

ASP.NET MVC动态二级域名及DNS（泛解析配置）

动态二级域名的实现：

应用场景：目前产品要实现SaaS功能，因为工作需要实现二级域名：[www.{CompanyUrl}.xxx.com](#)

假设产品主域名入口为：[www.xxx.com](#)

当a公司租户登录时：[www.a.xxx.com](#)

当b公司租户登录时：[www.b.xxx.com](#)

首先想到的是对Url的重写：（网上有关于UrlRewrite的实现。在ASP.NET中这也是常用的手法。）

Route简介：ASP.NET路由可以不用映射到网站特定文件的URL。由于该 URL 不必映射到文件，因此可以使用对用户操作进行描述因而更易于被用户理解的URL。 .NET Framework 3.5 SP1已经包含了ASP.NET Routing引擎。现在微软已经在ASP.NET WebForms 4.0中增加了对Routing引擎更好的支持，它使用表达式构造器进行双向Routing。

MVC 应用程序中的典型 URL 模式—来自MSDN

MVC 应用程序中用于路由的 URL 模式通常包括 {controller} 和 {action} 占位符。

当收到请求时，会将其路由到 [UrlRoutingModule](#) 对象，然后路由到 [MvcHandler](#) HTTP 处理程序。[MvcHandler](#) HTTP 处理程序通过向 URL 中的控制器值添加后缀“Controller”以确定将处理请求的控制器的类型名称，来确定要调用的控制器。URL 中的操作值确定要调用的操作方法。

MVC项目中添加路由，Global.asax 文件默认的MVC 路由的代码。

默认配置：

 View Code

涉及类参考

| 类 | 说明 |
|--------------------------------------|---|
| Route | 表示 Web 窗体或 MVC 应用程序中的路由。 |
| RouteBase | 用作表示 ASP.NET 路由的所有类的基类。 |
| RouteTable | 存储应用程序的路由。 |
| RouteData | 包含所请求路由的值。 |
| RequestContext | 包含有关对应于路由的 HTTP 请求的信息。 |
| RouteValueDictionary | 提供用于存储路由 Constraints 、 Defaults 和 DataTokens 对象的方法。 |
| VirtualPathData | 提供用于从路由信息生成 URL 的方法。 |

因为目前采用的是ASP.NET MVC 3进而可以利用扩展Route的方式实现。

首先定义DomainData、DomainRoute类

代码如下：

DomainRoute类：

```
1 public class DomainRoute : Route
2 {
3     private Regex domainRegex;
4     private Regex pathRegex;
5
6     public string Domain { get; set; }
7
8     public DomainRoute(string domain, string url, RouteValueDictionary defaults)
9         : base(url, defaults, new MvcRouteHandler())
10    {
11        Domain = domain;
12    }
13
14    public DomainRoute(string domain, string url, RouteValueDictionary defaults, IRouteHandler routeHandler)
15        : base(url, defaults, routeHandler)
16    {
17        Domain = domain;
18    }
19
20    public DomainRoute(string domain, string url, object defaults)
21        : base(url, new RouteValueDictionary(defaults), new MvcRouteHandler())
```

```
22     {
23         Domain = domain;
24     }
25
26     public DomainRoute(string domain, string url, object defaults, IRouteHandler routeHandler)
27         : base(url, new RouteValueDictionary(defaults), routeHandler)
28     {
29         Domain = domain;
30     }
31
32     public override RouteData GetRouteData(HttpContextBase httpContext)
33     {
34         // 构造 regex
35         domainRegex = CreateRegex(Domain);
36         pathRegex = CreateRegex(Url);
37
38         // 请求信息
39         string requestDomain = httpContext.Request.Headers["host"];
40         if (!string.IsNullOrEmpty(requestDomain))
41         {
42             if (requestDomain.IndexOf(":") > 0)
43             {
44                 requestDomain = requestDomain.Substring(0, requestDomain.IndexOf(":"));
45             }
46         }
47         else
48         {
49             requestDomain = httpContext.Request.Url.Host;
50         }
51         string requestPath = httpContext.Request.AppRelativeCurrentExecutionFilePath.Substring(2) +
httpContext.Request.PathInfo;
52
53         // 匹配域名和路由
54         Match domainMatch = domainRegex.Match(requestDomain);
55         Match pathMatch = pathRegex.Match(requestPath);
56
57         // 路由数据
58         RouteData data = null;
59         if (domainMatch.Success && pathMatch.Success)
60         {
61             data = new RouteData(this, RouteHandler);
62
63             // 添加默认选项
64             if (Defaults != null)
65             {
66                 foreach (KeyValuePair<string, object> item in Defaults)
67                 {
68                     data.Values[item.Key] = item.Value;
69                 }
70             }
71
72             // 匹配域名路由
73             for (int i = 1; i < domainMatch.Groups.Count; i++)
74             {
75                 Group group = domainMatch.Groups[i];
76                 if (group.Success)
77                 {
78                     string key = domainRegex.GroupNameFromNumber(i);
79
80                     if (!string.IsNullOrEmpty(key) && !char.IsNumber(key, 0))
81                     {
82                         if (!string.IsNullOrEmpty(group.Value))
83                         {
84                             data.Values[key] = group.Value;
85                         }
86                     }
87                 }
88             }
89
90             // 匹配域名路径
91             for (int i = 1; i < pathMatch.Groups.Count; i++)
92             {
93                 Group group = pathMatch.Groups[i];
94                 if (group.Success)
95                 {
96                     string key = pathRegex.GroupNameFromNumber(i);
97
98                     if (!string.IsNullOrEmpty(key) && !char.IsNumber(key, 0))
99                     {
100                         if (!string.IsNullOrEmpty(group.Value))
```

[illegible]

DomainData 类:

```
1 public class DomainData
2     {
```

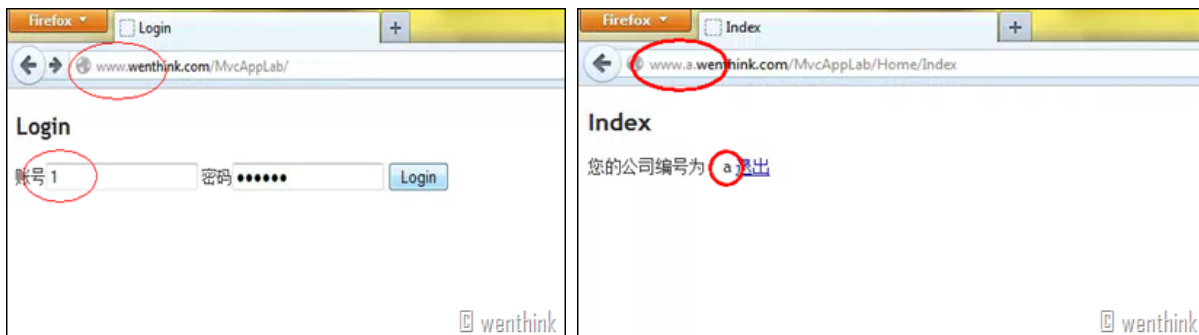
```
3     public string Protocol { get; set; }
4     public string HostName { get; set; }
5     public string Fragment { get; set; }
6 }
```

然后在Global中配置路由

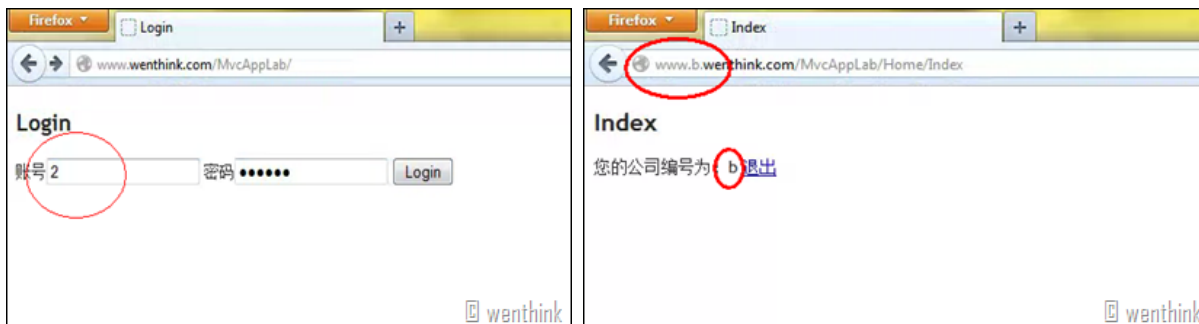
```
1: routes.Add("DomainRoute", new DomainRoute(
2:     "www.{companyUrl}.wenthink.com",    // Domain with parameters
3:     "{controller}/{action}/{id}",      // URL with parameters
4:     new { companyUrl = "", controller = "Home", action = "Index", id = "" } // Parameter defaults
5: ));
```

效果图:

a 公司用户登录时:



b 公司用户登录时:



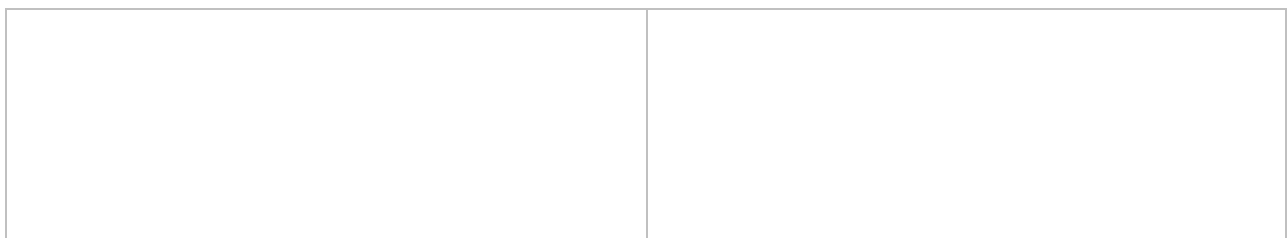
当用户尝试错误时: 可自定义提示页面

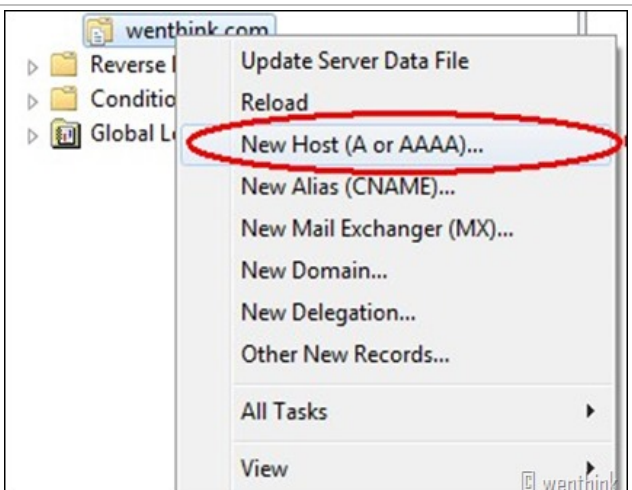
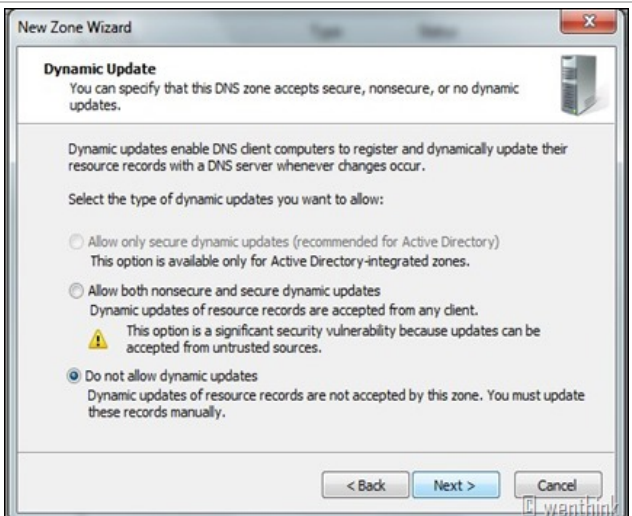
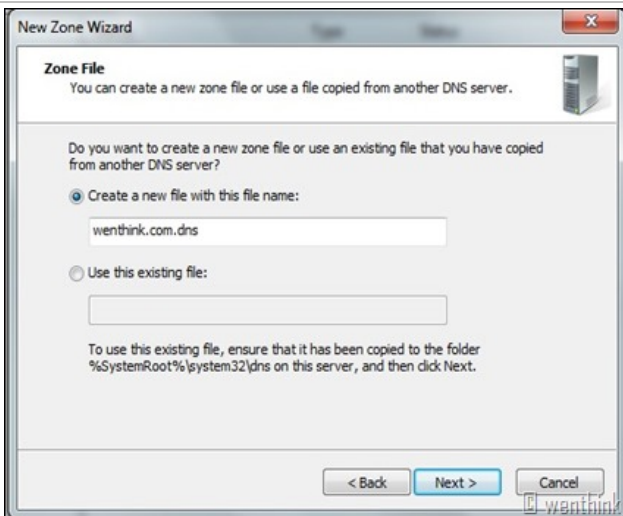
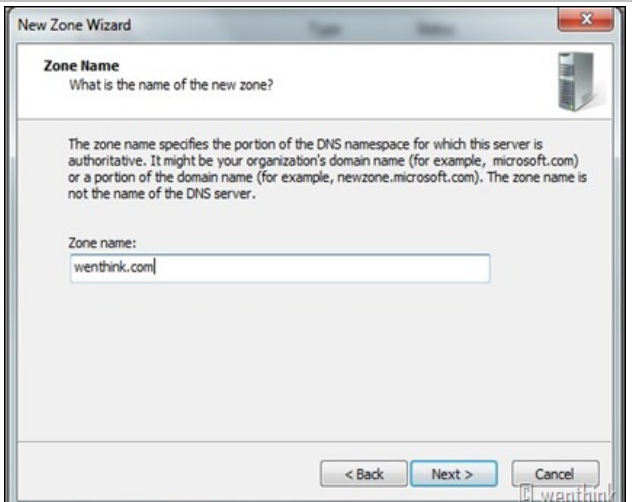
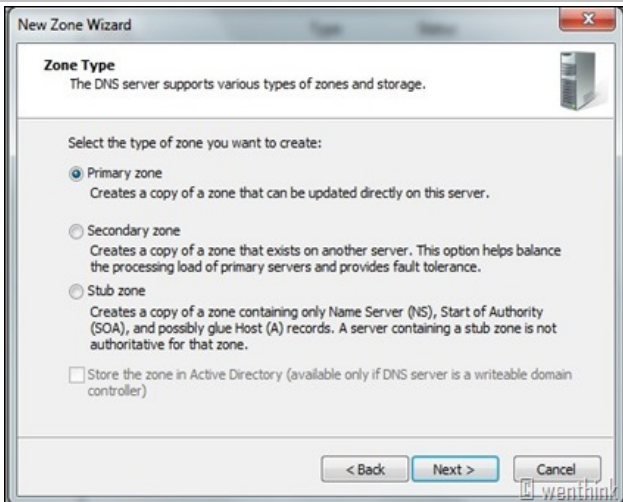
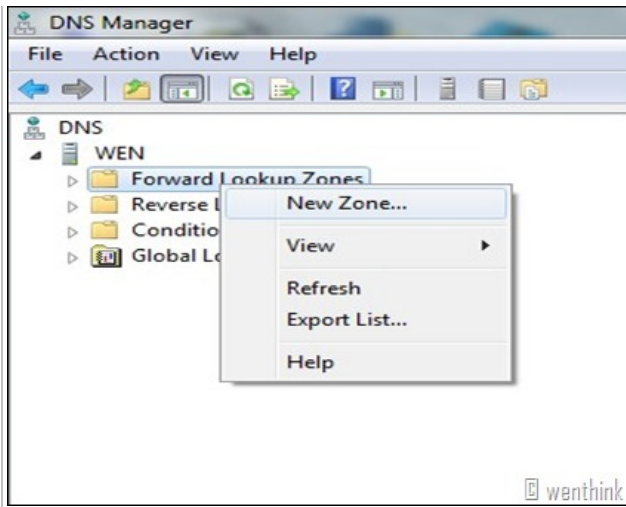


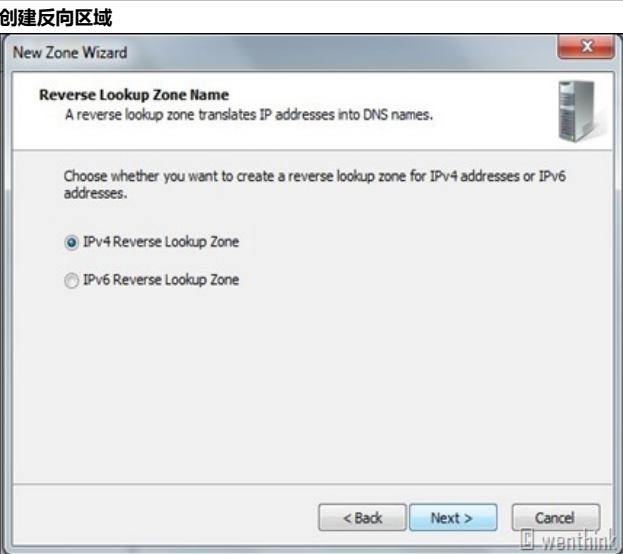
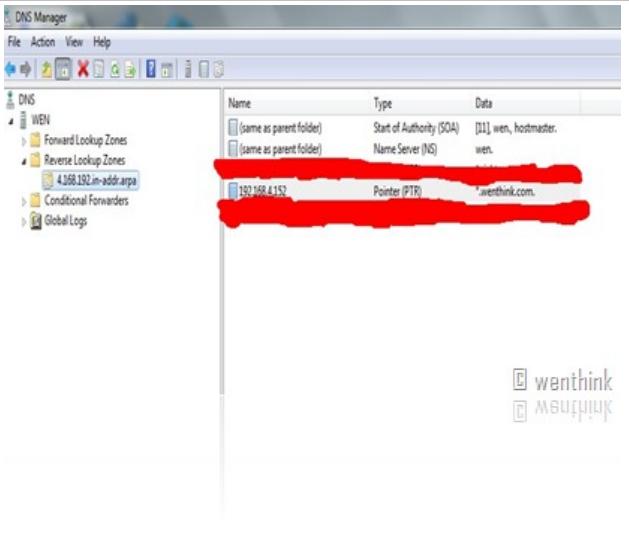
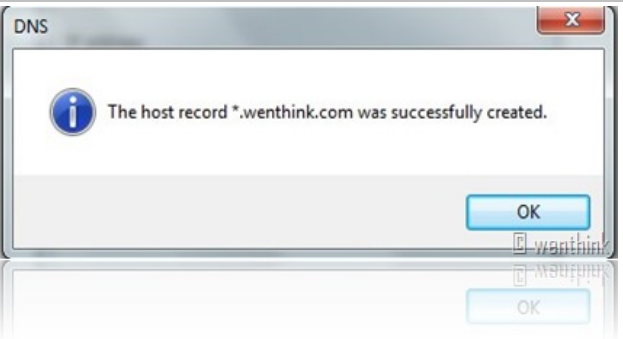
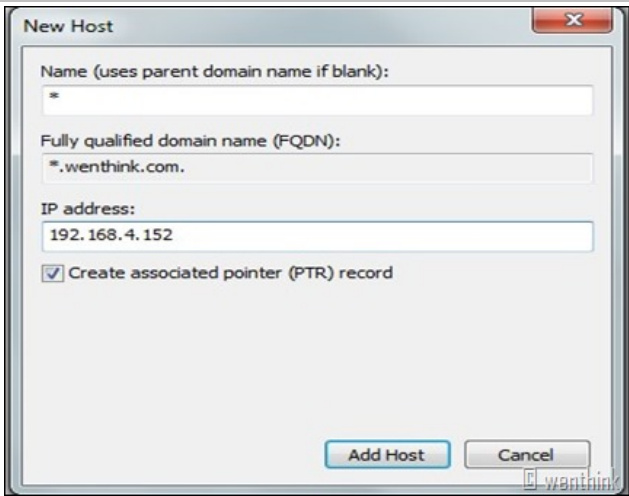
基础支持: 域名支持泛解析

然后要做的是: 配置DNS服务, 也就是让你的域名支持泛解析 😊 (Windows Server 才会有, 其他的Windows系统可以尝试修改Host文件, 便于测试)

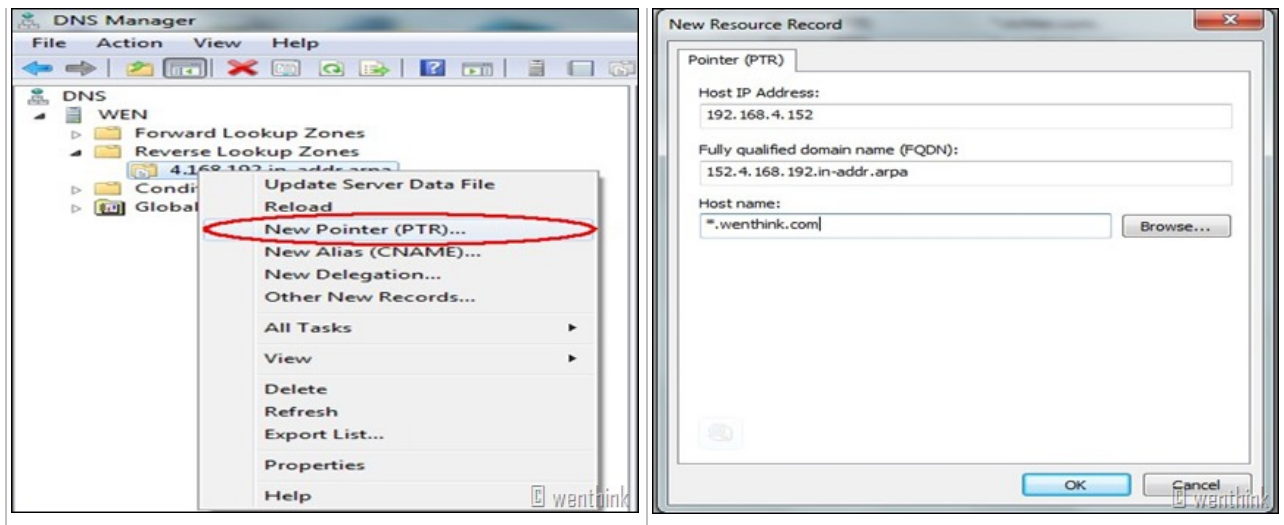
Step By Step







绑定指针



完成上面的操作, 基本可以实现DNS的泛解析了. 当然如果没有绑定域名的话, 只能修改Host文件来进行操作;

本机测试的情况下需要把Host文件中添加当前IP地址 所映射的域名, 如本文中的wenthink.com

备注: 以上解决了域名访问的问题, 但是会存在Session跨域访问的丢失的现象. 本例中采用了CrossDomainCookie的方式解决. 当然不是唯一的解决方案. 方法很简单, 这里就不多说明了, 需要的可以自己Google一下 :-)

参考资料: <http://blog.maartenballiauw.be/post/2009/05/20/ASPNET-MVC-Domain-Routing.aspx>

<http://www.cnblogs.com/cygl162/archive/2010/10/15/1851773.html>

以上为个人学习摘要, 如有错误, 欢迎指正!!

下期: ASP.NET管道机制及IIS的工作原理浅析.

上个草图先 :-P

作者: [wenthink](#)

出处: <http://www.cnblogs.com/wenthink/archive/2013/04/10/MvcDynamicSecondaryDomain.html>

本文版权归【wenthink】和博客园共有, 欢迎转载, 但未经作者同意必须保留此段声明, 且在文章页面明显位置给出原文连接, 否则保留追究法律责任的权利。

我们每个人都是梦想家，当离开家了，就只剩梦想了。。。道路，或许坎坷，但你终会到达！！！！

分类: [ASP.NET](#), [个人随笔](#)

标签: [MVC](#), [二级域名](#), [DNS泛解析](#)

绿色通道：

好文要顶

关注我

收藏该文

与我联系

 [wenthink](#)
[关注 - 34](#)
[粉丝 - 182](#)

荣誉：[推荐博客](#)
[+加关注](#)

« 上一篇：[否极博客，正式开通！](#)
» 下一篇：[瀚海拾贝（一）HTTP协议/IIS 原理及ASP.NET运行机制浅析【图解】](#)

posted @ 2013-04-10 13:36 [wenthink](#) 阅读(3902) 评论(23) [编辑](#) [收藏](#)

评论列表

- #1楼 2013-04-10 14:23 [Gamain](#)

“

学习了，非常感谢分享。

”

[支持\(0\)](#) [反对\(0\)](#)
- #2楼 2013-04-10 15:12 [浩然天地](#)

“

好有才华。。好牛x

”

[支持\(0\)](#) [反对\(0\)](#)
- #3楼 2013-04-10 16:30 [林悠扬](#)

“

^0^ ^0^ ^0^ smile.....赞一个。。。！！

”

[支持\(0\)](#) [反对\(0\)](#)
- #4楼 2013-04-10 16:30 [2604529](#)

“

霸气。。学习了

”

[支持\(0\)](#) [反对\(0\)](#)
- #5楼 2013-05-07 13:50 [StevenChennet](#)

“

专门来顶的

”

[支持\(0\)](#) [反对\(0\)](#)
- #6楼[楼主] 2013-05-07 15:09 [wenthink](#)

“

@Gamain
@StevenChennet
感谢支持!!! :-)

”

[支持\(0\)](#) [反对\(0\)](#)
- #7楼 2013-05-28 22:43 [永远的阿哲](#)

“

学习了

”

[支持\(0\)](#) [反对\(0\)](#)
- #8楼 2013-05-31 17:39 [2604529](#)

“

有源码嘛

”

[支持\(0\)](#) [反对\(0\)](#)
- #9楼[楼主] 2013-05-31 19:16 [wenthink](#)

“

@2604529
[引用](#)
有源码嘛

”

核心源码已经贴在博客中了.主要就是DNS泛解析配置. ^_^

[支持\(0\)](#) [反对\(0\)](#)

#10楼 2013-06-01 15:49 [zhyhunan](#)

“

学习了

”

[支持\(0\)](#) [反对\(0\)](#)
- http://www.cnblogs.com/wenthink/archive/2013/04/10/MvcDynamicSecondaryDomain.html
- 8/10

- #11楼

2013-06-03 09:09 2604529

@wenthink

跳转页面好了 但是我传值后取值

public ActionResult Main(int id)

{

@ViewBag.id = id;

return View();

}

页面@ViewBag.id 取不到值 想知道楼主登录后a b是怎么输出的

支持(0)

反对(0)
- #12楼[楼主]

2013-06-03 23:26 wenthink

@2604529

囧.刚刚翻了半天没翻到源码....可能忘记备份了的.当时记得是从Route里面读的.好像发给老王过一份.你问问.时间长了记不清了.

支持(0)

反对(0)
- #13楼

2013-06-30 15:15 要有好的心情

学习

支持(0)

反对(0)
- #14楼

2014-01-16 15:37 埋头前进的码农

IIS怎么配置呢？难道是不用改变IIS的配置？这样IIS会根据主机头重定向到了不同的网站上了！

支持(0)

反对(0)
- #15楼[楼主]

2014-01-24 11:11 wenthink

@李飞麟

引用

IIS怎么配置呢？难道是不用改变IIS的配置？这样IIS会根据主机头重定向到了不同的网站上了！

时间长了，记不大清除了。首先，部署的时候，会需要你填写一个网址，比如本文 wenthink.com; 然后修改host 文件，127.0.0.1 wenthink.com 这样就可以了。希望对你有帮助！ :-)

支持(0)

反对(0)

#16楼

2014-03-26 11:19 soulfree

@wenthink



您好我按照您所说的进行配置了，进行域名解析访问的时候出现：

支持(0)

反对(0)

#17楼[楼主]

2014-03-28 00:01 wenthink

@soulfree

引用

@wenthink

您好我按照您所说的进行配置了，进行域名解析访问的时候出现：

这个提示是IIS 不支持虚拟目录的意思，建议检查下设置：)

支持(0)

反对(0)

#18楼

2014-05-14 15:49 李寒斌

楼主，请教您一个问题？我在本地host文件配置了.example.com、www.example.com，然后路由添加 routes.Add("DomainRoute2", new DomainRoute(

http://www.cnblogs.com/wenthink/archive/2013/04/10/MvcDynamicSecondaryDomain.html

9/10

```
"home.example.com", // Domain with parameters
"{action}/{id}", // URL with parameters
new { controller = "Home", action = "Index", id = "" } // Parameter defaults
});
当我home.example.com+端口访问时，没有正确解析，抛出无法访问。
```

支持(0) 反对(0)

#19楼 2014-07-04 19:20 Dodu.Net

@李寒斌
你host没有配置 home.example.com,跟定无法正确解析

支持(0) 反对(0)

#20楼 2014-10-23 20:21 Hsuwen

求份源码，谢谢楼主，还不是不太清楚该怎么用，mr.hsuwen@qq.com

支持(0) 反对(0)

#21楼[楼主] 2014-10-25 01:32 wenthink

@Hsuwen
引用
求份源码，谢谢楼主，还不是不太清楚该怎么用，mr.hsuwen@qq.com

实在不好意思，源码没有保留。核心代码都已经贴出。理解：）

支持(0) 反对(0)

#22楼 2014-10-27 09:22 Hsuwen

@wenthink
您好，非常感谢您的回复。方便请教您几个问题吗？我在做一个类似博客一样的网站，每个用户都有个个人空间，希望完成的效果是为每个用户的个人空间的地址是以个人帐号为二级域名（就像博客园一样，比如您的帐号是wenthink，你在博客园的空间地址是http://wenthink.cnblogs.com/）。我有自己配置的其他路由规则，使用的是默认方法。我现在希望新建一个Zone的Area，去实现这个个人空间，我使用了您公开的DomainRoute类，添加到了Global的第一个（并保留了之前的路由配置），

```
1 routes.Add("DomainRoute", new DomainRoute(
2     "www.{user}.xxx.com", // Domain with parameters
3     "{controller}/{action}/{token}", // URL with parameters
4     new { user= @"[A-Za-z0-9]{3,12}",area="Zone",controller = "Home", action = "Index", token= string.Empty } // Parameter defaults
5 ));
```

在Area='zone'中有/Home/Index的Action，在默认的下面也有一个Home的控制，我用RouteDebug测试，访问 username.xxx.com能匹配到这个扩展的路由，但是会提示我匹配到了多个HomeController,我试着在这个"DomainRoute"路由的匿名类里添加Namespaces约束，但是用RouteDebug调试，显示此项没有添加任何的数据Token。
请问，我该如果实现希望的效果呢？方面回复下QQ，请教一下您吗？感激不尽！

支持(0) 反对(0)

#23楼 2015-04-27 15:37 Engineer

请教一下，这样会不会有跨域的问题，还有session共享的问题。

支持(0) 反对(0)