

# Java Applet 调用 ActiveX 控件的技术

杨少波 (中国科学院计算技术研究所 100080)

**摘要:**本文着重讨论了如何在 Java Applet 中调用 ActiveX 控件的方法,达到在网页中使用 ActiveX 控件以设计出动态交互式的网页。

**关键词:**Java Applet ActiveX 控件 交互式网页

## 一、概述

Java Applet 是 Sun 公司的 Java 语言网页小程序,它常被内嵌到网页文档中,丰富网页以产生动态可交互式的网页。在网页中内嵌 Java Applet 是采用 <Applet code = 程序名.class> </Applet> 标签及相应的属性参数。本例为:

```
<Applet code = updatecontrols.class ID = updatecontrols width = 270 height = 130>
  <param name = "CABBASE" value = "Javacab.cab"
>
</applet>
```

其中 ID 指示一个标识名,提供给 VBScript 网页脚本描述语言使用,而 CABBASE 参数用来指明经过数字签名的 CAB 文件。在网页中内嵌 Java Applet 控件的主要难点在于需经过复杂的编程实现。

ActiveX 控件是微软面向对象及分布系统的基石,它允许用户用任何一种语言及工具来创建软件组件,并可在任何一种容器环境下使用该组件。由于它具有强大的资源级访问能力,可充分使用本地机的资源,在 Internet 网中是 Sun 公司的 Java Applet 及 JavaBeans 的强有力的竞争对手。

Microsoft HTML Intrinsic Controls 是包含在微软浏览器 IE3.0 中的一种 ActiveX 控件,主要用来增强 HTML 固有控件的交互能力,并可被 VBScript 脚本描述语言驱动。而 HTML 固有控件是采用 HTML 超文本描述语言经过简单的描述而产生的,实现起来较容易。本例中的部份控件的描述如下:

```
<p><input type = button name = btnButton value
= "Text Button"><p>
<input type = checkbox name = chkCheck value = "
Checkbox"> Checkbox</td>
```

如能在网页中实现 Java Applet 与 HTML 控件之间的双向交互,则不仅可以增强网页的交互性能,同时也简化了控件的实现过程。本文通过实例详细地说明了如何在 JavaApplet 中调用 Microsoft HTML Intrinsic Controls ActiveX 控件的成员方法来响应 HTML 控件的用户事件,以改善网页的交互性能,并达到在网页中共享各种资源。

## 二、Java Applet 中引用 ActiveX 控件

由于 ActiveX 控件是 COM(Component Object Model) 组件,因而在 Java Applet 中要共享 COM 组件的资源,应首先将其转化为 Java 的类文件。

(1) 转化为 Java 类文件。在微软 Visual J++ 集成环境中,选中 Java Type Library Wizard 工具子菜单,将 Microsoft HTML Intrinsic Controls ActiveX 控件转化为 Java 类文件并产生接口说明 Summary.txt 文件(本例所产生的文件存储在 C:\windows\java\trustlib\intrinsic 文件夹中)。

(2) 使用 ActiveX 的 Java 类。利用 import intrinsic.\* 方式来调用转化后的 ActiveX 控件的 Java 类及接口中的 public 成员方法。本例中的调用形式为:

```
public boolean action(Event evt, Object what)
{
if (evt.target == updateBtn)
{
String str = newTextValue1.getText();
try
{
javaButton.putvalue(str); //ActiveX 控件的异常
处理代码
}
}
```

```

catch(com.ms.com.ComFailException e)
{
    showStatus("COM Exception:" + e.getMessage
());
}
if(str.compareTo(value1) == 0)
    newTxtValue1.setText(value2);
else
    newTxtValue1.setText(value1);
return true;
}
if (evt.target == updateCheck)
{ //ActiveX 控件的 public 成员方法
if (javaCheckbox.getchecked())
    javaCheckbox.putchecked(false);
else
    javaCheckbox.putchecked(true);
return true;
}
return false;
// . 其他事件响应
// .
// .
}

```

### 三、联接 Java Applet 和 ActiveX 控件

要实现在 Java Applet 中调用 ActiveX 控件的方法, 必须将它们两者相互关联起来。但由于 Java Applet 不能直接响应 ActiveX 控件的事件, 可采用两种方式联接它们。

(1) 利用 VBScript 来实现关联。VBScript 是微软设计的网页脚本描述语言, 采用 VB 语法。其实现原理是通过 VBScript 函数来处理 ActiveX 控件的事件, 然后在函数体中再调用 Java Applet 中的 public 成员方法。本例中的部份控件的关联代码如下:

```

<script language = VBScript >
<! --
sub LstList-OnFocus
    document.updatecontrols.setTextList "List Focus"
end sub

```

```

sub btnButton-OnClick
    document.updatecontrols.setTextText btnButton.value
end sub
sub chkCheck-onClick
    document.updatecontrols.setTextCheck chkCheck.value
end sub
! -- >
</script>

```

(2) 将 ActiveX 控件作为参数传给 Java Applet 的成员方法。关联 ActiveX 控件与 Java Applet 的另一种方式是将 ActiveX 控件作为实参, 在函数调用时传给 Java Applet。其实现方法是设计 window-onLoad 函数体代码, 在浏览器下载网页时调用 Java Applet 中的一个 public 成员方法 (如 loadControls()), 将网页中所定义出的各个 HTML 固有控件作为实参传给该成员方法。本例中的调用方式为:

```

<script language = VBScript >
<! --
Sub window-onLoad
    document.updatecontrols.loadControls lstList, btnButton, chkCheck
end sub
! -- >
</script>

```

### 四、Java Applet 中处理 ActiveX 控件的错误

ActiveX 控件的成员方法一般均返回一个 HRESULT 32 位的错误代码。微软 IE 浏览器定义了一个类 com.ms.com.ComException, 可以处理 COM 组件对象的错误, 但由于该类是派生于 RuntimeException, 因而可不必捕捉它们。

ComException 有两个子类, ComFailException 及 ComSuccessException, 其中 ComFailException 常是用户程序捕捉的对象。本例中的异常处理代码 (见前面)。

### 五、IE 浏览器之外实现 Java Applet 与 ActiveX 控件的交互

在微软的 Visual J++ 集成环境中, 自动将 Java Applet 类看成为信任类, 因而在 VisualJ++ 中执行网页中

的 Java Applet 时, 可以直接调用 ActiveX 的成员方法, 因而无需进行打包并数字签名。但在其外, 浏览器 IE 则将调用 ActiveX 中的成员方法的 Java Applet 类看成为不信任类, 因而应进行打包并对它进行数字签名。

微软的 CAB(cabinet)文件目前能组合 Java Applet 所需的各个类文件、声音文件及图象文件为单一 CAB 文件。由于它是压缩形式的, 因而下载较快并由浏览器自动解压; 另一方面, 如对此 CAB 文件进行数字签名, 则能确保用户在下载文件时不受到意外的损害。本例的具体做法如下。

(1) 利用 CABARC(CABinet ARChiver)工具来创建数字签名式 CAB 文件

```
C>Cabarc Javacab.cab * .class
```

(2) 利用 CODE SIGNING KIT 工具来数字签名一

个 CAB 文件, 并借助于 CODE SIGNING WIZARD 来简化操作过程, 从而生成带有数字签名的压缩文件 Java-cab.cab。而驻留在经过数字签名的 CAB 文件中的 Java Applet 类, 拥有对 ActiveX 控件的完整访问权限。

(3) 在 HTML 网页文档中指定数字签名的 CAB 文件。当使用 CAB 文件来存储 Java Applet 程序的类时, 必须在 HTML 页中指定 CAB 文件, 这可在 <APPLET> </APPLET> 标签中由 CABBASE 参数来完成。

### 参考文献

[1] 杨华中, 贾耀炜, Java 语言与程序设计, 人民邮电出版社, 1997, 2

(来稿时间: 1997 年 12 月)