET, FALSE, FALSE, FALSE
};
```

(3) 在 DoPropExchange() 函数中将 m_DanceTextFont 与 _fontdescDance 建立联系:

```
void CDanceCtrl::DoPropExchange(CPropExchange * pPX)
{
    PX_Font (pPX, _T("DanceTextFont"), m_DanceTextFont, &_fontdescDance); //字体
    PX_Short(pPX, _T("Interval"), m_nInterval, 100); //时间间隔, 默认为 100 毫秒
}
```

通过以上三步, 就将 m_DanceTextFont 与 _fontdescDance 联系在一起了。另外, 为了使用户可以通过属性页修改属性, 必须将 ControlWizard 生成的代码修改为:

```
BEGIN_PROPPAGEIDS(CDanceCtrl, 3)
PROPPAGEID(CDancePropPage, guid)
PROPPAGEID(CLSID_CFontPropPage)
PROPPAGEID(CLSID_CColorPropPage)
END_PROPPAGEIDS(CDanceCtrl)
```

CLSID_CfontPropPage 和 CLSID_CcolorPropPage 是 MFC 预定义值, MFC 利用它们将颜色属性页和预定义字体属性页加入控件的属性页中。

- 修改 CDancePropPage 类添加自定义属性页

利用对话框编辑工具建立自定义属性页, 在建立过程中系统自动修改 CDancePropPage 类中的相应代码。对于系统预定义的属性页都是一些标准页面, 用户只要在 CDanceCtrl 类的实现文件中作相应说明即可。

2.3 测试 DANCE.OCX 控件

- 在 ActiveX Control Test Container 中测试。

在 VC++ 环境下, 运行 Tools 菜单下的 ActiveX Control Test Container 应用程序

选择 Edit/Insert OLE Control 插入 DANCE.OCX 控件; 然后, 选择 View/Properties 改变控件的文本等属性, 看其效果。

- 还可以在 VB 或者 Delphi 中测试。

3 将 DANCE.OCX 控件加入网页

通过对 DANCE.OCX 控件进行全面测试, 证实其功能完全正常, 现在把它放入一个网页, 通过浏览器检验其功能。

利用 ActiveX Control Pad 工具, 可以节省开发时间。此工具是为将 ActiveX 控件插入网页设计的工具, 它也可将网页控件生成基本的 VBScript 和 JavaScript 源代码。

下面给出用手工实现的将 DANCE.OCX 控件插入网页的 HTML 文件(DANCE.HTM):

DANCE.HTM 完整源代码:

```
<HTML>

<HEAD>
<TITLE>会跳舞的文本</TITLE>
</HEAD>
<BODY>

<h2><!-- 会跳舞的文本 --></h2>
<br>
```

这是一个用 VC++ 5.0 编写的 ActiveX 控件。该控件可以使文本跳动, 看上去好象文本在跳舞一样。它支持以下特性: Caption(文本标题), Forecolor(前景色), Backcolor(背景色) 和 DanceTextFont(默认字体)。

```
<br>
<OBJECT ID="dance1" WIDTH=0 HEIGHT=0
CLASSID="CLSID:d01d019e-f89a-11d1-8572-d1b4310c665f"
codebase="/dance.ocx">
<param name="caption" value="ActiveX">
<param name="forecolor" value="255">
</object>
<!--
<OBJECT ID="dance2" WIDTH=0 HEIGHT=0
CLASSID="CLSID:d01d019e-f89a-11d1-8572-d1b4310c665f"
codebase="/dance.ocx">
<param name="caption" value="会跳舞的">
```

```
<param name="forecolor" value="65280">
</object>
<OBJECT ID="dance3" WIDTH=0 HEIGHT=0
CLASSID="CLSID:d01d019e-f89a-11d1-8572-d1b4310c665f"
codebase="/dance.ocx">
<param name="caption" value="文本">
<param name="forecolor" value="16711680">
</object>
<script language="vbscript">
<!--
sub window_unload()
dance1.Interval = 50
dance1.DanceTextFont.name = "Arial"
dance1.DanceTextFont.size = 20
dance1.DanceTextFont.Italic = True
dance1.DanceTextFont.Bold = True
dance2.Interval = 100
dance2.DanceTextFont.name = "黑体"
dance2.DanceTextFont.size = 30
dance2.DanceTextFont.Italic = True
dance2.DanceTextFont.Bold = True
dance3.Interval = 150
dance3.DanceTextFont.name = "幼圆"
dance3.DanceTextFont.size = 20
dance3.DanceTextFont.Italic = True
dance3.DanceTextFont.Bold = True
end sub
-->
</script>
</BODY>
</HTML>
```

以上代码中的 CLASSID 给出 DANCE.OCX 控件的类标识, 在网络中, 类标识唯一标识一个控件, 由系统自动随机生成并注册到系统登记表中。网页中插入了三个 DANCE.OCX 控件, 控制文本的内容、颜色、字体等。

4 结束语

ActiveX 控件是激活 Internet 的最前沿技术, 它为网络注入了生机。可以预料, 在未来的网络发展中 ActiveX 控件将起到决定性的作用。

另外, 本文虽以 VC++ 5.0 为例介绍 ActiveX 控件的开发过程, 实际利用其它语言也可以开发 ActiveX 控件。用不同语言开发的 ActiveX 控件是兼容的, 即 ActiveX 控件是与语言无关的。

参考文献

- [1] Kate Gregory. Visual C++ 5.0 开发使用手册[M]. 北京: 机械工业出版社, 1998.
- [2] Jeffry Dwight, 等. CGI 开发使用手册[M]. 北京: 机械工业出版社, 1998.
- [3] Que Corporation. Visual C++ 2.0 使用指南[M]. 北京: 清华大学出版社, 1995.