SUNING
苏宁金融

HTTP 技术分享

金融研发技术管理中心

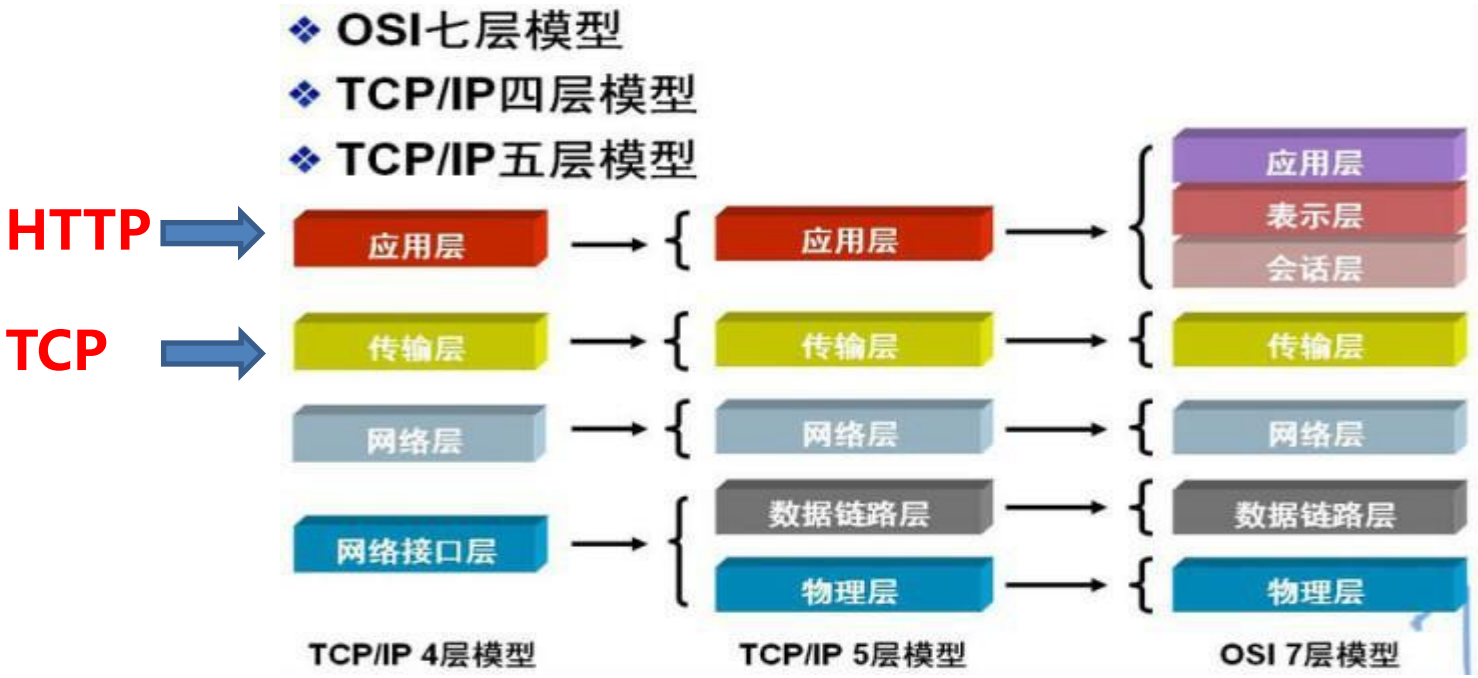
目录

HTTP 简介

SUNING
苏宁金融

HTTP协议是Hyper Text Transfer Protocol（超文本传输协议）的缩写,是用于从万维网（WWW:World Wide Web ）服务器传输超文本到本地浏览器的传送协议。

- 简单快速、灵活、无连接、无状态。
- HTTP使用**统一资源标识符**（Uniform Resource Identifiers, URI）来传输数据和建立连接。URL是一种特殊类型的URI，包含了用于查找某个资源的足够的信息。



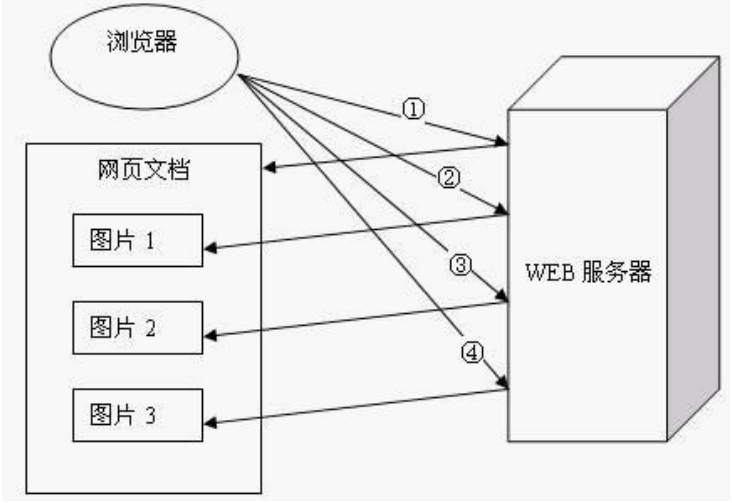
HTTP 1.0 与 HTTP 1.x

◆ **HTTP 1.1支持长连接（PersistentConnection） 和 请求的流水线（Pipelining）处理**

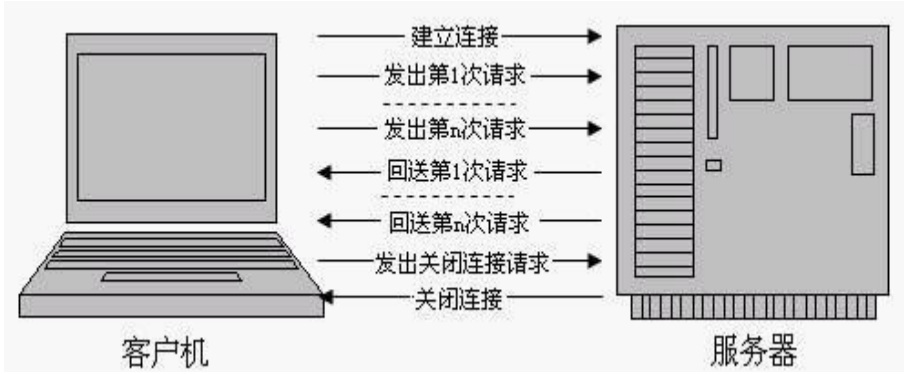
HTTP 1.0规定浏览器与服务器只保持**短暂的连接**，浏览器的每次请求都需要与服务器建立一个TCP连接，服务器完成请求处理后立即断开TCP连接，服务器不跟踪每个客户也不记录过去的请求。

HTTP 1.1支持持久连接，在一个TCP连接上可以传送多个HTTP请求和响应，减少了建立和关闭连接的消耗和延迟。

在一个TCP连接上可以传送多个HTTP请求和响应，减少了建立和关闭连接的消耗和延迟。



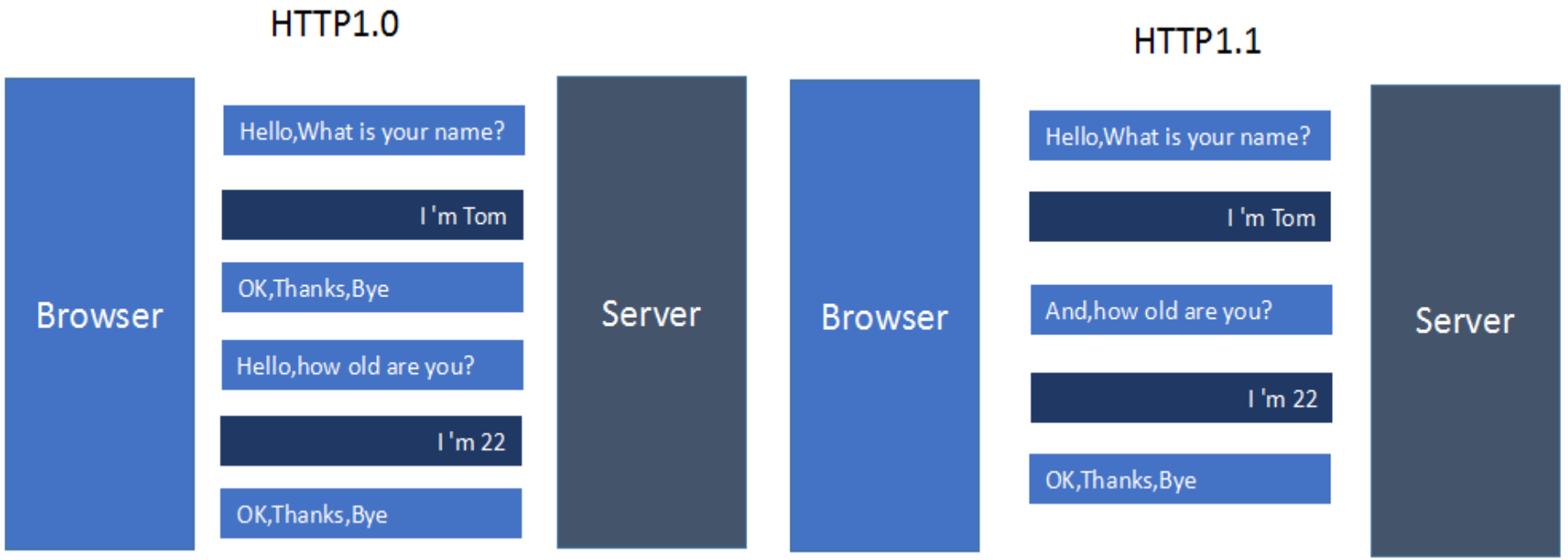
HTTP 1.0



HTTP 1.1

HTTP1.0 和 HTTP1.1 对比

SUNING
苏宁金融



HTTP 1.1扩展性增强

◆ **HTTP 1.1增加host字段**

在HTTP1.0中认为每台服务器都绑定一个唯一的IP地址，HTTP1.1在Request消息头里头多了一个Host域，比如：

```
GET /pub/WWW/TheProject.html HTTP/1.1
Host: www.w3.org
```

◆ **在请求消息中引入了range头域，支持传送内容的一部分、**断点续传

在响应消息中Content-Range头域，声明返回的这部分对象的偏移值和长度

◆ **HTTP/1.1加入了一个新的状态码100（Continue） (节约带宽) 。新 增了24个状态响应码**

◆ **HTTP/1.1中引入了Chunked transfer-coding**

发送方将消息分割成若干个任意大小的数据块，每个数据块在发送时都会附上块的长度，最后用一个零长度的块作为消息结束的标志。这种方法允许发送方只缓冲消息的一个片段，避免缓冲整个消息带来的过载。1.0通过Content-Length来给出消息结束标志

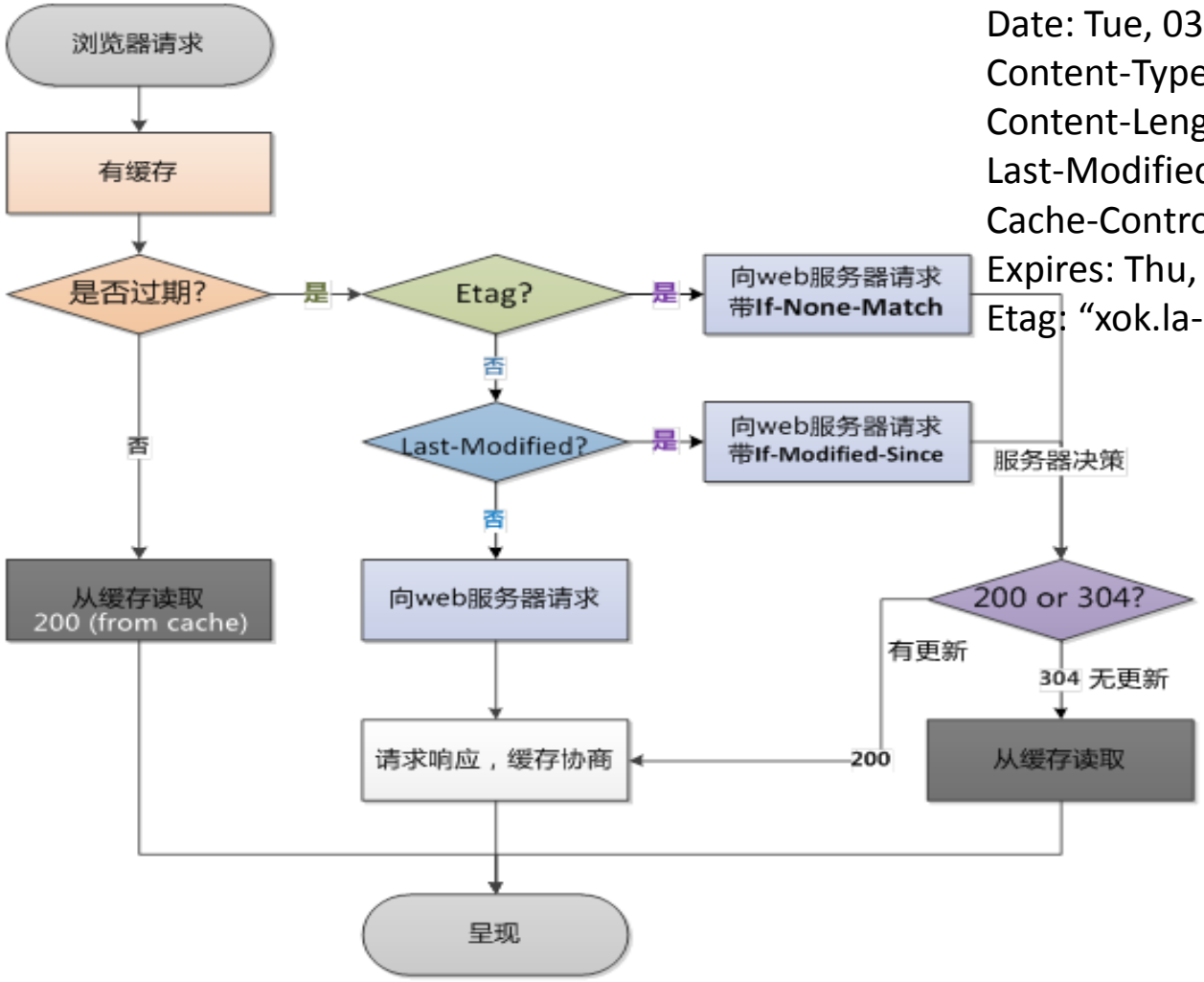
◆ **HTTP/1.1在1.0 Expires（过期时间）的加入了一些cache的新特性，Cache-Control头域**

max-age=[秒] 、public、no-cache、no-store、must-revalidate、Etag/ If-None-Match

HTTP 1.1 缓存机制

SUNING
苏宁金融

- 浏览器再次请求时：




```
HTTP/1.1 200 OK
Date: Tue, 03 Mar 2009 04:58:40 GMT
Content-Type: image/jpeg
Content-Length: 83185
Last-Modified: Mon, 22 Nov 2010 16:29:24 GMT
Cache-Control: max-age=2592000
Expires: Thu, 02 Apr 2009 05:14:08 GMT
Etag: “xok.la-961AA72-4CEA99B4415628”
```

HTTP 1.0 主要问题

◆ **连接无法复用**

导致每次请求都经历三次握手和慢启动。三次握手在高延迟的场景下影响较明显，慢启动则对文件类大请求影响较大。HTTP1.x在传输数据时，每次都需要重新建立连接，无疑增加了大量的延迟时间，特别是在移动端更为突出。

◆ **head of line blocking**

导致带宽无法被充分利用，以及后续健康请求被阻塞。

◆ HTTP1.0在传输数据时，所有传输的内容都是明文

◆ HTTP1.0在使用时，header里携带的内容过大

◆ keep-alive使用多了会给服务端带来大量的性能压力

HTTP 1.1 解决连接无法复用

◆ Connection:Keep-Alive

到http1.1之后Connection的默认值就是Keep-Alive。一段时间内的连接复用对PC端浏览器的体验帮助很大，因为大部分的请求在集中在一小段时间内。

◆ **基于tcp的长链接**

移动端app 建立一条自己的长链接通道，通道的实现是基于tcp协议。http短连接模式会频繁的创建和销毁连接。

◆ **http long-polling**

移动端app 建立一条自己。

◆ **http streaming**

server并不会结束初始的streaming请求，而是持续的通过这个通道返回最新的业务数据。

◆ **web socket**

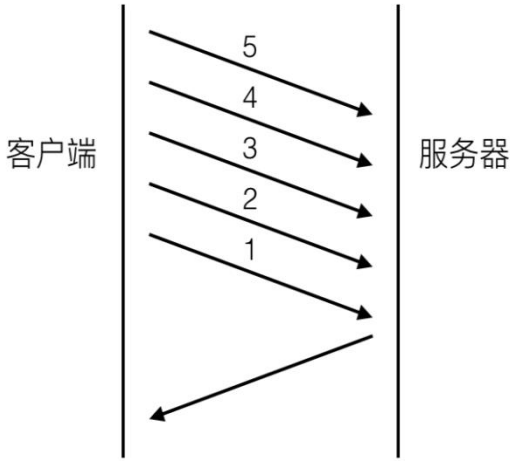
基于tcp协议，提供双向的数据通道。优势在于提供了message的概念，比基于字节流的tcp socket使用更简单，同时又提供了传统的http所缺少的长连接功能。

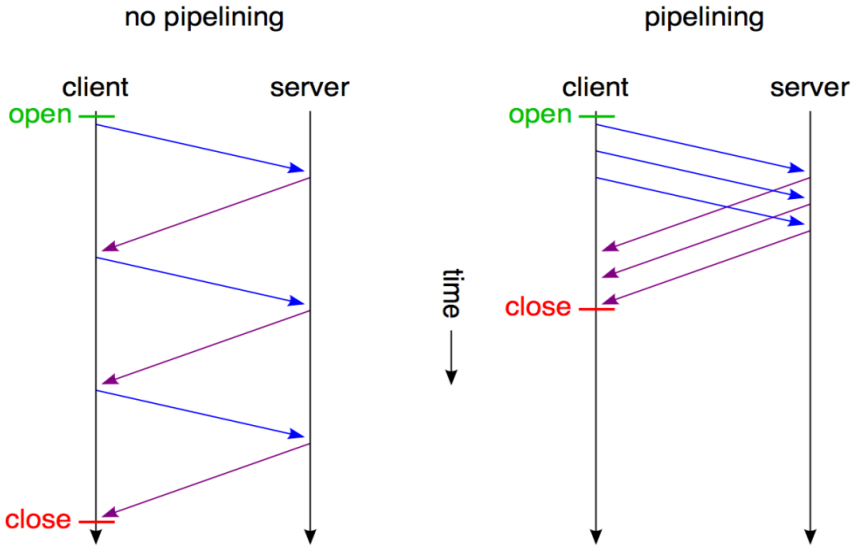
HTTP 1.1解决head of line blocking

SUNING 苏宁金融

◆ **http pipelining**

请求2，3，4，5不用等请求1的response返回之后才发出，而是几乎在同一时间把request发向了服务器。2，3，4，5及所有后续共用该连接的请求节约了等待的时间，极大的降低了整体延迟





缺点：

- pipelining只能适用于http1.1，支持http1.1的server都要求支持pipelining。只有幂等的请求（GET，HEAD）能使用pipelining，非幂等请求比如POST不能使用，因为请求之间可能会存在先后依赖关系。
- head of line blocking并没有完全得到解决，server的response还是要求依次返回，遵循FIFO(first in first out)原则。**串行的文件传输；连接数过多**。
- 绝大部分的http代理服务器不支持pipelining。
- 和不支持pipelining的老服务器协商有问题。

目录

- HTTP 1.0/1.1
- 开拓者SPDY
- 救世主HTTP2.0

SPDY协议

SUNING
苏宁金融

SPDY为speedy（单词原意：快速的）的缩写。2012 年Google提出的基于传输控制协议(TCP)的应用层协议，通过压缩、多路复用和优先级来缩短加载时间。该协议是一种更加快速的内容传输协议。

SPDY并不是一种用于替代HTTP的协议，而是对HTTP协议的增强。新协议的功能包括数据流的多路复用、请求优先级，以及HTTP包头压缩。谷歌已经开发一个网络服务器原型机，以及支持SPDY协议的Chrome浏览器版本。

SPDY对比HTTP的优势：

复用连接，可在一个TCP连接上传送多个资源。应对了TCP慢启动的特性。
请求分优先级，重要的资源优先传送。
HTTP头部数据也被压缩，省流量。
服务器端可主动连接客户端来推送资源（Server Push）。

缺点：

单连接会因TCP线头阻塞（head-of-line blocking）的特性而传输速度受限。加上存在可能丢包的情况，其负面影响已超过压缩头部和优先级控制带来的好处。

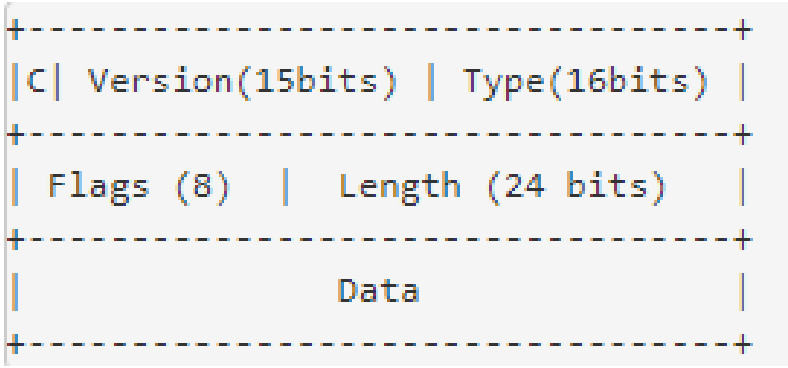
SPDY介绍

SUNING 苏宁金融

SPDY把一次单向传输（服务器到客户端或客户端到服务器）的内容称作**帧（frame）**，按协议组装帧内容称为**装帧（framing）**。帧内容分为头部（header）和载荷（payload），类似于HTTP的头部（header）和实体（entity），但有以下区别：

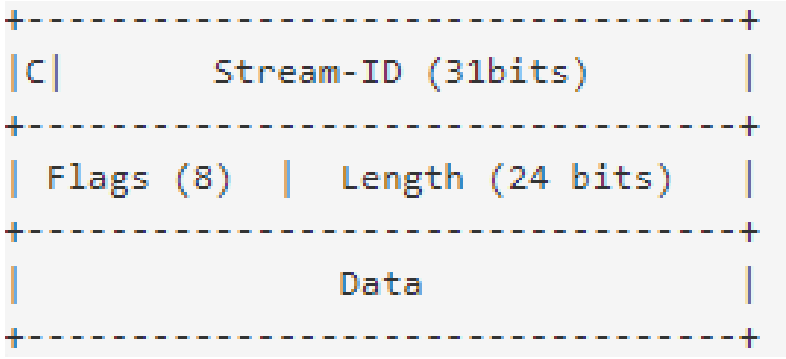
- SPDY的头部都是8个字节，根据其中一些位的数值不同来表示不同的信息，并把HTTP的头部放到SPDY的载荷里。
- HTTP的实体（除POST信息外）是文件数据（data），SPDY的载荷除了可以是文件数据还可以是其它信息。根据载荷的内容，帧分为控制帧和数据帧。

控制帧的数据格式：



SYN_STREAM：打开流;
SYN_REPLY：远程确认新打开的流;
RST_STREAM：关闭流;

数据帧的数据格式：



SPDY把一次HTTP Request/Response来回称作流（Stream）

基本概念介绍

SUNING 苏宁金融

◆ TLS 为安全传输层协议（**Transport Layer Security**）

用于在两个通信应用程序之间提供保密性和数据完整性。该协议由两层组成： TLS 记录协议（TLS Record）和 TLS 握手协议（TLS Handshake）。较低的层为 TLS 记录协议，位于某个可靠的传输协议（例如 TCP）上面，与具体的应用无关，所以，一般把TLS协议归为传输层安全协议。TLS 是独立于应用协议。高层协议可以透明地分布在 TLS 协议上面。

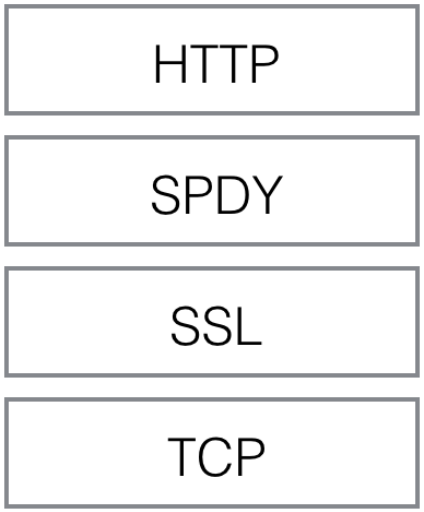
◆ Next Protocol Negotiation (NPN)

NPN是一个使SPDY在TLS服务器上对使用何种应用层协议进行协商的协议。NPN简单来说就是在**TLS的握手阶段**增加一些字段来表明服务器端和客户端希望**在TLS基础上**使用HTTP之外的（SPDY）协议。 NPN同样是Google提出的，为SPDY铺路。

Client端程序的实现是：握手前对OpenSSL（或封装它的库）设置可接受哪些协议，握手后获取服务器选择了哪个协议，然后按选择的协议进行通信。

SPDY解决哪些问题

- 降低延迟，客户端的单连接单请求，server的FIFO响应队列都是延迟的大头。
- http最初设计都是客户端发起请求，然后server响应，server无法主动push内容到客户端。
- 压缩http header，http1.x的header越来越膨胀，cookie和user agent很容易让header的size增至1kb大小，甚至更多。而且由于http的无状态特性，header必须每次request都重复携带，很浪费流量。



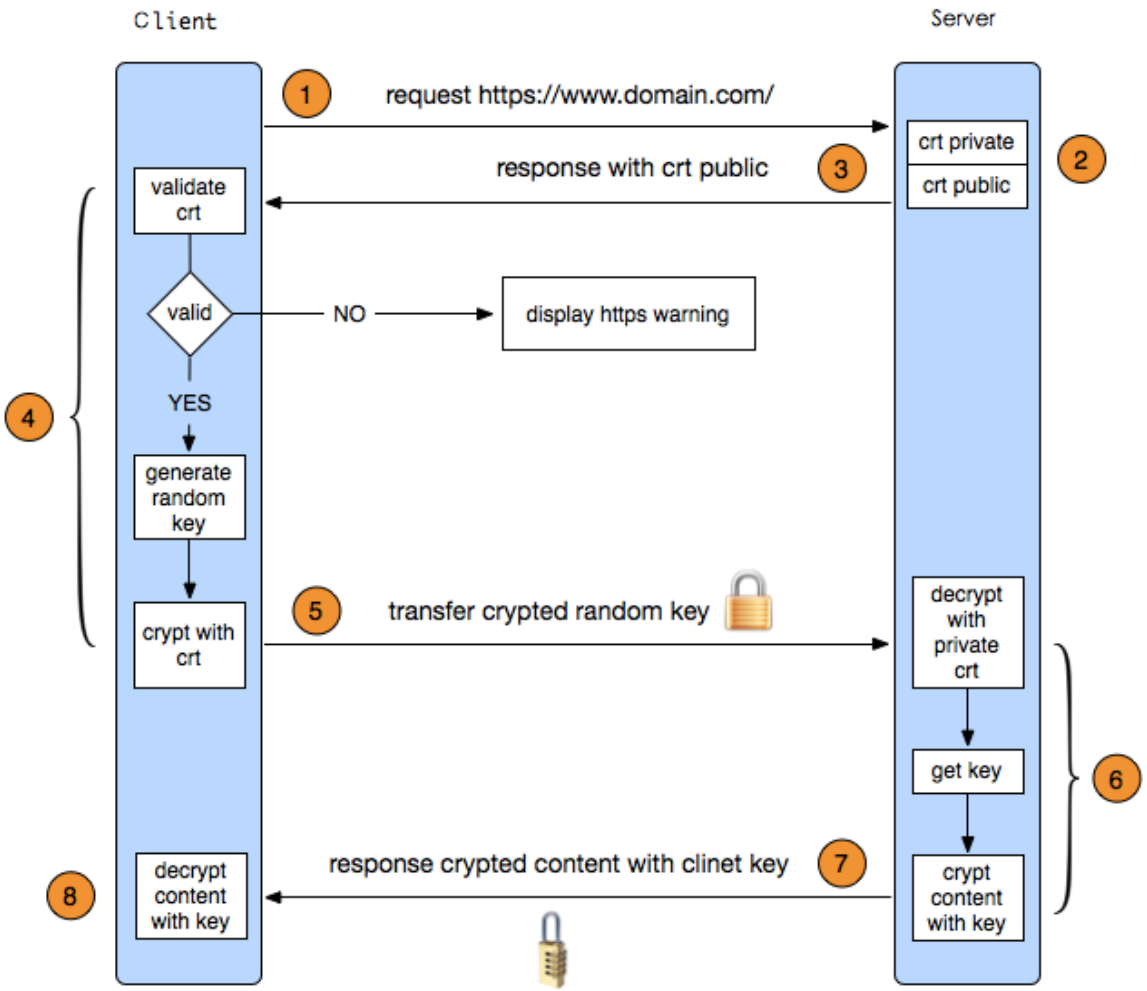
SPDY位于HTTP之下，TCP和SSL之上，这样可以轻松兼容老版本的HTTP协议(将http1.x的内容封装成一种新的frame格式)，同时可以使用已有的SSL功能。

SPDY规定建立在TLS之上，即URL scheme为https。

HTTPS

SUNING 苏宁金融

- HTTP协议传输的数据
  传输隐私信息非常不
  景公司设计了SSL（S
  据进行加密，从而就
- Https在真正请求数据
  份，以下图为例



推荐书籍：《https权威指南》

SPDY交互过程示例

SUNING
苏宁金融

以一个最简单的HTTP GET为例SPDY的交互过程是这样的：

```
[客户端] -SSL CLIENT HELLLO-> [服务器]
[客户端] <-SSL SERVER HELLO/CERTIFICATE/NPN (HTTP/1.1, SPDY/3, SPDY/2)等
[服务器] # 服务器通过SSL的NPN扩展告诉客户端我这支持HTTP 1.1也支持SPDY 2和3
[客户端] -SSL CLIENT CERTIFICATE/NPN(SPDY/3)等-> [服务器] # 客户端告诉服务器我选择SPDY3
[客户端] <-SSL SERVER FINISHED- [服务器] # SSL握手完成
[客户端] -SSL加密的SYN FRAME(HTTP GET)-> [服务器]
# SYN FRAME是SPDY版的HTTP GET，意思是一样的
[客户端] <-SSL加密的SYN REPLY FRAME(200 OK)- [服务器]
 # SYN REPLAY FRAME是SPDY版的200 OK，意思是一样的
```

SPDY基础功能

◆ **多路复用（multiplexing）。**

多路复用通过多个请求stream共享一个tcp连接的方式，解决了http1.x holb（head of line blocking）的问题，降低了延迟同时提高了带宽的利用率。

◆ **请求优先级（request prioritization）。**

多路复用带来一个新的问题是，在连接共享的基础之上有可能会导致关键请求被阻塞。SPDY允许给每个request设置优先级，这样重要的请求就会优先得到响应。比如浏览器加载首页，首页的html内容应该优先展示，之后才是各种静态资源文件，脚本文件等加载，这样可以保证用户能第一时间看到网页内容。

◆ **header压缩。**

前面提到过几次http1.x的header很多时候都是重复多余的。选择合适的压缩算法可以减小包的大小和数量。SPDY对header的压缩率可以达到80%以上，低带宽环境下效果很大。

SPDY高级功能

◆ **server推送（server push）。**

http1.x只能由客户端发起请求，然后服务器被动的发送response。开启server push之后，server通过X-Associated-Content header（X-开头的header都属于非标准的，自定义header）告知客户端会有新的内容推送过来。在用户第一次打开网站首页的时候，server将资源主动推送过来可以极大的提升用户体验。

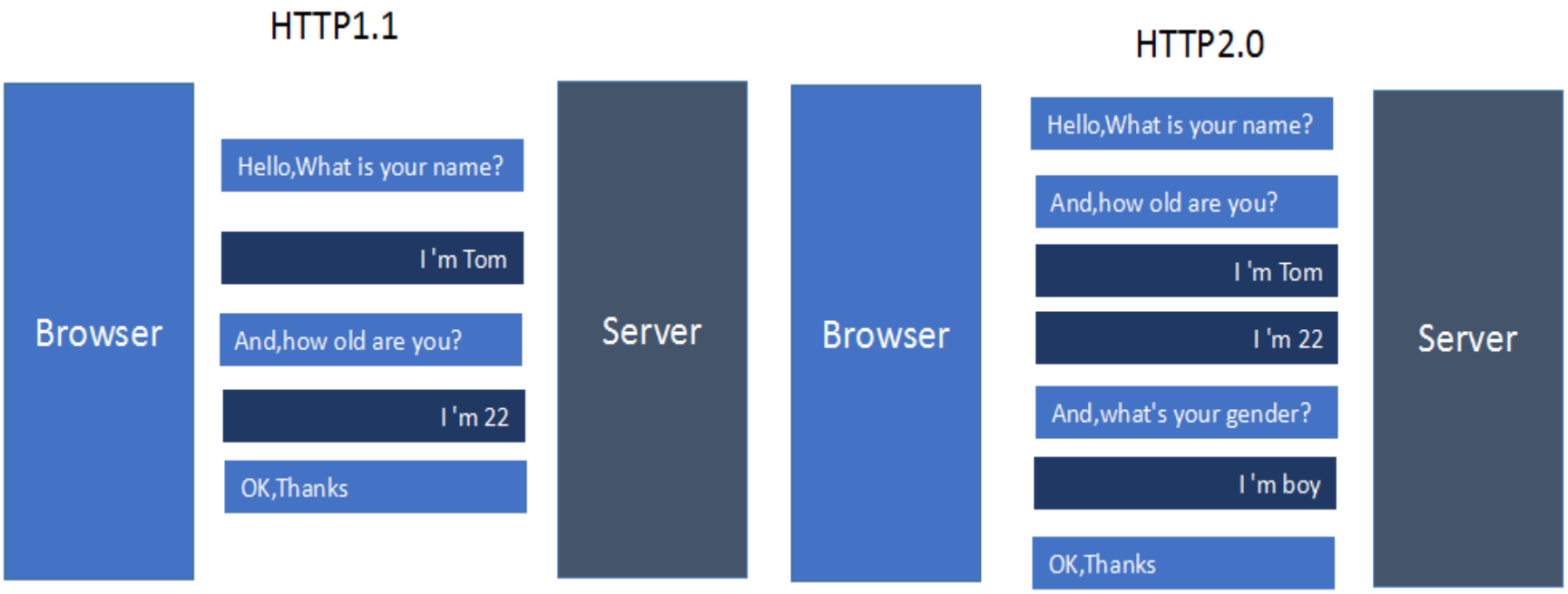
◆ **server暗示（server hint）。**

和server push不同的是，server hint并不会主动推送内容，只是告诉有新的内容产生，内容的下载还是需要客户端主动发起请求。server hint通过X-Subresources header来通知，一般应用场景是客户端需要先查询server状态，然后再下载资源，可以节约一次查询请求。

目录

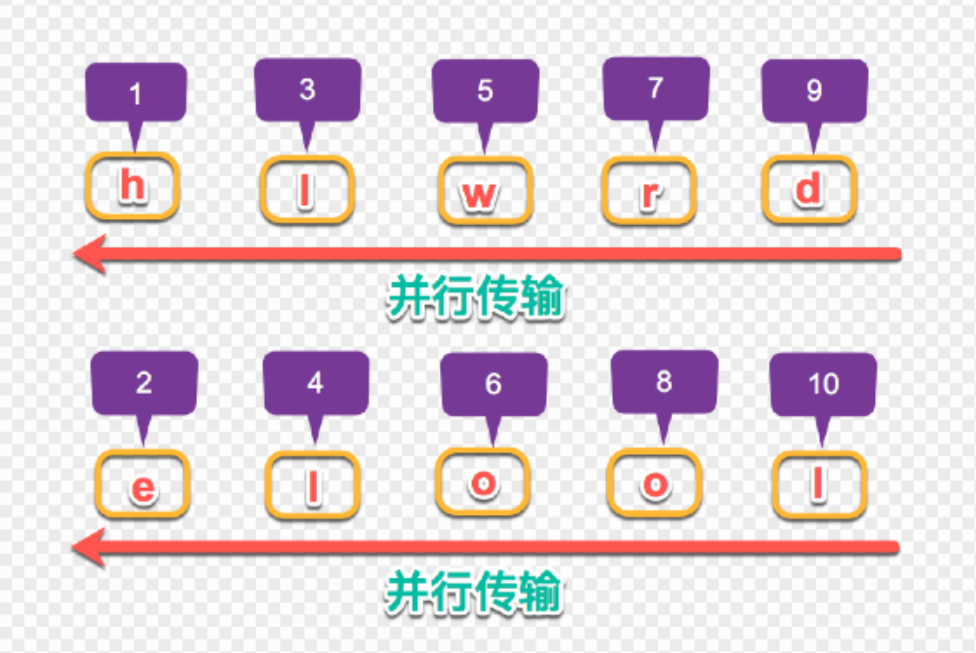
HTTP1.1 和 HTTP2.0 对比

SUNING
苏宁金融



HTTP2.0 多路复用解决的问题

- 在HTTP1.1的协议中，传输的request和response都是基本于文本的，这样就会引发一个问题：所有的数据必须按顺序传输。HTTP/2引入二进制数据帧和流的概念，其中帧对数据进行顺序标识。
- HTTP/2对同一域名下所有请求都是基于流，也就是说同一域名不管访问多少文件，也只**建立一路连接**。同样Apache的最大连接数为300，因为有了这个新特性，最大的并发就可以提升到300，比原来提升了6倍。



HTTP2.0 介绍

HTTP/2，也就是超文本传输协议第2版，不论是1还是2，HTTP的基本语义是不变的：

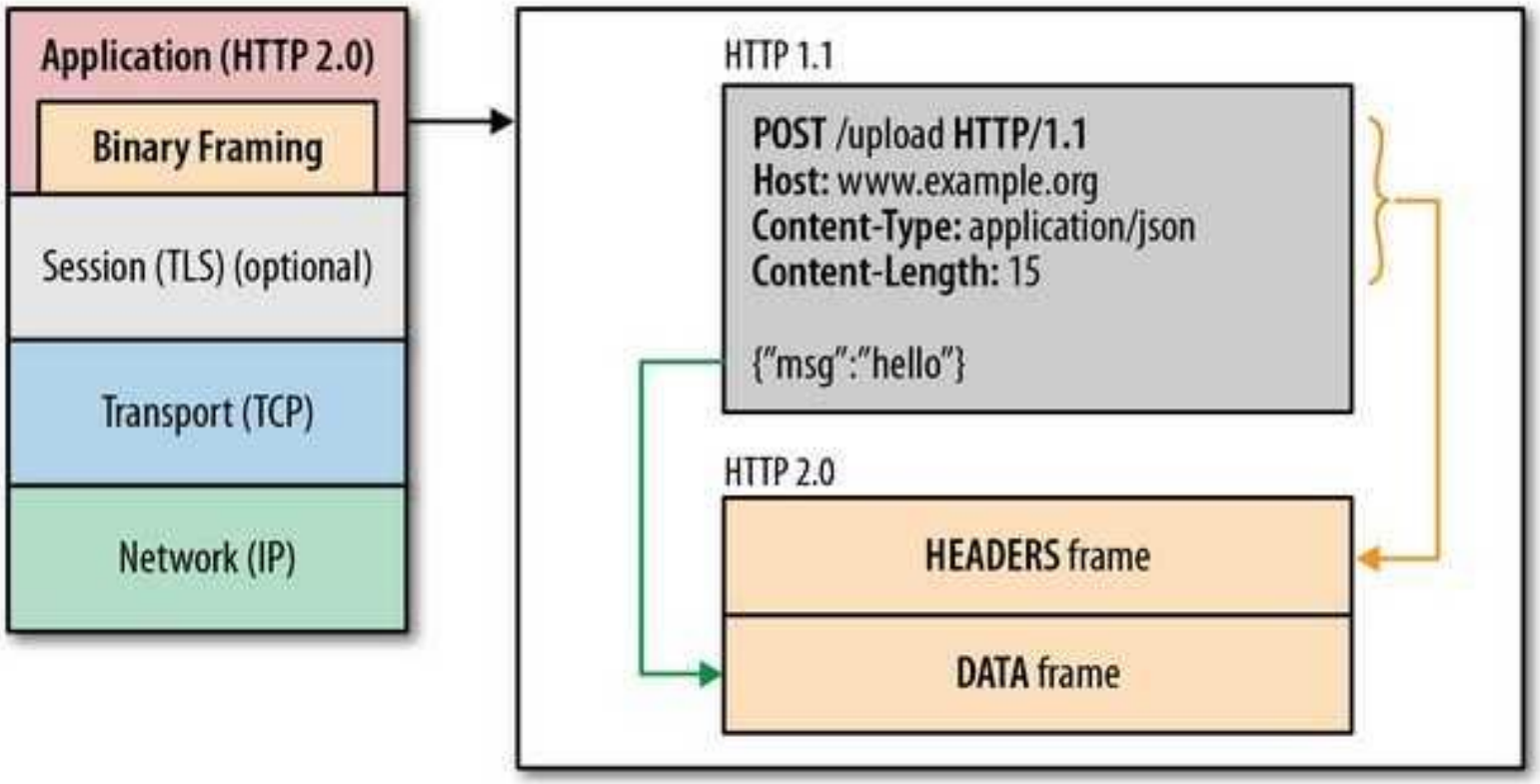
- HTTP/2 传输的数据是**二进制**的。更小的传输体积，这就意味着更低的负载。二进制的帧也更易于解析而且不易出错，相比 HTTP/1.1 的纯文本数据，在解析的时候还要考虑处理空格、大小写、空行和换行等问题。

- 在 HTTP/2 的语境下，有三个概念：**流（Stream）**、**消息（Message）** 和**帧（Frame）**。
  - **Stream** 处于一个连接中的双向二进制数据流，可以包含一个或者多个 Message。
  - **Message** 一个完整的请求或者响应，包含多个 Frame 序列。
  - **Frame** HTTP/2 通讯中的最小传输单位，至少含有一个 Frame header，能够表示它属于哪一个 Stream。
- **与SPDY的区别**

  HTTP2.0 支持明文 HTTP 传输，而 SPDY 强制使用 HTTPS

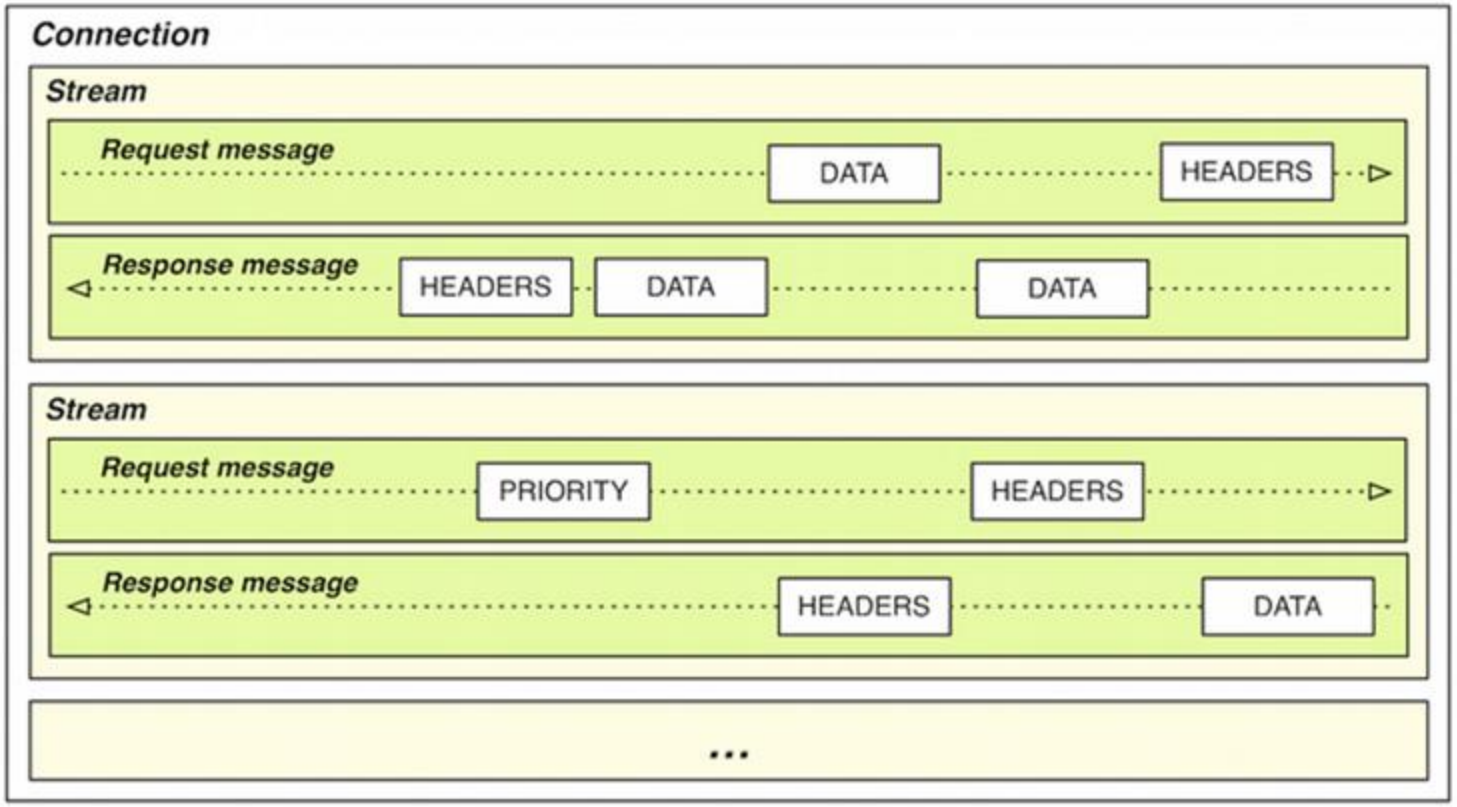
  HTTP2.0 消息头的压缩算法采用 HPACK，而非 SPDY 采用的 DEFLATE

二进制分帧层


Application (HTTP 2.0)
Binary Framing
Session (TLS) (optional)
Transport (TCP)
Network (IP)
HTTP 1.1
POST /upload HTTP/1.1
Host: www.example.org
Content-Type: application/json
Content-Length: 15
{"msg":"hello"}
HTTP 2.0
HEADERS frame
DATA frame

HTTP2.0 报文示意

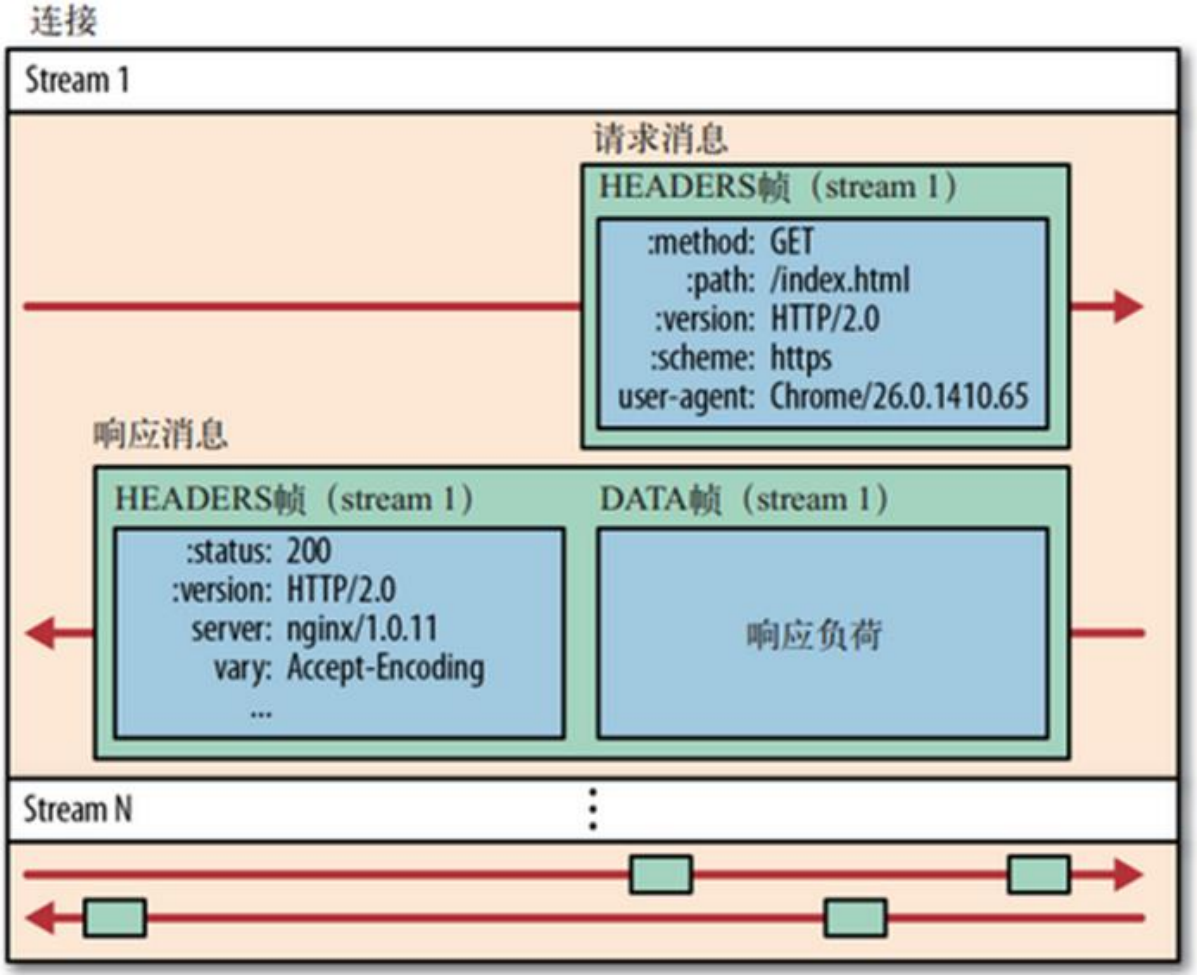
如下图所示：



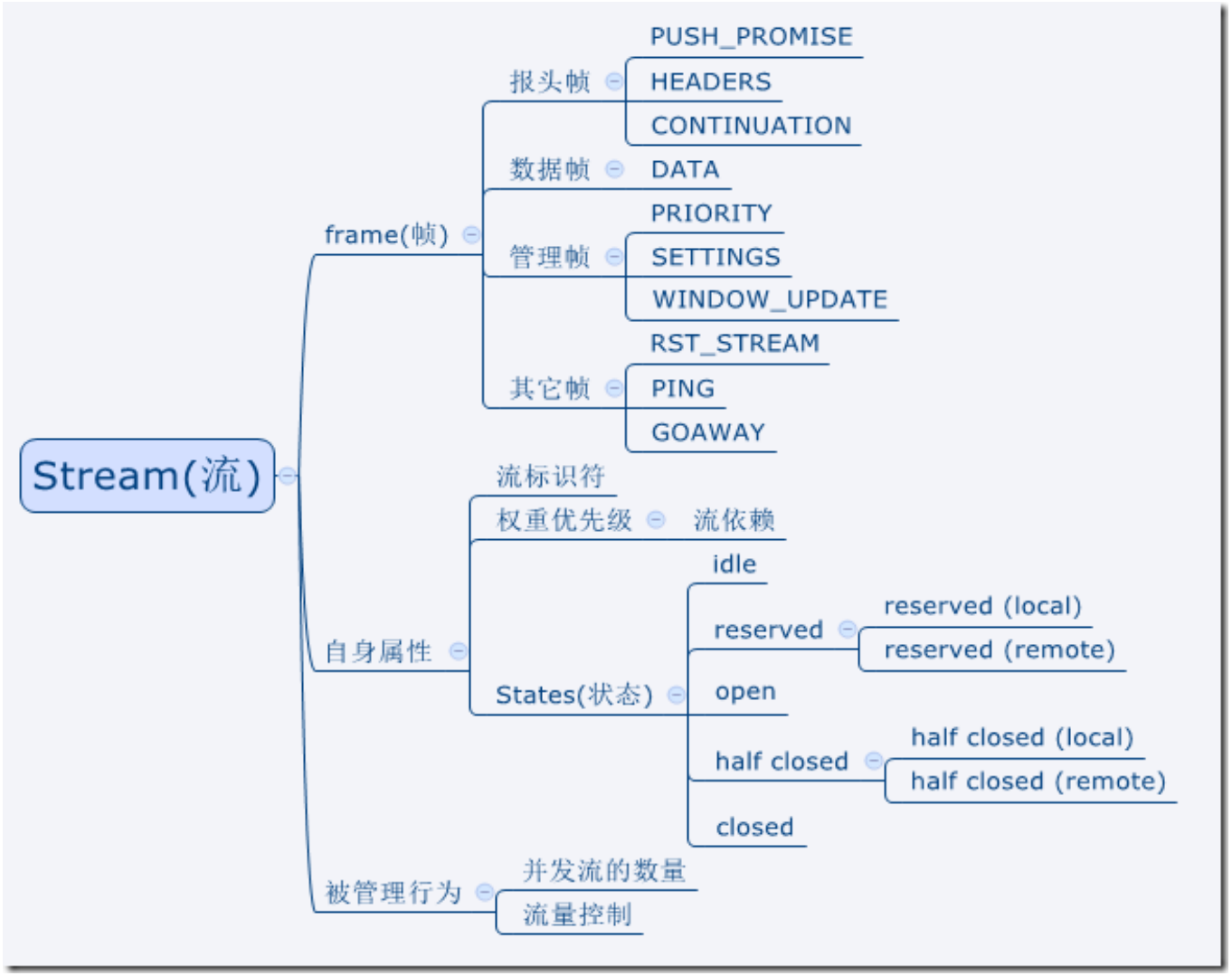
HTTP2.0 请求示例

SUNING
苏宁金融

一个具体请求的HTTP2类似如下图：



HTTP2.0 流的组成

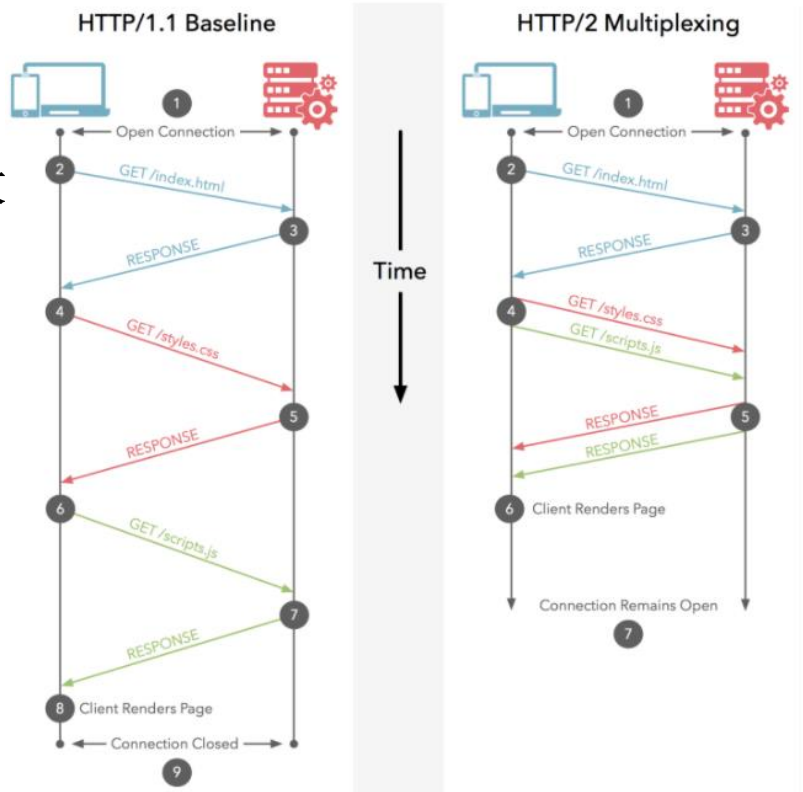

Stream(流)
frame(帧)
报头帧
PUSH_PROMISE
HEADERS
CONTINUATION
数据帧
DATA
管理帧
PRIORITY
SETTINGS
WINDOW_UPDATE
其它帧
RST_STREAM
PING
GOAWAY
自身属性
流标识符
权重优先级
流依赖
States(状态)
idle
reserved
reserved (local)
reserved (remote)
open
half closed
half closed (local)
half closed (remote)
closed
被管理行为
并发流的数量
流量控制

多路复用的流

SUNING
苏宁金融

- HTTP2.0允许**多个并发 HTTP 请求**共用一个 TCP会话，而不是为每个请求单独开放连接，这样只需建立一个 TCP 连接就可以传送网页上所有资源，不仅可以减少消息交互往返的时间还可以避免创建新连接造成的延迟，使得 TCP 的效率更高。

- 优势：
  - 单连接多资源的方式，减少服务端的链接压力,内存占用更少,连接吞吐量更大
  - 由于 TCP 连接的减少而使网络拥塞状况得以改善，同时慢启动时间的减少,使拥塞和丢包恢复速度更快



首部压缩(HPACK 算法)

SUNING 苏宁金融

需要在支持 HTTP/2 的浏览器和服务端之间：

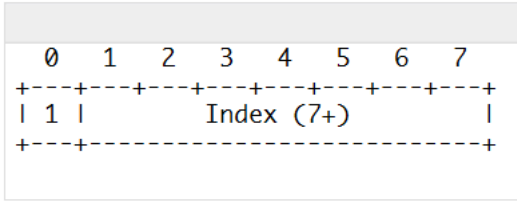
- 维护一份相同的**静态字典**（Static Table），包含常见的头部名称，以及特别常见的头部名称与值的组合；
- 维护一份相同的**动态字典**（Dynamic Table），可以动态地添加内容；
- 支持基于静态**哈夫曼码表**的哈夫曼编码（Huffman Coding）；

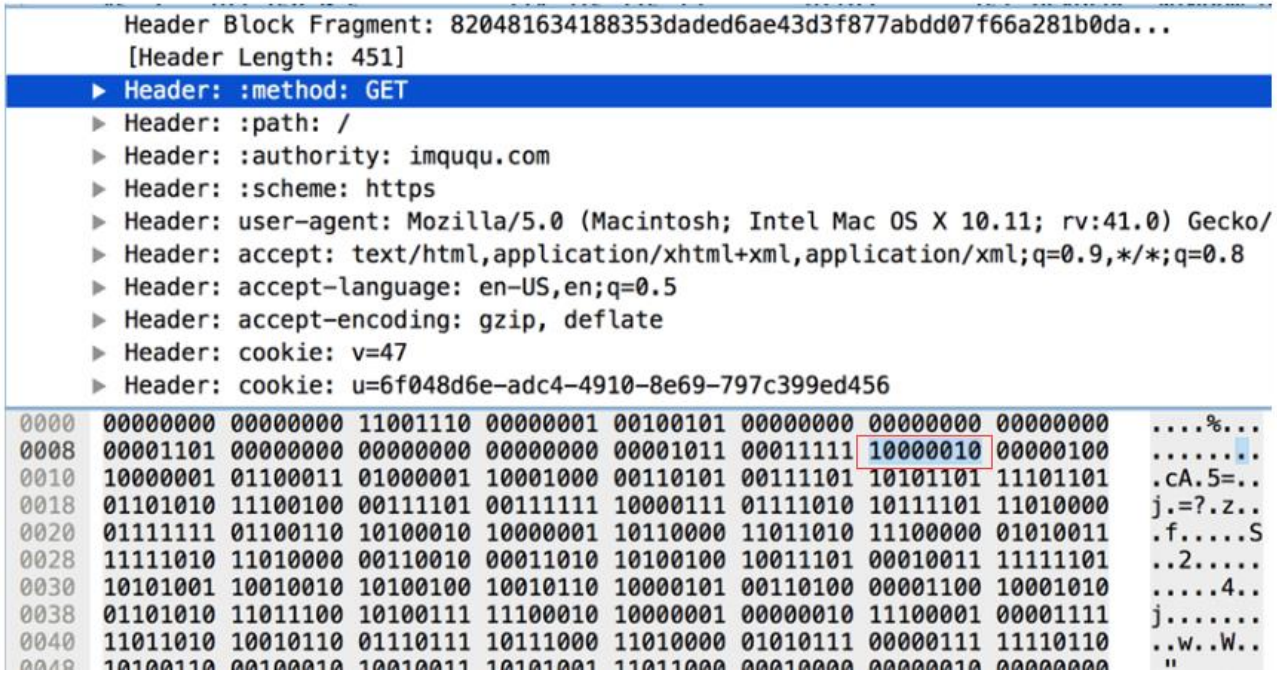
Index	Header Name	Header Value
1	:authority	
2	:method	GET
3	:method	POST
4	:path	/
5	:path	/index.html
6	:scheme	http
7	:scheme	https
8	:status	200
...	...	...
32	cookie	
...	...	...
60	via	
61	www-authenticate	

首部压缩举例

SUNING
苏宁金融

1）整个头部键值对都在字典中
2）头部名称在字典中，更新动态字典
3）头部名称不在字典中，更新动态字典
4）头部名称在字典中，不允许更新动态字典
5）头部名称不在字典中，不允许更新动态字典

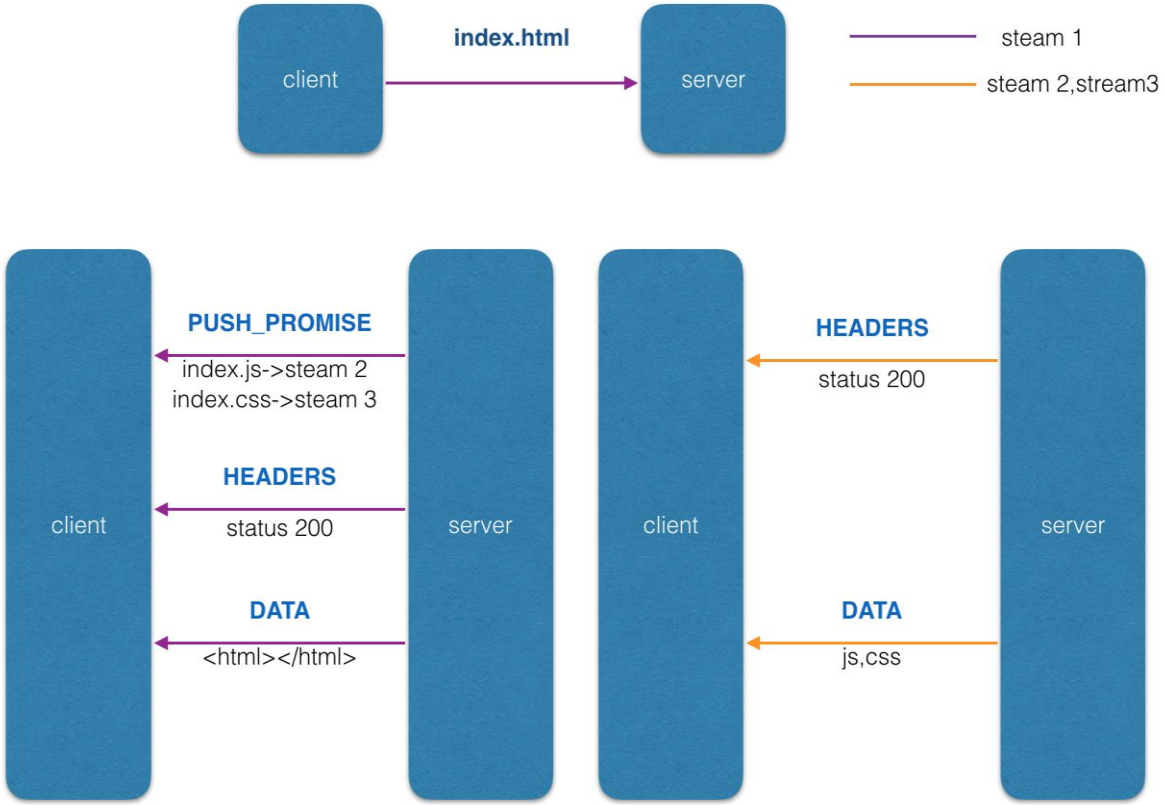




Server Push

SUNING 苏宁金融

当用户的浏览器和服务器在建立链接后，服务器主动将一些资源**推送**给浏览器**并缓存**起来，这样当浏览器接下来请求这些资源时就直接**从缓存中读取**，不会在从服务器上拉了，提升了速率。



对开发的影响举例

- 不需要**JS文件的合并，共用一个http连接，不存在重新下载、重新文件缓存问题。各模块可以单独压缩上线。**

- **不需要多域名提高浏览器的下载速度，减少成本**。例如原来是把css文件和js放到不同域名下，同时下载，但DNS解析时间会长 .

应用场景情况

- 客户端向server发送request这种基本模型不会变。
- 老的scheme不会变，使用http://和https://的服务和应用不会要做任何更改，不会有http2://。
- 使用http1.x的客户端和服务器可以无缝的通过代理方式转接到http2.0上。
- 不识别http2.0的代理服务器可以将请求降级到http1.x。

使用 ALPN

HTTP2.0使用了tls的拓展ALPN来做协议升级，除此之外加密这块还有一个改动，HTTP2.0对tls的安全性做了近一步加强，通过**黑名单机制**禁用了几百种不再安全的加密算法，一些加密算法可能还在被继续使用。

ALPN（Application Layer Protocol Negotiation）

区别就在于谁持有会话协议的决定权。**ALPN**是由客户端给服务器发送一个协议清单，**由服务器**来最终选择一个。而NPN则正好相反。

架构于tls之上就成为http/2的事实上的标准。所有协商协议也都支持NPN（tls的一个扩展）。

http 1.1 与 http2 速度比较

SUNING
苏宁金融

https://http2.akamai.com/demo

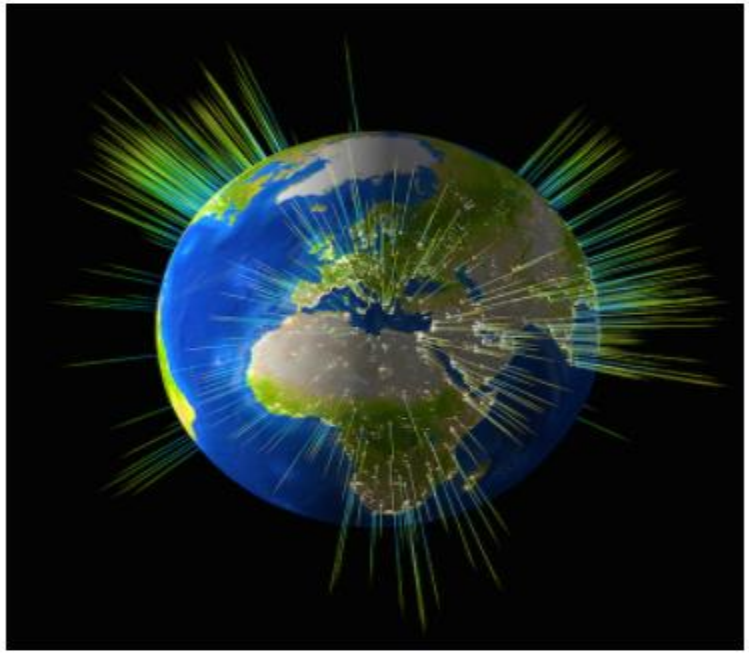
HTTP/2 is the future of the Web, and it is here!

Your browser supports HTTP/2!

This is a demo of HTTP/2's impact on your download of many small tiles making up the Akamai Spinning Globe.

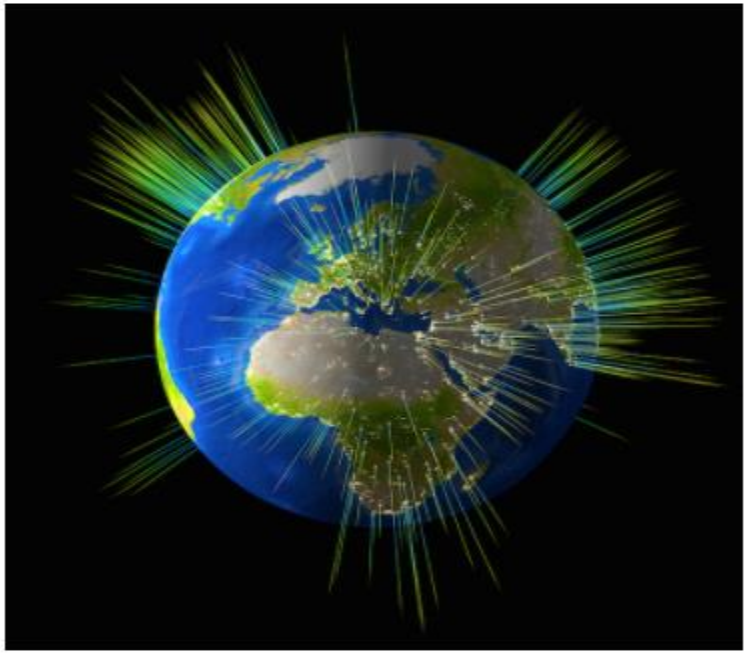
HTTP/1.1

Latency: 88ms
Load time: 6.61s



HTTP/2

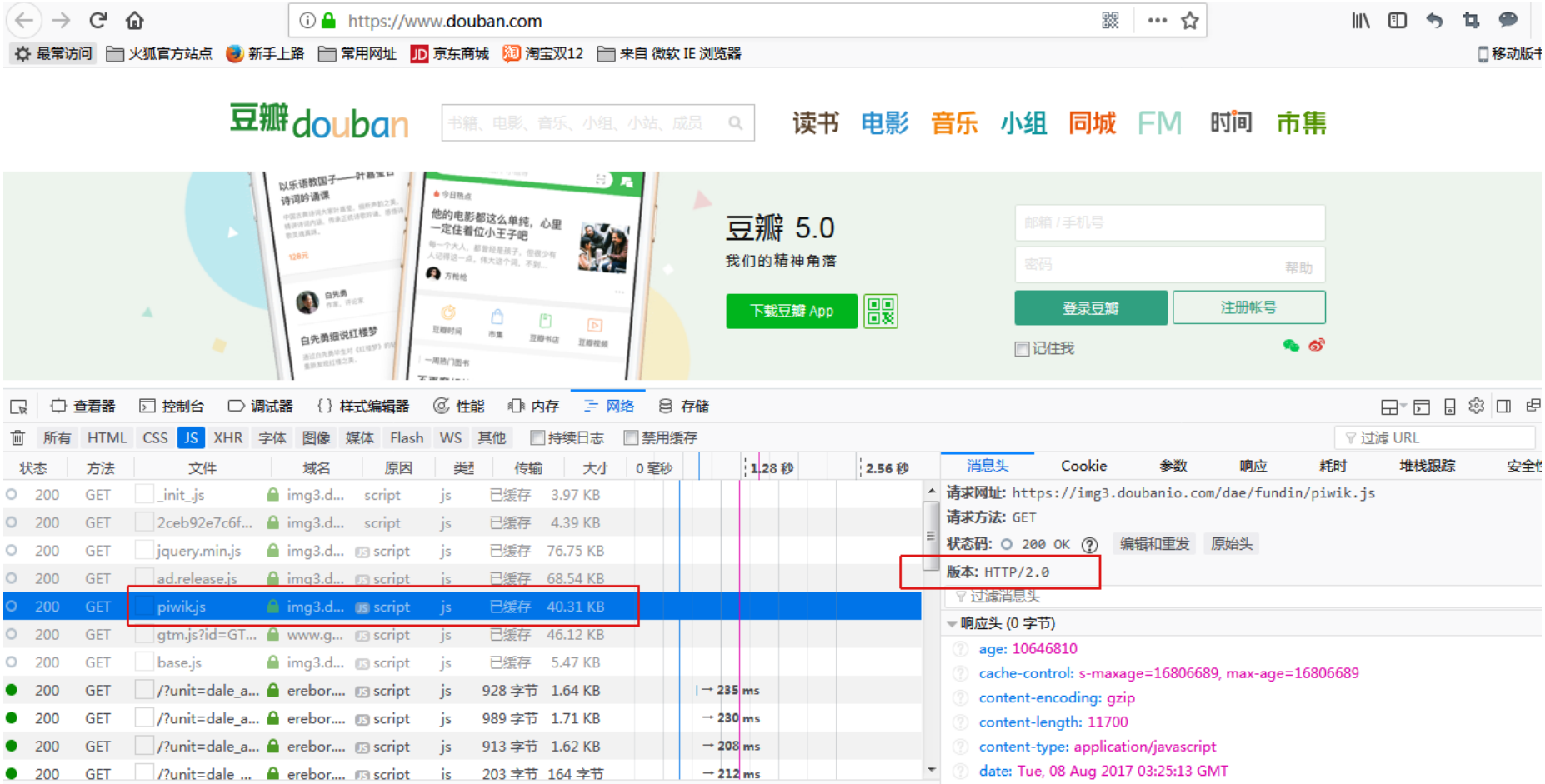
Latency: 87ms
Load time: 2.96s



http2 支持的网站

SUNING 苏宁金融

- HTTP2.0的实现已经比较成熟，截至2016年12月，前1000万个网站中，有10.8％支持HTTP2.0，其中有Google、Twitter等行业先驱。
- 国内网站中，百度、豆瓣、知乎、QQ邮箱、携程、搜狐、蘑菇街及部分直播平台等已经开始用HTTP2.0。



SUNING
苏宁金融

SUNING 苏宁金融
全渠道 更安心

谢 谢 欣 赏

THE END

thank you!