性能

## HTTP/2

Michael Gooding 为什么最新版本的HTTP将会是一个大的突破。

HTTP/2已经到来，其目标很简单：提高现有HTTP协议的性能。通俗的说，就是更快的加载网页。新的标准有许多功能来改进现有网页的使用问题。我接下来将一一阐述。

多路复用

多路复用允许同时通过单一的连接发起多重的请求-响应消息，解决了HTTP/1.1中串行阻塞问题。当前，客户端需要等待服务器响应上一个请求后才能发送下一个请求。如果一个请求暂时阻塞了，整个页面的下载就会延迟。HTTP/2 是基于二进制帧的协议，请求和响应会分解称为帧（frame）的块，帧携带了所属的请求/响应的元信息。这就允许资源可以并行传输而不产生冲突，并且可以以任何次序发送/接收。由此，初始的请求可能花稍长的时间得到响应，但是不再会阻塞后续资源的下载，意味着页面能够更快的呈现。

头部压缩

HTTP头部变得越来越大。通过组合使用查找表和霍夫曼编码，HTTP/2能够把请求头中发送的字节数降低为0。在通常的网页会话中，平均90%的压缩率并不罕见。

在HTTP/1.1中，请求一个包含75个资源（图片，脚本，样式等），外加500字节头部信息的普通页面，需要花费4次TCP周期。HTTP/2 没有优化响应的大小，但是在请求部分的改进却是显著的。得益于相同参数的复用和90%的压缩率，HTTP/2可以在一次TCP周期中发送所有的请求。

依赖关系与优先级

普通浏览器都引入了预加载器，以确保他们优先请求最重要的资源。但讽刺的是，在新模式下，如果浏览器只是简单的请求所有资源并且只关心服务器的响应速度的话，页面性能反而会降低，因为页面渲染的关键资源可能不会被优先传输。因此HTTP/2的设计者们指出：通过告知服务器各部分资源的依赖关系并列出优先级，服务器就可以确保关键数据得以正确传输。

服务器端推送

服务器端推送可以让服务器在接收到客户端请求之前主动向浏览器推送数据，即使是在最差的条件下也可以保证页面即时传输。然而，为了保证传输数据的正确性同时避免带宽浪费，服务器需要知道客户端下一步需要的是什么以及当前浏览器的缓存状态。不过目前HTTP/2已经为服务器端推送提供了可用工具，所以我相信在未来几年会涌现出很多创新性的使用方法。

当前使用者需要做什么

对于当前使用者来说，完全无需更改网站或者应用就可以保持正常工作。因为HTTP/2是与现有代码和接口文档相兼容的，不过你仍然需要做好准备开始应用TLS协议了（如果你们还没开始使用的话）。

而对于第三方组织来说，他们需要开始想一想如何利用HTTP/2提供的性能优势和安全能力了。

*\**扩展阅读：*netm.ag/http2-271*

简介：Akamai的前开发者Michael (@Michael\_G\_81)目前主要负责探索性能瓶颈以及帮助世界上一些大型网站解决复杂问题。