

用 VC5 确保网页上的图像和超链接正常显示和使用

未 未 (清华大学自动化系, 100084)

摘要:本文指出了经本地计算机处理后的网页在浏览器上重新显示时可能出现的问题,分析了产生此类问题的原因,最后提出了解决此类问题的方法。

关键词:网络 HTML 图像 超链接

现在越来越多的程序,如网络翻译浏览器、网页内容过滤器(通过它家长可选择让孩子观看的网页内容)以及各种网络信息搜集程序等等,都需要“截获”浏览器正在显示或即将显示的网页进行处理,然后将它们重新加以显示。但在实现过程中常会碰到这样的问题:经本地程序处理过的网页再显示时页面上的图象和超链接有时无法正常显现和使用。笔者从自身的开发实践中总结出一个方法与读者共享。

= "image/a1.jpeg">所示的相对寻址方式的话,则其实际代表的绝对地址应是服务器端硬盘上的 C:\inetpub\wwwroot\image\a1.jpeg(一般说来服务器域名部分对应服务器硬盘上的"C:\inetpub\wwwroot\"目录);而处理后的网页在显示时这种相对寻址方式会被浏览器认为代表的是客户端硬盘上的 D:\HTML\image\a1.jpeg。若路径不对,图象就无法显示了。超链接也会因为此类问题而无法实现链接功能。

一、网页图象和超链接不能正常显示的原因

网页中的图象、超链接类标签多以 `` 和 `` 为代表,它们的 src 值或 href 值既可采用如上所示的绝对 URL 寻址方式,可采用各自对应的 `` 和 `` 所示的相对寻址方式。相对寻址方式的意义是相对于正在显示的网页的 URL 来进行叠加寻址。如果正在显示的网页的 URL 为 `http://www.tsinghua.edu.cn`, 相对地址为 `image/a1.jpeg`, 则浏览器认为叠加后的实际地址为二者的组合: `http://www.tsinghua.edu.cn/image/a1.jpeg` —— 这与 C 语言中 open 语句的文件 path 参数既可采用绝对路径,也可采用相对路径的问题类似。

之所以会出现图片和超链接显示使用不正常,是因为很多处理网页的程序通常是将网页中的 HTML 标签原封不动地保留下来,仅对其显示的文本内容进行处理,又将处理后的网页以一个本地硬盘 HTML 文件的形式显示在浏览器上。这时显示的网页文件的路径可能已变成了 `D:\HTML\index.html`, 而对应的原网页在服务器上的地址却可能是 `C:\inetpub\wwwroot\index.html`。如果原网页中的图片或超链接采用了如 ``, 则我们仅需截取地址框显示的 URL 前面的服务器域名部分,即 `http://www.tsinghua.edu.cn/`, 将其加到相对寻址方式之前即可变为 `http://www.tsinghua.edu.cn/image/a1.jpeg` 的绝对寻址方式。下面是以此图象标签为例的有关程序段。

1. 截取服务器域名

```
void WebInterface::AnalyzeURL()
```

```
{
```

```
// m_URLString 为完整 URL: http://www.ts-
```

```

inghua.edu.cn/index.html
    NeatestURLWithoutAppendedFileName = m _ URL-
String;
    for ( int k = 7; ( k < NeatestURLWith-
outAppendedFileName. GetLength()) &&
        ( NeatestURLWithoutAppendedFileName [ k ]!
= '/' ); k + + )
        {};
    //成为 http://bbs.njtu.edu.cn/的形式
    NeatestURLWithoutAppendedFileName = Neates-
tURLWithoutAppendedFileName. Left(k + 1);
}

```

2. 改造相对寻址方式

```

void WebInterface::ChangeImageURLToAbsolute
URL(CString &
    MainHTMLFileTestSubString)
{
    // MainHTMLFileTestSubString 中有 HTML 标签:
    <img src = "image/a1.jpeg">
    CString HTMLMarkString = MainHTML-
FileTestSubString;
    CString HTMLMarkStringLeft = "";
    CString HTMLMarkStringRight = "";
    int i = 0;
    //去掉 HTML 标签前后的“<”和“>”并取得其第一
    个空白符之前的小字符串 img
    MainHTMLFileTestSubString = MainHTML-
FileTestSubString. Mid(1, MainHTMLFileTestSu-
bString. GetLength() - 2);
    MainHTMLFileTestSubString = MainHTML-
FileTestSubString. SpanExcluding(" ");
    //判断是否 img 并处理
    if(MainHTMLFileTestSubString. GetLength() = = 3)
    {
        if(MainHTMLFileTestSubString = = "img")
        {
            for(int SubstituteFlag2 = 0; SubstituteFlag2 = =
0;)
            {
                while(HTMLMarkString. Mid(i, 1)! = = "")
                {i + +;}
                //跳过 = 和"使 i 对准 URL 的第一个字符

```

```

        i = i + 2;
        //判断是否相对寻址方式
        if(HTMLMarkString[i] = = '/' ||
            (HTMLMarkString. Mid(i, 5)! = "http:" &&
            HTMLMarkString. Mid(i, 4)! = "ftp:"))
        {
            //在中间加入服务器域名
            HTMLMarkStringLeft = HTMLMarkString. Left
(i);
            HTMLMarkStringRight = HTMLMarkString. Right
(HTMLMarkString. GetLength() - i);
            AnalyzeURL();
            HTMLMarkStringLeft + = NeatestURLWith-
outAppendedFileName;
            HTMLMarkString = HTMLMarkStringLeft +
HTMLMarkStringRight;
        }
        SubstituteFlag2 = 1;
    }
    //否则, HTML 标签不变
    MainHTMLFileTestSubString = HTMLMark-
String;
}
MainHTMLFileTestSubString = HTMLMarkString;
//否则, HTML 标签不变
}
else
{MainHTMLFileTestSubString = HTMLMarkString;
//否则, HTML 标签不变}
}

```

其他类型的标签,如超链接、背景图、音乐等,均可用极为类似的程序段来处理。

可能的读者会看到这样一个问题:有时相对寻址方式为,也即多出了一个"/"号。如果仍用上述程序段处理的话,其结果就成了,这样会多出一个"/",浏览器会不会“认识”呢?笔者曾亲自试验过,这种情况浏览器可以自动将其认为是http://www.tsinghua.edu.cn/image/a1.jpeg的正常形式(这是因为现在的浏览器多具有自动纠正URL中某些常见错误的功能),因此无需对此进行专门处理。

(来稿时间:1999年8月)