**HTTP相关**

**1. HTTP有什么特点**

* 简单快速：客户向服务器请求服务时，只需传送请求方法和路径
* 灵活：HTTP允许传输任意类型的数据对象。正在传输的类型由 Content-Type 加以标记
* 无连接：无连接的含义是限制每次连接只处理一个请求。服务器处理完客户的请求，并收到客户的应答后，即断开连接 (深入-持久连接、管线化)
* 无状态：HTTP协议是无状态协议( Cookie 的出现)

**2. http和https协议有什么区别（参考 http与https的区别）**

http: 是互联网上应用最为广泛的一种网络协议，是一个客户端和服务器端请求和应答的标准（TCP），用于从WWW服务器传输超文本到本地浏览器的传输协议，它可以使浏览器更加高效，使网络传输减少

https: 是以安全为目标的HTTP通道，简单讲是 HTTP 的安全版，即 HTTP 下加入 SSL 层，HTTPS 的安全基础是 SSL ，因此加密的详细内容就需要 SSL

* http 是超文本传输协议，信息是明文传输，https 则是具有安全性的 ssl 加密传输协议
* http 和 https 使用的是完全不同的连接方式，用的端口也不一样，前者是 80 ，后者是 443
* http 的连接很简单，是无状态的；HTTPS 协议是由 SSL+HTTP 协议构建的可进行加密传输、身份认证的网络协议，比 http 协议安全

**3. http状态码有那些？分别代表是什么意思**

常用 http 状态码：

* 200 OK 服务器成功处理了请求
* 301/302 Moved Permanently（重定向）请求的URL已移走
* 404 Not Found (页面丢失)未找到资源
* 403 服务器拒绝请求
* 408 （请求超时） 服务器等候请求时发生超时
* 501 Internal Server Error 服务器遇到一个错误，使其无法对请求提供服务
* 502 （错误网关） 服务器作为网关或代理，从上游服务器收到无效响应
* 504 （网关超时） 服务器作为网关或代理，但是没有及时从上游服务器收到请求
* 304 (SC\_NOT\_MODIFIED)是指缓冲的版本已经被更新并且客户端应刷新文档
* 403 (SC\_FORBIDDEN)的意思是除非拥有授权否则服务器拒绝提供所请求的资源。这个状态经常会由于服务器上的损坏文件或目录许可而引起。
* 405 (SC\_METHOD\_NOT\_ALLOWED)指出请求方法(GET, POST, HEAD, PUT, DELETE, 等)对某些特定的资源不允许使用。该状态码是新加入 HTTP 1.1中的。

**4. 什么是HTTP持久化和管线化**

出现背景： HTTP 最初的版本中，每进行一次 HTTP 通信，就要断开一次 TCP 连接（无连接）

为解决上述问题，HTTP/1.1 增加了持久连接（HTTP Persistent Connections ）的方法，其特点是，只要一方未明确提出断开连接，则另一方保持 TCP 连接状态

管线化是指将多个 HTTP 请求整批发送，在发送过程中不用等待对方响应

管线化是在持久连接的基础上实现的，管线化的实现，能够同时并行发送多个请求，而不需要一个接一个的等待响应

**5. Http报文**

HTTP 报文是面向文本的，报文中的每一个字段都是一些 ASCII 码串，各个字段的长度是不确定的。HTTP 有两类报文：请求报文和响应报文

HTTP的这两种报文都由三部分组成：开始行、首部行、实体主体

参考 [这里](https://www.jianshu.com/p/a2c4ede32d11)

**6. 从输入URL到页面加载全过程**

* DNS解析
* TCP连接
* 发送HTTP请求
* 服务器处理请求并返回HTTP报文
* 浏览器解析渲染页面

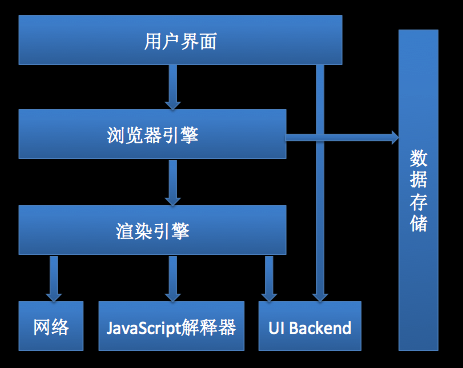
连接结束**7. 为什么利用多个域名来存储网站资源会更有效**

* CDN 缓存更方便
* 突破浏览器并发限制
* 节约 cookie 带宽
* 节约主域名的连接数，优化页面响应速度
* 防止不必要的安全问题

**浏览器相关**

**1. 浏览器是由什么组成的**

从原理构成上分为七个模块，分别是 User Interface（用户界面）、 Browser engine（浏览器引擎） 、 Rendering engine（渲染引擎） 、 Networking（网络） 、 JavaScript Interpreter（js解释器） 、 UI Backend（UI后端） 、Date Persistence（数据持久化存储）

[](https://raw.githubusercontent.com/ltadpoles/web-document/master/Other/images/%E6%B5%8F%E8%A7%88%E5%99%A8.png)

其中，最重要的是渲染引擎（内核）和 JavaScript 解释器（JavaScript引擎）

浏览器内核主要负责 HTML 、CSS 的解析，页面布局、渲染与复合层合成; JavaScript 引擎负责 JavaScript 代码的解释与执行

**2. 浏览器缓存机制**

浏览器的缓存机制也就是我们说的 HTTP 缓存机制，其机制是根据 HTTP 报文的缓存标识进行的

**3. 浏览器渲染机制**

从耗时的角度，浏览器请求、加载、渲染一个页面，时间花在下面五件事情上：

* DNS 查询
* TCP 连接
* HTTP 请求即响应
* 服务器响应
* 客户端渲染

本文讨论第五个部分，即浏览器对内容的渲染，这一部分（渲染树构建、布局及绘制），又可以分为下面五个步骤：

* 处理 HTML 标记并构建 DOM 树。
* 处理 CSS 标记并构建 CSSOM 树。
* 将 DOM 与 CSSOM 合并成一个渲染树。
* 根据渲染树来布局，以计算每个节点的几何信息。
* 将各个节点绘制到屏幕上。

**4. 几个很实用的BOM属性对象方法**

location 对象：主要存储 url 相关信息

history 对象：浏览历史信息相关

history.go() // 前进或后退指定的页面数 history.go(num);

history.back() // 后退一页

history.forward() // 前进一页

navigator 对象：浏览器信息相关

navigator.userAgent //返回用户代理头的字符串表示(就是包括浏览器版本信息等的字符串)

navigator.cookieEnabled // 返回浏览器是否支持(启用)cookie

[](https://raw.githubusercontent.com/ltadpoles/web-document/master/Other/images/location.jpg)

**其他**

**1. 谈谈你对SEO的理解（**[**前端SEO优化**](https://imweb.io/topic/5682938b57d7a6c47914fc00)**）**

SEO：搜索引擎优化，其目的是为了使网站能够更好的被搜索引擎抓取，提高在搜索引擎内的自然排名，从而带来更多的免费流量，获取收益

SEO主要有两种方法，站内优化和站外优化

**2. 前端怎么控制管理路由**

路由就是浏览器地址栏中的 url 与所见网页的对应关系

前端路由的实现方式：

基于 hash（ocation.hash+hashchange事件）

展示层面也就是切换 # 后面的内容，呈现给用户不同的页面。现在越来越多的单页面应用，基本都是基于 hash 实现

特性：

* url 中 hash 值的变化并不会重新加载页面
* hash 值的改变，都会在浏览器的访问历史中增加一个记录，也就是能通过浏览器的回退、前进按钮控制 hash 的切换
* 我们可以通过 hashchange 事件，监听到 hash 值的变化，从而响应不同路径的逻辑处理

基于 istory 新 API（ history.pushState()+popState 事件）

window.history.pushState(null, null, "http://www.google.com");

这两个 API 的相同之处是都会操作浏览器的历史记录，而不会引起页面的刷新。不同之处在于，pushState 会增加一条新的历史记录，而 replaceState 则会替换当前的历史记录

**3. 防抖和节流的区别**

防抖：任务频繁触发的情况下，只有任务触发的间隔超过指定间隔的时候，任务才会执行

在触发点击事件后，如果用户再次点击了，我们会清空之前的定时器，重新生成一个定时器。意思就是：这件事儿需要等待，如果你反复催促，我就重新计时！

节流：指定时间间隔内只会执行一次任务

**4. 页面重构怎么操作**

页面重构就是根据原有页面内容和结构的基础上，通过 div+css 写出符合 web 标准的页面结构。

具体实现要达到以下三点：

* 功能不全页面的重构：页面功能符合用户体验、用户交互结构完整，可通过标准验证，
* 代码重构：代码质量、SEO 优化、页面性能、更好的语义化、浏览器兼容、CSS 优化
* 充分考虑到页面在站点中的“作用和重要性”，并对其进行有针对性的优化

**5. DNS解析**

网络客户端就是我们平常使用的电脑，打开浏览器，输入一个域名。比如输入www.163.com，这时，你使用的电脑会发出一个DNS请求到本地DNS服务器。本地DNS服务器一般都是你的网络接入服务器商提供，比如中国电信，中国移动。

查询www.163.com的DNS请求到达本地DNS服务器之后，本地DNS服务器会首先查询它的缓存记录，如果缓存中有此条记录，就可以直接返回结果。如果没有，本地DNS服务器还要向DNS根服务器进行查询。

根DNS服务器没有记录具体的域名和IP地址的对应关系，而是告诉本地DNS服务器，你可以到域服务器上去继续查询，并给出域服务器的地址。

本地DNS服务器继续向域服务器发出请求，在这个例子中，请求的对象是.com域服务器。.com域服务器收到请求之后，也不会直接返回域名和IP地址的对应关系，而是告诉本地DNS服务器，你的域名的解析服务器的地址。

最后，本地DNS服务器向域名的解析服务器发出请求，这时就能收到一个域名和IP地址对应关系，本地DNS服务器不仅要把IP地址返回给用户电脑，还要把这个对应关系保存在缓存中，以备下次别的用户查询时，可以直接返回结果，加快网络访问。

**6、TCP三次握手**

所谓三次握手(Three-way Handshake)，是指建立一个TCP连接时，需要客户端和服务器总共发送3个包。  
三次握手的目的是连接服务器指定端口，建立TCP连接,并同步连接双方的序列号和确认号并交换 TCP 窗口大小信息.在socket编程中，客户端执行connect()时。将触发三次握手

第一次握手:

客户端发送一个TCP的SYN标志位置1的包指明客户打算连接的服务器的端口，以及初始序号X,保存在包头的序列号(Sequence Number)字段里。

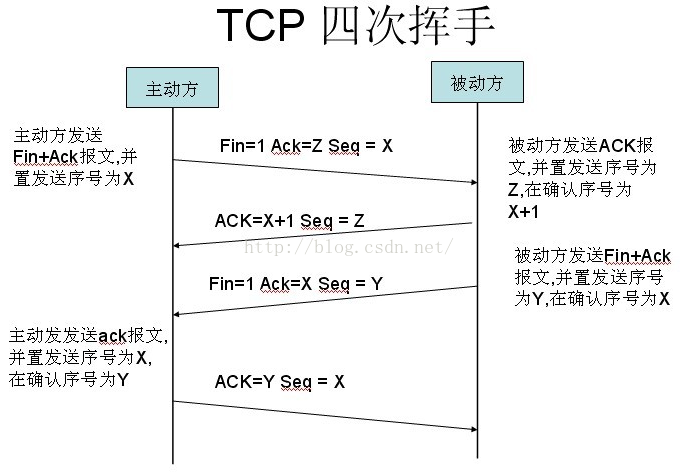
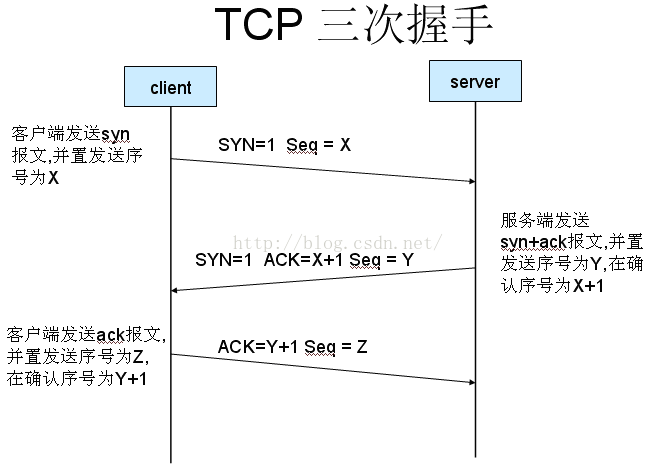
第二次握手:

服务器发回确认包(ACK)应答。即SYN标志位和ACK标志位均为1同时，将确认序号(Acknowledgement Number)设置为客户的I S N加1以.即X+1。  
服务器发回确认包(ACK)应答。即SYN标志位和ACK标志位均为1同时，将确认序号(Acknowledgement Number)设置为客户的I S N加1以.即X+1。

第三次握手.

客户端再次发送确认包(ACK)SYN标志位为0,ACK标志位为1.并且把服务器发来ACK的序号字段+1,放在确定字段中发送给对方.并且在数据段放写ISN的+1

**7. TCP 四次挥手**

TCP的连接的拆除需要发送四个包，因此称为四次挥手(four-way handshake)。客户端或服务器均可主动发起挥手动作，在socket编程中，任何一方执行close()操作即可产生挥手操作。

### 为什么建立连接是三次握手，而关闭连接却是四次挥手呢？

这是因为服务端在LISTEN状态下，收到建立连接请求的SYN报文后，把ACK和SYN放在一个报文里发送给客户端。而关闭连接时，当收到对方的FIN报文时，仅仅表示对方不再发送数据了但是还能接收数据，己方也未必全部数据都发送给对方了，所以己方可以立即close，也可以发送一些数据给对方后，再发送FIN报文给对方来表示同意现在关闭连接，因此，己方ACK和FIN一般都会分开发送。