# 计算机网络：

## http的请求方式有哪些？可以自己定义新的请求方式么？

get:

A：请求参数跟在url之后，信息不安全。

B：请求数据大小有限制(1024K)。

C：会产生一个TCP数据包，data与header一起发送。

D：GET用于信息获取，而且应该是安全的和幂等的

post:

A：请求数据跟在请求后面。信息安全。

B：理论上请求数据大小无限制。

C：产生两个tcp数据包，先发送header，服务器返回100后再发送data包。

D：POST表示可能修改变服务器上的资源的请求

head:与get方式相同，但是head不返回报文的主体部分。

## Socket通信，以及长连接，分包，连接异常断开处理。

1. Socket通信模型的使用，AIO和NIO。
2. **Socket框架netty的使用，以及NIO的实现原理，为什么是异步非阻塞。**

## 同步和异步，阻塞与非阻塞IO；

**同步与异步**  
同步和异步关注的是**消息通信机制** (synchronous communication/ asynchronous communication)  
所谓同步，就是在发出一个\*调用\*时，在没有得到结果之前，该\*调用\*就不返回。但是一旦调用返回，就得到返回值了。  
换句话说，就是由\*调用者\*主动等待这个\*调用\*的结果。

而异步则是相反，**\*调用\*在发出之后，这个调用就直接返回了，所以没有返回结果**。换句话说，当一个异步过程调用发出后，调用者不会立刻得到结果。而是在\*调用\*发出后，\*被调用者\*通过状态、通知来通知调用者，或通过回调函数处理这个调用。

阻塞与非阻塞

阻塞和非阻塞关注的是程序在等待调用结果（消息，返回值）时的**状态**.

阻塞调用是指调用结果返回之前，当前线程会被挂起。调用线程只有在得到结果之后才会返回。

非阻塞调用指在不能立刻得到结果之前，该调用不会阻塞当前线程。

## udp，tcp之间的区别。

1、Tcp面向连接的，UDP不是面向连接的。

2、因此tcp只支持点对点的全双工通信，而udp支持一对多，一对一，多对多的通信。

3、Tcp是面向字节流操作的，对数据不打包，而udp是面向报文操作的，对数据打包，大小限制在64k。

4、Tcp保证传输数据的可靠性，udp不保证传输数据的可靠性。

6、Tcp有拥塞控制，udp没有。

7、Udp首部开销小，只有8个字节，tcp首部开销需要20个字节。

## TCP如何实现可靠传输

## 负载均衡

<https://www.cnblogs.com/danbing/p/7459224.html>

<https://tech.meituan.com/MGW.html>

<https://www.jianshu.com/p/fa937b8e6712>

## 说说浏览器访问[www.taobao.com](http://www.taobao.com)经历了什么过程？

1、DNS解析（应用层）

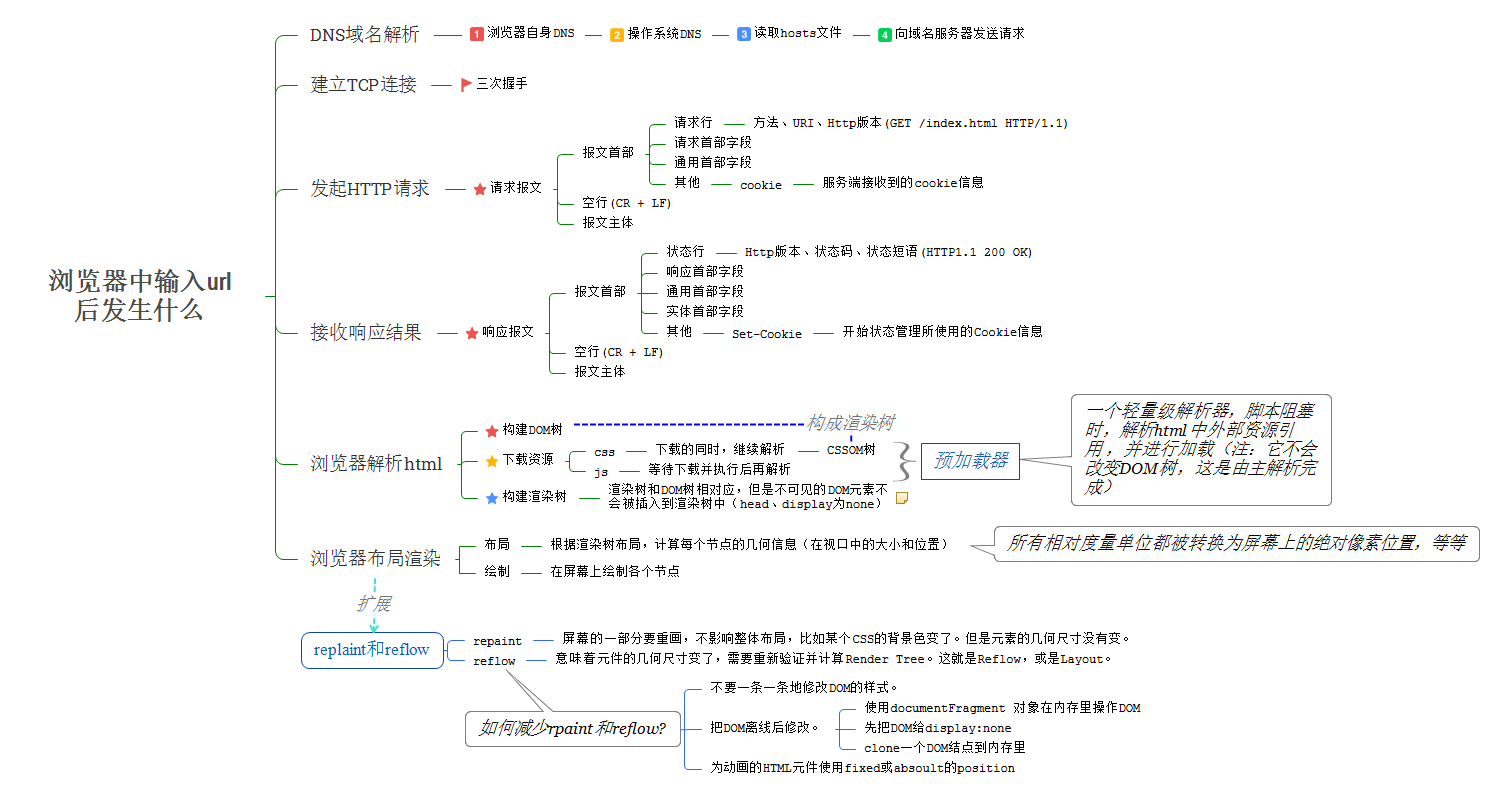
2、TCP连接（传输层）

3、发送HTTP请求（应用层）

4、服务器处理请求并返回HTTP报文

5、浏览器解析渲染页面

6、连接结束



参考： <https://www.jianshu.com/p/c1dfc6caa520>

<http://www.xuecaijie.com/it/157.html>

## HTTP协议，HTTPS协议，SSL协议及完整的交互过程？

由于HTTP协议是明文传输，不验证通信者身份和不保证报文的完整性，因此安全性较低，就出现了在HTTP协议基础上加上安全套接字层（SSL）协议加密的HTTPS协议。但https协议需要向ca申请证书，不是免费的。他们采用的连接方式也不一样，使用的端口也不一样，http是80端口，https是443端口。

SSL协议：

## Tcp的拥塞，快重传，ip的报文丢弃。

Tcp的拥塞控制算法：慢开始，拥塞避免，快重传，快恢复。

慢开始：发送方维持一个拥塞控制窗口（cwnd），当连接刚建立时，不是立即发送大量数据包，而是有规律的递增发送数据。新建连接时，cwnd窗口大小为1，当有一个报文被确定收到后，cwnd就增加一个最大报文段大小，这样下去，cwnd能迅速增加起来，速度也是可观的。

拥塞避免：但是cwnd的大小是不能无限增长下去的，因此有一个慢开始门限值，当超过该值时，慢开始过程结束。进入拥塞避免过程，每次报文确定后，cwnd加一，避免了指数性增长带来拥塞。

当tcp发现网络拥塞时，会将慢开始门限值减半，并将cwnd窗口置为1，重新开始慢开始和拥塞避免过程，但是快重传和快恢复却不是这样做的。

快重传：当报文丢失，接受方立即向发送方发送重复确认，当发送接收了三次重复确认后，不等待超时重传时间，会立即重传丢失的报文段。

快恢复：当收到三次重复确认后，将慢开始门限值减半，但不将cwnd置为1，而是置为慢开始门限值减半后的值，之后进入拥塞避免状态。

## http响应状态码。

1xx：表示请求成功收到，需要请求者继续操作。100：continue，101：切换协议

2xx：表示一个请求成功收到并处理。200 OK：请求成功。

3xx：表示重定向，301：永久重定向；302：临时重定向。

4xx：表示客户端的错误。404：资源没找到，可能地址输错，403：收到请求，但禁止访问。

5xx:服务器端错误。500：服务器内部发生了错误，无法处理请求

503：由于超载或者维护，服务器暂时无法处理请求。

## https的一个过程。

## TCP三次握手与四次挥手；

**三次握手：**

**四次挥手：**