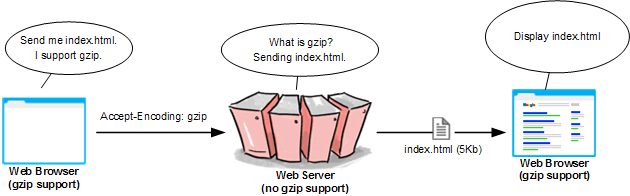
**[翻译] gzip 压缩原理**

**浏览器请求有无 gzip 压缩的高级预览**

**服务器未 gzip：**



**浏览器：**

* 连接服务器并请求页面。
* 通知服务器，浏览器支持 gzip “Accept-Encoding: gzip”。

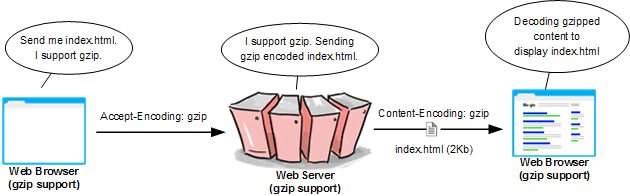
**服务器：**

* 不支持 gzip。忽略 gzip 请求。发送未压缩的页面。

**浏览器：**

* 接收页面。
* 显示页面。

**服务器 gzip：**



**浏览器：**

* 连接服务器。
* 通知服务器，浏览器支持 gzip “Accept-Encoding: gzip”。

**服务器：**

* 收到 gzip 支持通知。
* 发送 gzip 编码过的页面，并设置响应头“Content-Encoding: gzip”。

**浏览器：**

* 接收页面。
* 根据响应头“Content-Encoding: gzip” 解码 gzip 编码过的页面。
* 显示页面。

**如何进行 gzip 压缩**

简单的说，gzip 压缩是在一个文本文件中找出类似的字符串，并临时替换他们，使整个文件变小。这种形式的压缩对 web 来说特别适合，因为 HTML 和 CSS文件通常包含大量的重复字符串，例如空格，标签，及样式定义。

**例子**

在这个例子中，我将用一小段代码演示 gzip 压缩中用相同的标签跟不同的标签的对比。

在第一段代码中，我用了5个标题标签。

未压缩: 69 bytes

压缩后: 85 bytes (译注：某天[77](http://blog.gulu77.com)问我怎么压缩后比压缩前还大？我回来后看了下原文并测试了一下，事实如此，并非翻译错)

<h1>One</h1><h2>Two</h2><h3>Three</h3><h4>Four</h4><h5>Five</h5>

把标题标签改为相同的 div 标签，源代码增大了 10 bytes，但是压缩后，它缩小到 66 bytes，比先前的源代码片段还小！

未压缩: 79字节

压缩后: 66 字节

<div>One</div><div>Two</div><div>Three</div><div>Four</div><div>Five</div>

许多开发者可能依此只使用 div 和 span 标签而使页面更大程度的压缩。在多数情况下当然无可非议，但必须注意到正确的语义化标记对可访问性（accessibility）来说非常重要，而且使页面更容易阅读。即便如此，可以使用此方法做一些优化，这取决于你，开发者来决定哪些是合适的。

英文地址：

[code.google.com/intl/zh-CN/speed/articles/gzip.html](http://code.google.com/intl/zh-CN/speed/articles/gzip.html)

最小负载：

[code.google.com/intl/zh-CN/speed/page-speed/docs/payload.html#GzipCompression](http://code.google.com/intl/zh-CN/speed/page-speed/docs/payload.html#GzipCompression)