**[优化网站设计（八）：将脚本文件和样式表作为外部文件引用](http://www.cnblogs.com/chenxizhang/archive/2013/05/01/3053562.html)**

**前言**

网站设计的优化是一个很大的话题,有一些通用的原则,也有针对不同开发平台的一些建议。这方面的研究一直没有停止过，我在不同的场合也分享过这样的话题。

作为通用的原则，雅虎的工程师团队曾经给出过35个最佳实践。这个列表请参考

Best Practices for Speeding Up Your Web Site  <http://developer.yahoo.com/performance/rules.html>

同时，他们还发布了一个相应的测试工具Yslow <http://developer.yahoo.com/yslow/>

我强烈推荐所有的网站开发人员都应该学习这些最佳实践，并结合自己的实际项目情况进行应用。

接下来的一段时间，我将结合ASP.NET这个开发平台，针对这些原则，通过一个系列文章的形式，做些讲解和演绎，以帮助大家更好地理解这些原则，并且更好地使用他们。

**准备工作**

为了跟随我进行后续的学习，你需要准备如下的开发环境和工具

1. Google Chrome 或者firefox ，并且安装 Yslow这个扩展组件.请注意，这个组件是雅虎提供的，但目前没有针对IE的版本。
   1. <https://chrome.google.com/webstore/detail/yslow/ninejjcohidippngpapiilnmkgllmakh>

Technorati Tags: [Performance](http://technorati.com/tags/Performance),[Web design](http://technorati.com/tags/Web+design),[ASP.NET](http://technorati.com/tags/ASP.NET)

* 1. <https://addons.mozilla.org/en-US/firefox/addon/yslow/>
  2. 你应该对这些浏览器的开发人员工具有所了解，你可以通过按下F12键调出这个工具。

1. Visaul Studio 2010 SP1 或更高版本，推荐使用Visual Studio 2012
   1. <http://www.microsoft.com/visualstudio/eng/downloads>
2. 你需要对ASP.NET的开发基本流程和核心技术有相当的了解，本系列文章很难对基础知识做普及。

**本文要讨论的话题**

这一篇我和大家讨论第八个原则：[Make JavaScript and CSS External](http://developer.yahoo.com/performance/rules.html#external) （将脚本文件和样式表作为外部文件引用）

我承认，这是一个有点争议的话题。这里的争议点并不在于要不要将脚本文件和样式表作为外部文件引用（因为通常情况下，大家都同意这样做是有好处的），而是在于如何把握一个度。因为在某些情况下，确实将脚本或者样式定义在页面中可能更好一些。

我这里就整理一下大家对于这个做法的优缺点分析

**好处：**

1. 提高了脚本文件和样式表的复用性。（无需在每个页面中都定义一次）
2. 减小了页面体积，可以提高页面加载速度。（脚本文件和样式表可以被浏览器单独缓存）
3. 提高了脚本和样式的可维护性。（这个虽然与性能无关，但其实也是很重要的）

**缺点：**

1. 因为有单独的文件，所以可能会增加额外的请求。这违背了 [Make Fewer HTTP Requests](http://developer.yahoo.com/performance/rules.html#num_http) 这个原则
   * 但由于这些文件可以被单独缓存，所以实际上这个的影响不见得那么大

很显然，**采纳这条原则的好处通常是多于坏处的**。尤其是如果你懂得结合如下几条原则的话：

1. [Use a Content Delivery Network (CDN)](http://developer.yahoo.com/performance/rules.html#cdn)
2. [Add Expires or Cache-Control Header](http://developer.yahoo.com/performance/rules.html#expires)
3. [Gzip Components](http://developer.yahoo.com/performance/rules.html#gzip)
4. [Put Stylesheets at Top](http://developer.yahoo.com/performance/rules.html#css_top)
5. [Put Scripts at Bottom](http://developer.yahoo.com/performance/rules.html#js_bottom)
6. [Minify JavaScript and CSS](http://developer.yahoo.com/performance/rules.html#minify)
7. [Remove Duplicate Scripts](http://developer.yahoo.com/performance/rules.html#js_dupes)
8. [Split Components Across Domains](http://developer.yahoo.com/performance/rules.html#split)

但是，确实有的时候，我们不会采纳这条原则。下面这篇文章做了比较好的介绍 ：<http://www.thefutureoftheweb.com/blog/when-to-use-inline-javascript>

文章说到，在下面三种情况下应该考虑直接在页面中定义脚本和样式

1. 这些脚本和样式是专用于某个页面的（换句话说，它们不需要复用）
2. 这些页面并不经常访问（这句话的潜台词是，既然不经常访问，那么页面体积大一些，甚至访问慢一些也无伤大雅）
3. 这些脚本和样式很少（我觉得这是一条最关键的）

[](http://images.cnitblog.com/blog/9072/201305/01203828-3cf0bb04cf884892a5d8998a40a72737.png)