# StaticHtml使用教程

首先请将StaticHtml.dll添加引用，或者拷贝到bin目录中，下面是具体配置说明

注意：下面的配置步骤都是在web.config中操作。

## 添加StaticHtml配置文件读取节点

在configuration/configSections添加如下节点

<section name="staticHtml" type="StaticHtml.StaticHtmlSection,StaticHtml"/>

添加StaticHtml

## 添加HttpModule

在configuration/system.web/httpModules中，添加

<add name="staticHtml" type="StaticHtml.HttpModule,StaticHtml"/>

如果这样配置，用浏览器访问任意一个页面，如果根目录下没有生成staticHtml\_log.txt这个文件，请将上面的节点，添加到configuration/system.webServer/httpModule中

**当然，请确保你服务器能正确加载httpmodule，因为有可能你的服务器配置，根本就不能加载任何httpmodule。请参考asp.net mvc配置**

## 配置自己站点的生成缓存规则

在configuration中添加如下节点

<staticHtml skip=”admin/”>

<rule name=" index ">

<patten type="StaticHtml.RegexPatten,StaticHtml" pars="RegPatten= index \.aspx"/>

<store type="StaticHtml.MemStore,StaticHtml"/>

<genKey type="StaticHtml.UrlMd5GenKey,StaticHtml"/>

<expire type="StaticHtml.TimeExpire,StaticHtml” pars=”Second=300”/>

</rule>

<rule name="content">

<patten type="StaticHtml.RegexPatten,StaticHtml" pars="RegPatten=article/"/>

<store type="StaticHtml.FileStore,StaticHtml" pars="Path=cacheHtml\_article/"/>

</rule>

</staticHtml>

上面的配置，定义了2个规则，

第一个规则，

<rule name=" index ">

<patten type="StaticHtml.RegexPatten,StaticHtml" pars="RegPatten=index\. aspx "/>

<store type="StaticHtml.MemStore,StaticHtml"/>

</rule>

规则，这个规则的名字叫做index,

patten节点 如果url中匹配index\. aspx 这个正则表达式，则缓存。

store节点 将缓存的html保存到内存中

genKey 节点 将request.rawRul通过md5加密生成key，在statichtml非常重要，代表了一个唯一的缓存页面，

expire节点 意味着缓存300秒，5分钟

第二个规则

<rule name="content">

<patten type="StaticHtml.RegexPatten,StaticHtml" pars="RegPatten=article/"/>

<store type="StaticHtml.FileStore,StaticHtml" pars="Path=cacheHtml\_article/"/>

</rule>

这个规则的名字叫做content（规则名字不能重复）

patten 节点 如果url中匹配article/这个正则表达式的，就缓存

store节点 将缓存的保存到根目录下面/cacheHtml\_article中

StaticHtml节点 skip属性 admin/ 代表如果url能匹配这个正则表达式，则不缓存。

提示：staticHtml节点skip 属性，patten节点的 pars属性， 都是可以配置正则表达式的， 就是一个纯粹的正则表达式，可以写的很复杂， 也可以很简单。

## 其他补充

在staticHtm中rule节点实际上有5个节点可以配置patten，store，genKey，expire，genHtml

patten没有默认值，必须配置

Store如果不配置，将是

<store type="StaticHtml.FileStore,StaticHtml" pars=”Path=cacheHtml/”/>

genKey如果不配置，将是

<genKey type="StaticHtml.UrlMd5GenKey,StaticHtml"/>

Expire如果不配置，将是

<expire type="StaticHtml.TimeExpire,StaticHtm” pars=”Second=300”/>

genHtml一般不要配置

也许您也发现了，配置节点中type实际上就是一个**类**的说明，staticHtml使用反射加载这个类，并设置pars中的参数信息，比如这个

<expire type="StaticHtml.TimeExpire,StaticHtm” pars=”Second=300”/>

系统将通过Activator.CreateInstance("StaticHtml","StaticHtm.TimeExpire”)加载StaticHtml程序集中StaticHtm.TimeExpire类，然后根据pars中的值，设置这个类的属性，

比如这里，就会将这个类的Second设置为300，pars如果有多个属性，使用;分号分开。注意参数都是通过字符串设置的，如果你的属性是不是字符串类型，请自己处理。

也就是说，您完全可以自己定义各个配置节点的类，当然了需要实现staticHtml库中的对应的接口，如果想自定义store节点，那么就应该是想staticHtml.IStore接口。每个接口实际上都非常简单。