关于http/2 基础信息的了解，优点，使用方式，使用条件，调试和性能优化，http/2和https

<https://imququ.com/>

# 一、基础信息

1、http/2简称h2或者h2c(非加密连接)。

2、1999年http1.1至2015年首次更新（14年）

3、主要基于[SPDY](https://zh.wikipedia.org/wiki/SPDY" \o "SPDY)协议。（[SPDY](https://zh.wikipedia.org/wiki/SPDY)：一种[开放](https://zh.wikipedia.org/wiki/%E9%96%8B%E6%94%BE%E5%8E%9F%E5%A7%8B%E7%A2%BC" \o "开放源代码)的[网络传输协议](https://zh.wikipedia.org/wiki/%E7%B6%B2%E8%B7%AF%E5%82%B3%E8%BC%B8%E5%8D%94%E5%AE%9A" \o "网络传输协议)，由[Google](https://zh.wikipedia.org/wiki/Google" \o "Google)开发，用来发送网页内容。基于[传输控制协议](https://zh.wikipedia.org/wiki/%E4%BC%A0%E8%BE%93%E6%8E%A7%E5%88%B6%E5%8D%8F%E8%AE%AE" \o "传输控制协议)（TCP）的[应用层](https://zh.wikipedia.org/wiki/%E5%BA%94%E7%94%A8%E5%B1%82" \o "应用层)协议）

4、开发小组：一个组织IETF，一个工作小组httpbis.

IETF: 互联网工程任务小组，负责[互联网](https://zh.wikipedia.org/wiki/%E4%BA%92%E8%81%94%E7%BD%91)标准的开发和推动

组织形式： 负责特定议题的工作组（主席和副主席），工作组再用主题组织为领域（领域指导和副指导）

5、审批组织：IESG: [Internet Engineering Steering Group](https://zh.wikipedia.org/w/index.php?title=IESG&action=edit&redlink=1)

是由AD和IETF主席构成，负责IETF的整体运作。

6、HTTP/2的标准化工作由[Chrome](https://zh.wikipedia.org/wiki/Google_Chrome)、[Opera](https://zh.wikipedia.org/wiki/Opera%E7%80%8F%E8%A6%BD%E5%99%A8)、[Firefox](https://zh.wikipedia.org/wiki/Firefox)[[6]](https://zh.wikipedia.org/wiki/HTTP/2#cite_note-6)、[Internet Explorer 11](https://zh.wikipedia.org/wiki/Internet_Explorer_11)、[Safari](https://zh.wikipedia.org/wiki/Safari)、[Amazon Silk](https://zh.wikipedia.org/w/index.php?title=Amazon_Silk&action=edit&redlink=1)及[Edge](https://zh.wikipedia.org/wiki/Microsoft_Edge)等浏览器提供支持。

7、根据[W3Techs](https://zh.wikipedia.org/w/index.php?title=W3Techs&action=edit&redlink=1)的数据，在2017年5月，在排名前一千万的网站中，有13.7%支持了HTTP/2。

# 二、协议的制定（h2做了什么事情）

https://zh.wikipedia.org/wiki/HTTP/2#.E5.8D.8F.E8.AE.AE.E7.9A.84.E5.88.B6.E5.AE.9A

1、创建一个协商协议标准，即应用层协议协商（ALPN），以便客户端能够从http/1.0，http/1.1、http/2乃至其他非http协议中做出选择。

2、与HTTP/1.1在请求方法、状态码乃至URI和绝大多数HTTP头部字段等方面保持高度兼容性。

3、通过以下举措，减少 网络延迟，提高浏览器的页面加载速度

1）对HTTP头字段进行数据压缩(即HPACK算法)

<http://www.jianshu.com/p/f44b930cfcac>

https://my.oschina.net/CandyDesire/blog/592446

2）HTTP/2 服务端推送(Server Push) <http://www.jianshu.com/p/16429243c868>

3）请求管线化

<https://zh.wikipedia.org/wiki/HTTP%E7%AE%A1%E7%B7%9A%E5%8C%96>

4）修复HTTP/1.0版本以来未修复的 [队头阻塞](https://zh.wikipedia.org/wiki/%E9%98%9F%E5%A4%B4%E9%98%BB%E5%A1%9E) 问题

5）对数据传输采用[多路复用](https://zh.wikipedia.org/wiki/%E5%A4%9A%E8%B7%AF%E5%A4%8D%E7%94%A8)，让多个请求合并在同一 [TCP](https://zh.wikipedia.org/wiki/TCP) 连接内

4、支持现有的HTTP应用场景，包括桌面和移动设备浏览器，网络API，不同规格的[网络服务器](https://zh.wikipedia.org/wiki/%E7%BD%91%E7%BB%9C%E6%9C%8D%E5%8A%A1%E5%99%A8)和[正向代理](https://zh.wikipedia.org/w/index.php?title=%E6%AD%A3%E5%90%91%E4%BB%A3%E7%90%86&action=edit&redlink=1)、[反向代理](https://zh.wikipedia.org/wiki/%E5%8F%8D%E5%90%91%E4%BB%A3%E7%90%86)服务器软件，[防火墙](https://zh.wikipedia.org/wiki/%E9%98%B2%E7%81%AB%E5%A2%99)，CDN等。

# 三、协议之间的比较

1、HTTP/2 和HTTP/1.1比较

1）HTTP/2 相比 HTTP/1.1 的修改并不会破坏现有程序的工作，但是新的程序可以借由新特性得到更好的速度。

2）HTTP/2 保留了 HTTP/1.1 的大部分语义，例如[请求方法](https://zh.wikipedia.org/wiki/%E8%B6%85%E6%96%87%E6%9C%AC%E4%BC%A0%E8%BE%93%E5%8D%8F%E8%AE%AE#.E8.AF.B7.E6.B1.82.E6.96.B9.E6.B3.95)、[状态码](https://zh.wikipedia.org/wiki/%E8%B6%85%E6%96%87%E6%9C%AC%E4%BC%A0%E8%BE%93%E5%8D%8F%E8%AE%AE#.E7.8A.B6.E6.80.81.E7.A0.81)乃至[URI](https://zh.wikipedia.org/wiki/URI)和绝大多数[HTTP头部](https://zh.wikipedia.org/w/index.php?title=HTTP%E5%A4%B4%E9%83%A8&action=edit&redlink=1)字段一致。而 HTTP/2 采用了新的方法来编码、传输客户端——服务器间的数据。

### 2、HTTP/1.1与SPDY的区别

[SPDY](https://zh.wikipedia.org/wiki/SPDY) (发音为"speedy") 是一个由 [Google](https://zh.wikipedia.org/wiki/Google) 主导的研究项目发明的HTTP替代协议。[[13]](https://zh.wikipedia.org/wiki/HTTP/2#cite_note-extremetech-13)SPDY一开始主要关注降低延迟，采用了TCP通道，但是使用了不同的协议来达到此目的。其与HTTP/1.1相比，主要的改变有：[[14]](https://zh.wikipedia.org/wiki/HTTP/2#cite_note-Grigorik-14)

* 实现无需先入先出的[多路复用](https://zh.wikipedia.org/wiki/%E5%A4%9A%E8%B7%AF%E5%A4%8D%E7%94%A8)
* 为简化客户端和服务器开发的消息—帧机制
* 强制性压缩（包括HTTP头部）
* 优先级排序
* 双向通讯

### 3、HTTP/2与SPDY的比较

HTTP/2的开发基于SPDY进行跃进式改进。在诸多修改中，最显著的改进在于，HTTP/2使用了一份经过定制的压缩算法，基于[霍夫曼编码](https://zh.wikipedia.org/wiki/%E9%9C%8D%E5%A4%AB%E6%9B%BC%E7%BC%96%E7%A0%81)，以此替代了SPDY的动态流压缩算法，以避免对协议的Oracle攻击——这一类攻击以[CRIME](https://zh.wikipedia.org/wiki/CRIME)为代表。此外，HTTP/2禁用了诸多加密包，以保证基于TLS的连接的前向安全。

# 四、新特性

### 1、帧、消息、流和TCP连接

### 流编码 ===》 TCP通道

协议将每个请求分区为二进制的控制帧与数据帧部分，以便解析。

这个举措在SPDY中的实践表明，相比HTTP/1.1，新页面加载可以加快11.81% 到 47.7%

Message

Message

F

### 2、HPACK 算法

用于对HTTP头部做压缩

原理：两个table一个coding。

### 3、服务器推送

即服务端向客户端发送比客户端请求更多的数据。

# 五、浏览器支持

1、截至2015年末，主要的浏览器的最新版本已经支持HTTP/2这一协议。[[8]](https://zh.wikipedia.org/wiki/HTTP/2#cite_note-browser_support-8)其中：

* [Google Chrome](https://zh.wikipedia.org/wiki/Google_Chrome)、[Mozilla Firefox](https://zh.wikipedia.org/wiki/Mozilla_Firefox)、[Microsoft Edge](https://zh.wikipedia.org/wiki/Microsoft_Edge)和[Opera](https://zh.wikipedia.org/wiki/Opera)已支持HTTP/2，并默认启用。
* [Internet Explorer](https://zh.wikipedia.org/wiki/Internet_Explorer)自[IE 11](https://zh.wikipedia.org/wiki/IE_11)开始支持HTTP/2，并预设激活

### 2、h2c 的支持度：被主流浏览器废弃。

### 3、SPDY 退出历史舞台

2015年9月，Google 宣布了计划，移除对SPDY的支持，拥抱 HTTP/2[[24]](https://zh.wikipedia.org/wiki/HTTP/2#cite_note-24)，并将在Chrome 51中生效。

# 六、有故事的协议

### 1、参考：<https://zh.wikipedia.org/wiki/HTTP/2>

# 七、h2与性能优化

资料汇总：https://imququ.com/post/http2-resource.html

开始使用h2: <https://imququ.com/post/use-http2-now.html>

[使用 nghttp2 调试 HTTP/2 流量](https://imququ.com/post/intro-to-nghttp2.html)

[性能优化一](https://imququ.com/post/http2-and-wpo-1.html)

[性能优化二](https://imququ.com/post/http2-and-wpo-2.html)

[性能优化三](https://imququ.com/post/http2-and-wpo-3.html)

[机遇与挑战PPT](https://www.dropbox.com/s/4duv6cqrhud4qzw/HTTP2%EF%BC%9A%E6%96%B0%E7%9A%84%E6%9C%BA%E9%81%87%E4%B8%8E%E6%8C%91%E6%88%98.pdf?dl=0)

[基于 HTTP/2 的 WEB 内网穿透实现](https://imququ.com/post/tunnel-to-localhost-base-on-http2.html)

[http/2简介（google dev）](https://developers.google.com/web/fundamentals/performance/http2/?hl=zh-cn)

配置http/2环境：<https://juejin.im/entry/57f3a546da2f60004f6eebbc>

分享思路：

1. 介绍一下http1.x的问题和解决方案
2. spdy，http2.0的前身
3. 介绍一下http2的基本信息 https://developers.google.com/web/fundamentals/performance/http2/?hl=zh-cn#\_1
4. 有了http2是怎么实现性能优化的，并且依然需要保留的优化方案。和可以舍弃的优化方案。

http2的优化：头部压缩，server push，多路复用

使用http2的调试方式。

队首阻塞

服务器推送：<http://www.alloyteam.com/2017/01/http2-server-push-research/>

http://www.alloyteam.com/2017/04/guide-http2-server-push-part1/

<http://www.alloyteam.com/2017/04/guide-http2-server-push-part2/>

**https://segmentfault.com/a/1190000009084692#articleHeader5**

3、http/1.1 和http/2的性能对比效果。<https://http2.akamai.com/demo> （隐身模式打开）

4、目前使用了http/2的网站：

https://www.byvoid.com/zhs/blog/http-keep-alive-header