**ASP.NET中的多种path详解**

2009-08-13 11:44 zyip 博客园 字号：[T](javascript:setfont(12);) | [T](javascript:setfont(16);)

[一键收藏，随时查看，分享好友！](javascript:favorBox('open');)

在Web服务器端开发中经常会遇到各种不同形式的path，而且经常记不清楚如何获取到某种特定格式，今天抽了点时间把常用的path获取方法与格式做了一个简单的对照表，以备日后使用ASP.NET中的多种path查找。

AD：[51CTO 网+ 第十二期沙龙：大话数据之美\_如何用数据驱动用户体验](http://mobile.51cto.com/mobile/mdsa12/)

**ASP.NET中的多种path详解**

**Request获取的信息**

ASP.NET Path **~**/Views/Home/index.cshtml “**~**” 是指网站的根目录

Request.AppRelativeCurrentExecutionFilePath                  ~/SysOption/BillingSetup1.aspx

Request.ApplicationPath                                                 /

Request.CurrentExecutionFilePath                                   /SysOption/BillingSetup1.aspx

Request.FilePath                                                            /SysOption/BillingSetup1.aspx

Request.Path                                                                /SysOption/BillingSetup1.aspx

Request.PathInfo                                                          什么都没有（待测其他用法）

Request.PhysicalApplicationPath                                      D:\VssWorkFolder\British\_School\_MIS\src\WebSite\

Request.PhysicalPath                                                      D:\VssWorkFolder\British\_School\_MIS\src\WebSite\SysOption\BillingSetup1.aspx

Request.UserHostAddress                                              192.168.1.6

Request.UserHostName                                                  192.168.1.6

Request.Url.ToString()                                                    <http://192.168.1.6/SysOption/BillingSetup1.aspx?x=d>

Request.UrlReferrer                                                        空引用或者跳转源页面地址

Request.ServerVariables集合中获取到的相关信息:

左列是服务器变量名，右侧是值，值是通过Request.ServerVariables[服务器变量名]获取的

1. APPL\_MD\_PATH : /LM/W3SVC/894523/Root
2. APPL\_PHYSICAL\_PATH : D:\VssWorkFolder\British\_School\_MIS\src\WebSite\
3. INSTANCE\_META\_PATH : /LM/W3SVC/894523
4. LOCAL\_ADDR : 192.168.1.6
5. PATH\_INFO : /SysOption/BillingSetup1.aspx
6. PATH\_TRANSLATED : D:\VssWorkFolder\British\_School\_MIS\src\WebSite\SysOption\BillingSetup1.aspx
7. REMOTE\_ADDR : 192.168.1.6
8. REMOTE\_HOST : 192.168.1.6
9. SCRIPT\_NAME : /SysOption/BillingSetup1.aspx
10. SERVER\_NAME : 192.168.1.6
11. URL : /SysOption/BillingSetup1.aspx

Request.ServerVariables是一个很强大的工具，可以帮助我们获取很多client和web宿主的信息，有兴趣的朋友可以通过以下代码看看它到底包含什么信息

1. foreach (string s in Request.ServerVariables)
2. {
3. Response.Write(s + "  :  " + Request.ServerVariables[s] + "

6. }

**ASP.NET中的多种path转换**

1.转换为服务器端路径（Server.MapPath）

web服务器端开发设计一个有趣的问题就是，地址转换。比如http地址/images/a.txt，如果你想在服务器端通过io读取这个 文件，就得有这个文件的“本机地址（形如c:\windows\system32\xx.dll）”,这时Server.MapPath就很有用了

Response.Write(Request.MapPath(Request.Path));        输出为 D:\VssWorkFolder\British\_School\_MIS\src\WebSite\SysOption\BillingSetup1.aspx

2.转换为http地址（Page.ResolveClientUrl Page.ResolveUrl）

Response.Write(Page.ResolveClientUrl("~/a/a.jpg"));      输出为 ../a/a.jpg

Response.Write(Page.ResolveUrl("~/a/a.jpg"));              输出为 /a/a.jpg

ASP.NET中的多种path详解就介绍到这里。

## [ASP.NET中关于路径(Path)的相关知识](http://www.cnblogs.com/purplefox2008/archive/2010/05/06/1729019.html)

2010-05-06 16:37 by 追忆似水流年, 1162 阅读, 0 评论, [收藏](http://www.cnblogs.com/purplefox2008/archive/2010/05/06/1729019.html), [编辑](http://i.cnblogs.com/EditPosts.aspx?postid=1729019)

也许你曾遇到过这样的情况  
运行类似的代码

StreamReader sr=new StreamReader("Test.txt");

运行时会提示："Could not find file 'C:"windows"system32"Test.txt".  
这就是路径不对造成的  
幸好在asp.net的request对象中有许多与路径相关的属性

|  |  |
| --- | --- |
| ApplicationPath | 程序的虚拟根目录路径 |
| CurrentExecutionFilePath | 当前请求的虚拟路径 |
| FilePath |
| Path |
| PathInfo |
| PhysicalApplicationPath | 程序虚拟根目录的物理路径 |
| PhysicalPath | 请求URL的物理路径 |
| RawUrl | 当前请求的原始URL |
| Url | 当前请求的URL信息 |

例如 如果我有一个名为ASPNETC的网站 它是我网站根目录下的一个子目录C:\Webs   
输出如下：  
ApplicationPath: "/"  
CurrentExecutionFilePath: /aspnetc/testpath.aspx  
FilePath: /aspnetc/testpath.aspx  
Path: /aspnetc/testpath.aspx  
PathInfo:  
PhysicalApplicationPath: C:\webs\  
PhysicalPath: C:\webs\aspnetc\testpath.aspx  
RawUrl: /aspnetc/testpath.aspx  
  
RawUrl包含QueryString信息 PathInfo可以用来Url重写  
  
非常有用的 "~" and Page.ResolveUrl 两者结合起来能很方便的获取路径  
"~"代表当前应用程序的根目录，经常在控件中使用，用来定位图片，css等文件的位置。  
如果应用程序搬家了，放到不同的根目录下面，则会出现问题  
Page的ResolveUrl正是用来解决这个问题  
  
如果你的程序在"/SomeDir"  
Page.ResolveUrl("~/images/image1.jpg") 返回 "/Somedir/images/image1.jpg"  
在根目录  
Page.ResolveUrl("~/images/image1.jpg") will simply return "/images/image1.jpg"   
  
Server.MapPath方法相关  
Server.MapPath("/") Server.MapPath("~")得到当前应用程序根目录的物理路径  
Server.MapPath(".")得到当前目录的物理路径  
Server.MapPath("..")得到父目录的物理路径  
  
用Path.Combine连接路径

//Returns C:"testdir"images"image1.jpg  
Path.Combine(@"C:"testdir", @"Images"image1.jpg");

# NET中Path类的几个方法

1. Path.combine(string, string)  
     根据给出的两个路径, 返回一个路径.  
      例如:

     string CompletePath = System.IO.Path.Combine(@"c:\MyApp", @"Images\skyline.jpg");  
     将会返回一个全路径 c:\MyApp\Images\skyline.jpg  
     第一个参数中有无"\"结尾都可以.

2. Path.GetExtension(string)  
    返回给定文件路径的扩展名.例如:

http://www.cnblogs.com/Images/OutliningIndicators/None.gifstring FileExtention = System.IO.Path.GetExtention(@"C:\MyApp\Images\skyline.jpg");  
    将会返回 "jpg"  
  
3. Path.GetFileName(string)  
     给出文件名的全路径,返回文件名(包括扩展名).例如:

http://www.cnblogs.com/Images/OutliningIndicators/None.gifstring fileName = System.IO.Path.GetFileName(@"c:\MyApp\Images\skyline.jpg");  
    将会返回"skyline.jpg"

[ASP.NET中Request.ApplicationPath、Request.FilePath、Request.Path、.Request.MapPath](http://www.cnblogs.com/caishuhua226/p/4178692.html)

1.**Request.ApplicationPath**->当前应用的目录

2.**Request.FilePath**->对应于iis的虚拟目录   
  如 URL http://mockte.com/1/index.html/pathinfo   
  FilePath = /1/index.html   
  
3.**Request.Path**->当前请求的虚拟路径   
  Path 是 FilePath 和 PathInfo 尾部的串联。例如 URL http://mockte.com/1/index.html/pathinfo   
  那么Path = /1/index.html/pathinfo   
  
4.**Request.MapPath(string url)**->将url映射为iis上的虚拟目录   
  这个目录都是相对于application的根目录的   
  于Server.MapPath相比,不会包含类似c:/这样的路径   
  可以理解为是相对路径(对比的Server.MapPath就是绝对路径)   
  
5.**Server.MapPath(string url)**->将url映射为服务器上的物理路径   
  例如 http://mockte.com/1/index.html  假设你的应用程序在c:/iis/MySite中   
  那么就是 c:/iis/MySite/1/index.html

6.底下这张表就是根据Browser Request 的网址相关的属性的用法

|  |  |
| --- | --- |
| 网址：http://localhost:1897/News/Press/Content.aspx/123?id=1#toc | |
| Request.ApplicationPath | **/** |
| Request.PhysicalPath | **D:/Projects/Solution/web/News/Press/Content.aspx** |
| System.IO.Path.GetDirectoryName(Request.PhysicalPath) | **D:/Projects/Solution/web/News/Press** |
| Request.PhysicalApplicationPath | **D:/Projects/Solution/web/** |
| System.IO.Path.GetFileName(Request.PhysicalPath) | **Content.aspx** |
| Request.CurrentExecutionFilePath | **/News/Press/Content.aspx** |
| Request.FilePath | **/News/Press/Content.aspx** |
| Request.Path | **/News/Press/Content.aspx/123** |
| Request.RawUrl | **/News/Press/Content.aspx/123?id=1** |
| Request.Url.AbsolutePath | **/News/Press/Content.aspx/123** |
| Request.Url.AbsoluteUri | **http://localhost:1897/News/Press/Content.aspx/123?id=1** |
| Request.Url.Scheme | **http** |
| Request.Url.Host | **localhost** |
| Request.Url.Port | **1897** |
| Request.Url.Authority | **localhost:1897** |
| Request.Url.LocalPath | **/News/Press/Content.aspx/123** |
| Request.PathInfo | **/123** |
| Request.Url.PathAndQuery | **/News/Press/Content.aspx/123?id=1** |
| Request.Url.Query | **?id=1** |
| Request.Url.Fragment |  |
| Request.Url.Segments | **/** **News/** **Press/** **Content.aspx/** **123** |

# asp.net获取网站绝对路径示例

作者： 字体：[[增加](javascript:turnbig()) [减小](javascript:turnsmall())] 类型：转载 时间：2014-02-07 [我要评论](http://www.jb51.net/article/46528.htm#comments)

在编写 ASP.NET 应用程序的时候，有时为了更好地进行控制静态文件的路径，如在模板页或者用户控件中设置js或者css文件的路径等，采用绝对路径是难免的。下面就是几种获取绝对路径的几种方法

复制代码 代码如下:

VirtualPathUtility.ToAbsolute( " ~/ " )  
HttpRuntime.AppDomainAppVirtualPath  
Request.ApplicationPath  
Page.ResolveUrl( " ~ " )

以上代码生成的结果如下：  
当以网站的方式访问时，结果如下：

复制代码 代码如下:

VirtualPathUtility.ToAbsolute("~/") = /  
HttpRuntime.AppDomainAppVirtualPath = /  
Request.ApplicationPath = /  
Page.ResolveUrl("~") = /

当以虚拟目录（http://localhost:806/web2/url.aspx）访问时，结果如下：

复制代码 代码如下:

VirtualPathUtility.ToAbsolute("~/") = /web2/  
HttpRuntime.AppDomainAppVirtualPath = /web2  
Request.ApplicationPath = /web2  
Page.ResolveUrl("~") = /web2/

采用第二种和第三种方法，还需要做一下处理，因为网站访问时是以/结尾，而以虚拟目录访问时则是没有/的，进行还得进行一次判断，稍微麻烦一点。  
但是，这些方法在页面中使用是没有任何问题的，但是如果在 Global 的 Application\_Start 事件里需要得到网站的绝对路径，则需要使用前面2种方法，如果使用第三种方法，则会报告如下的错误：  
Request is not available in this context   
所以，只能采取头2种方法。例如

复制代码 代码如下:

void Application\_Start( object sender, EventArgs e)  
{  
    System.IO.StreamWriter s = new System.IO.StreamWriter(HttpRuntime.AppDomainAppPath + " log.txt " );  
    s.WriteLine(VirtualPathUtility.ToAbsolute( " ~/ " ));  
    s.WriteLine(HttpRuntime.AppDomainAppVirtualPath);  
    s.Close();  
}