

5. 前端的性能优化

雅虎14军规

1. 尽可能的减少 HTTP 的请求数	content
2. 使用 CDN (Content Delivery Network)	server
3. 添加 Expires 头(或者 Cache-control)	server
4. Gzip 组件	server
5. 将 CSS 样式放在页面的上方	css
6. 将脚本移动到底部 (包括内联的)	javascript
7. 避免使用 CSS 中的 Expressions	css
8. 将 JavaScript 和 CSS 独立成外部文件	javascript css
9. 减少 DNS 查询	content
10. 压缩 JavaScript 和 CSS (包括内联的)	javascript css
11. 避免重定向	server
12. 移除重复的脚本	javascript
13. 配置实体标签 (ETags)	css
14. 使 AJAX 缓存	

此外还有base64、精灵图、svg

1. 可以合并文字信息、图片信息、脚本文件和样式文件，减少http请求
2. 使用CDN，将用户的请求重新导向离用户近的服务器节点上。使用户能就近的访问所需内容，解决网络拥挤的状况，提高用户访问网站的相应速度
3. 添加Exprire/Cache-Control头，简单说就是服务器在返回请求的时候发了一个时间值，而本机则是通过这个时间值来判断时候重新加载页面，没有过期就还是用这个资源
4. 使用Gzip压缩，将文件在服务器上进行压缩，然后再传输
5. 将Css放在页面的最上面，为了避免页面显示空白或者闪错的问题，我们应该将Css样式放在页面的头部head下
- 6、将script放在页面的最下面
- 7、避免使用Css Expressions (CSS表达式)
Css Expressions这个共能就是在CSS样式中插入JS代码，他不符合web标准、效率低下、有可能带来安全隐患
- 8、将脚本文件和样式文件都放在外部文件中

这个说法并不是绝对的，下面是将CSS单独提取出来和写在页面里面的优缺点。这两个方法我们要学会取舍着用。



9. 减少DNS查找，既然是查找必定消耗时间的，这段时间浏览器显示一片空白

10、最小化JavaScript和Css，减小文件体积

减小文件体积的方法

- 去除不必要的空白符、格式符、注释符
- 简写方法名、参数压缩JS脚本

11、避免重定向，增加了浏览器到服务器的之间的往返次数，增加了访问时间，301表示永久的重定向，302表示临时的重定向

12、移除重复的脚本

13、配置实体标签ETag（使用特殊的字符串来标识某个资源版本），一致时就会使用以前的资源，减轻了服务器的负担

14、使用Ajax缓存（Asynchronous JavaScript and XML）

Ajax可以使网站在不刷新页面的情况下加载数据，可以使网站分批加载，实现局部更新

ajax的两种方法，Post和get

Post

Post请求时每次都要发给服务器请求的，服务器每次都会返回一个状态码200。因为这个方法是每次都需要执行的，所以他是不被缓存的。

Get

get请求，除非制定不同的地址，否则同一个地址不重复执行。它返回的状态码是304。所以他的数据是被缓存的。

GET	POST
把参数数据队列加到提交表单的ACTION属性所指的URL中，值和表单内各个字段一一对应，在URL中可以看到。	通过HTTP POST机制，将表单内各个字段与其内容放置在HTML HEADR内一起传送到ACTION属性所指的URL地址。用户看不到这个过程。
服务器端用Request.QueryString获取变量的值。	服务器端用Request.Form获取提交的数据。
传送的数据量较小，不能大于2KB	传送的数据量较大，一般被默认为不受限制。但理论上，因服务器的不同而异。
安全性非常低。	安全性较高。
<form method="get" action="a.asp?b=b"> 跟<form method="get" action="a.asp"> 是一样的，也就是说，method为get时 action页面后边带的参数列表会被忽视。	<form method="post" action="a.asp?b=b"> 跟<form method="post" action="a.asp"> 是不一样的。
它会将数据添加到URL中，通过这种方式传递到服务器，通常利用一个问号?代表URL地址的结尾与数据参数的开端，后面的参数每一个数据参数以“名称=值”的形式出现，参数与参数之间利用一个连接符&来区分。	数据是放在HTTP主体中的，其组织方式不只一种，有&连接方式，也有分隔符方式，可隐藏参数，传递大批数据，比较方便。