

在浏览器输入一个网址（URL）发生了什么？

这是一道非常经典的题目，经常出现在面试中。

开放性的题目，没有固定的答案，涉及计算机图形学、操作系统、编译原理、计算机网络、通信原理、分布式系统、浏览器原理等多个不同的学科、领域。

答案：

一共分七步，即：

第一步：域名解析

第二步：建立TCP链接

第三步：发送HTTP请求

第四步：服务器处理请求

第五步：返回响应结果

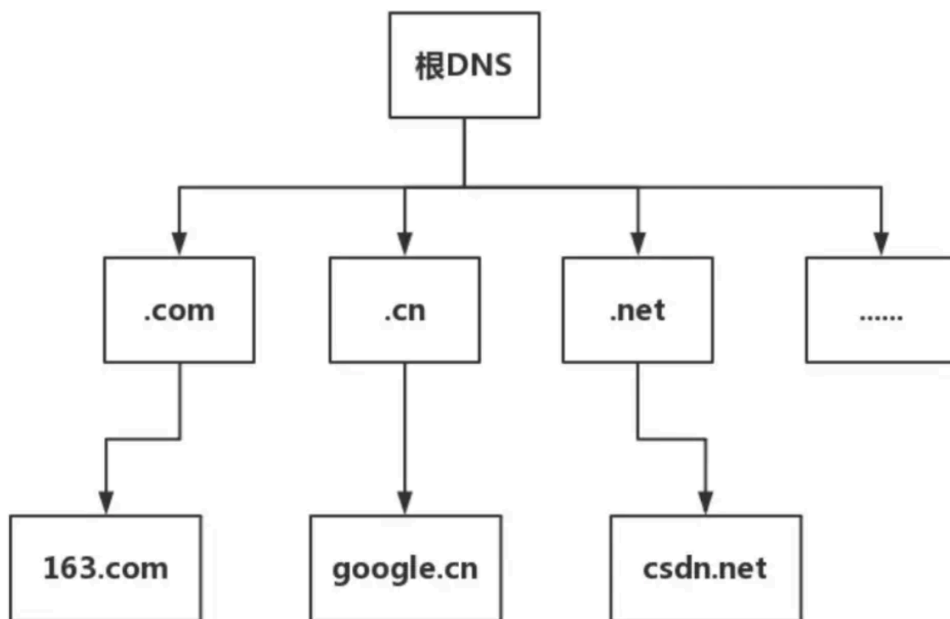
第六步：关闭TCP连接

第七步：浏览器解析资源，渲染布局

具体分析：

1、第一步：域名解析

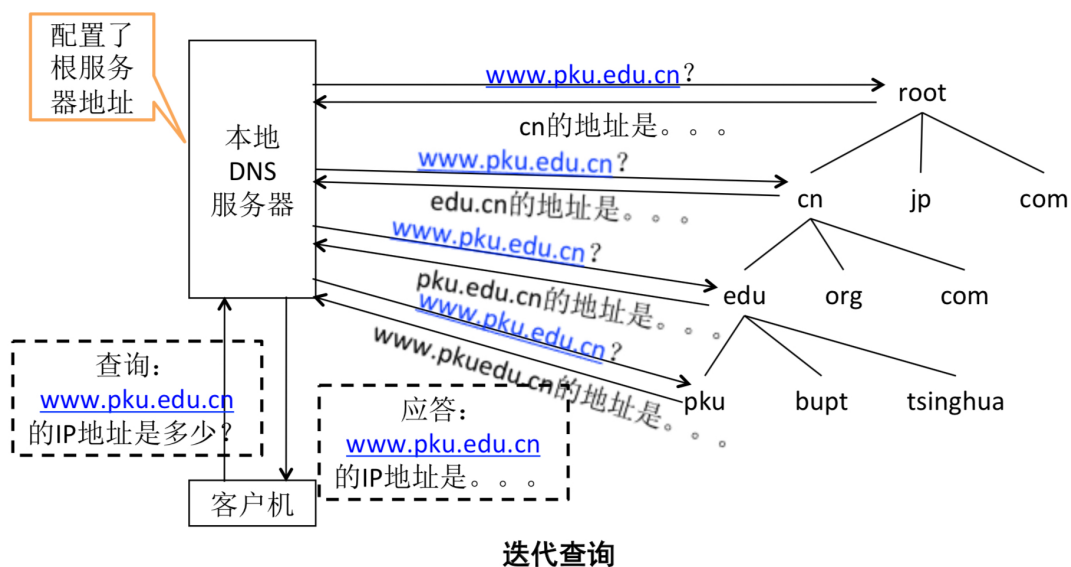
比如北大官网www.pku.edu.cn，浏览器客户端向本地DNS服务器发送一个含有域名www.pku.edu.cn的DNS查询报文，本地DNS服务器把查询报文转发到根DNS服务器，然后逐步溯源，最后用负责该域名的权威DNS服务器的IP地址作为回应。



递归查询DNS的过程：

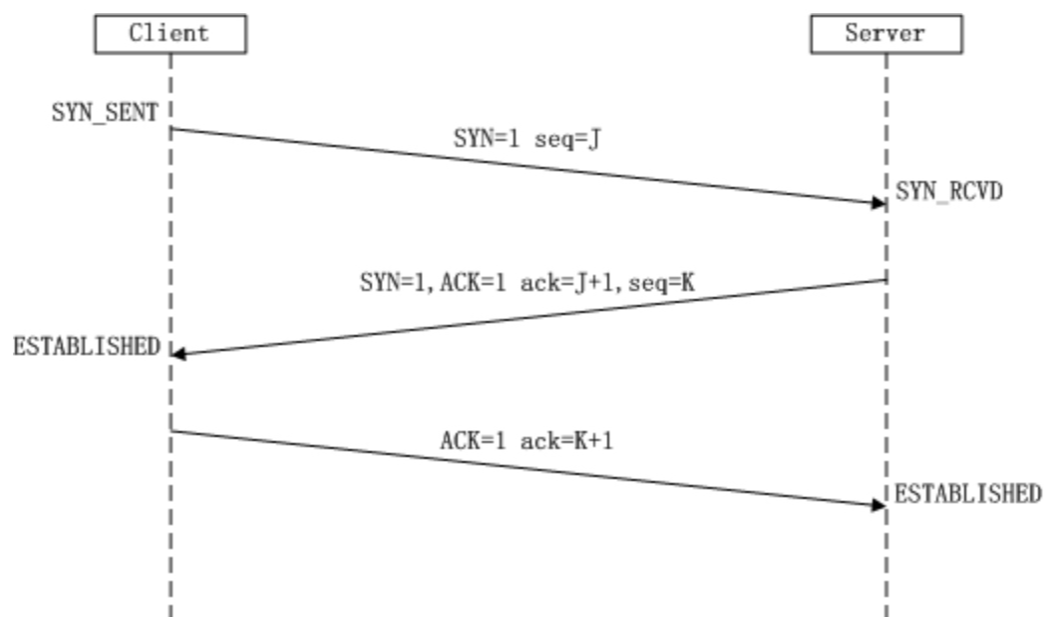
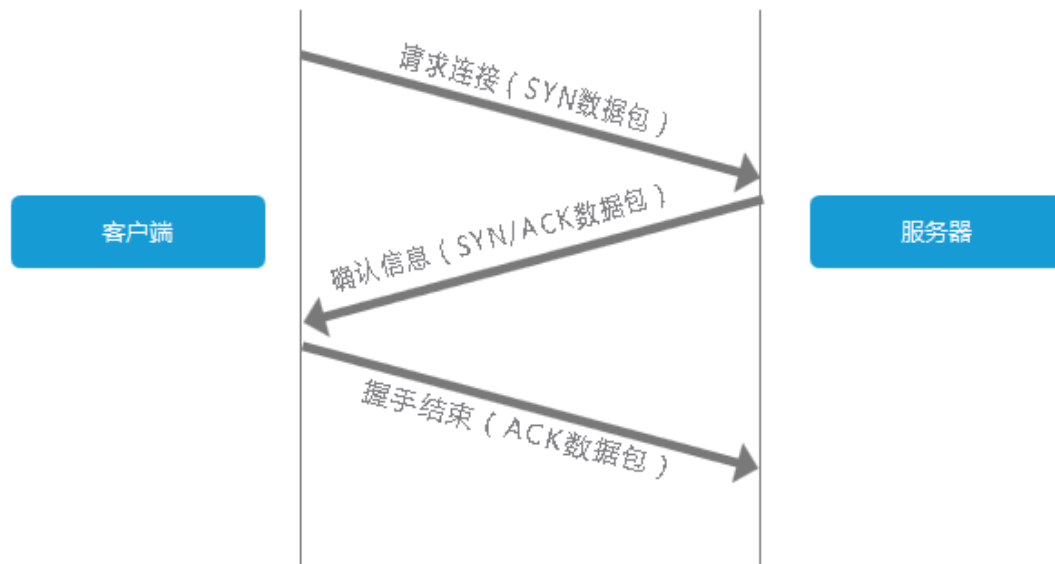


域名的解析过程



2、第二步：建立TCP链接

找到服务器ip后，就是客户端与服务器建立TCP链接，三次握手。

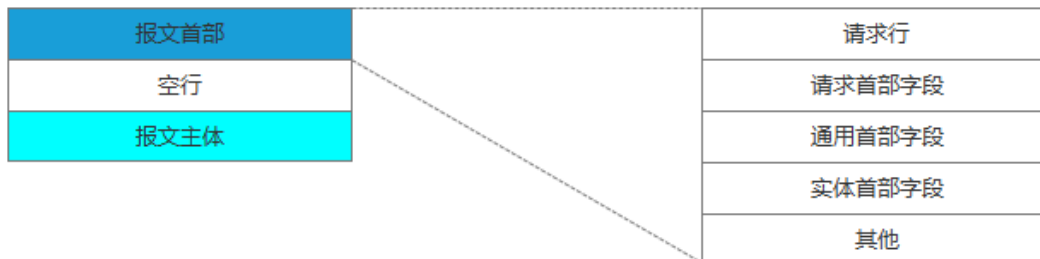


TCP原理详解: <https://note.youdao.com/s/5YtU7S1N>

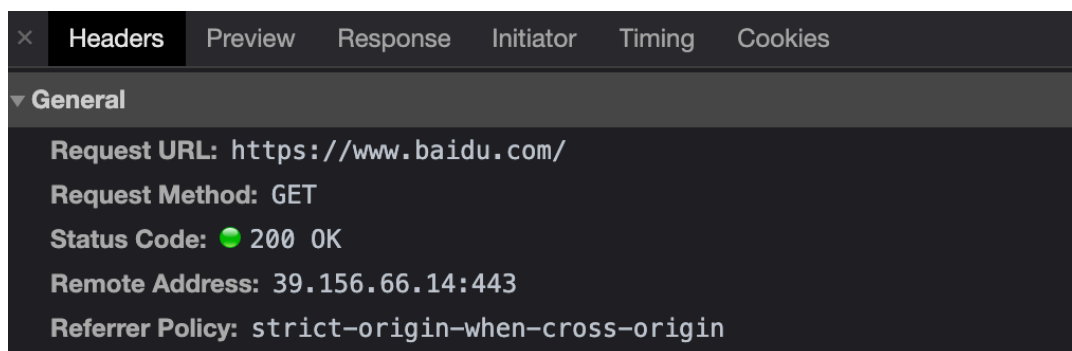
3、第三步：发送HTTP请求

与服务器建立了连接后，就可以向服务器发起请求了。这里看下请求报文的

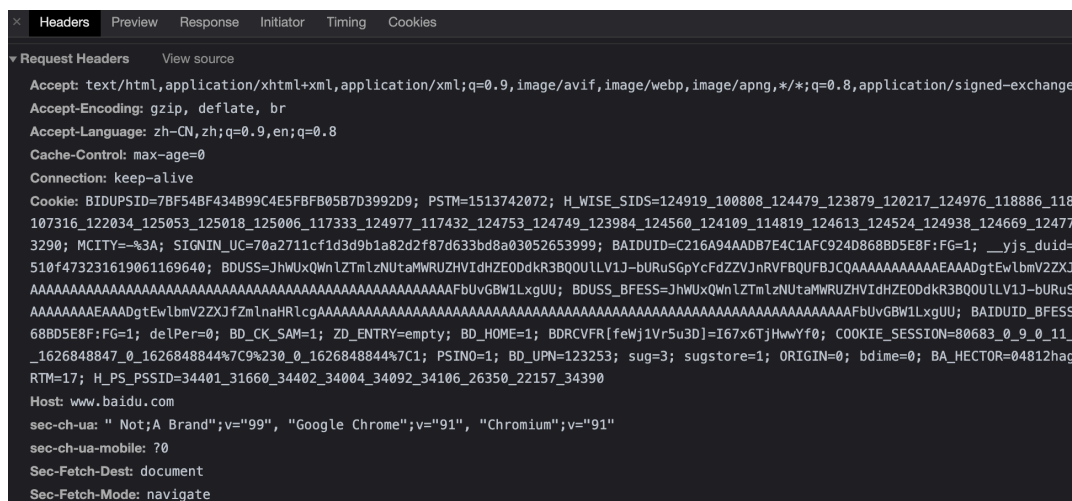
结构：



请求报文，在浏览器中查看报文头，以www.baidu.com为例：



请求行包括：请求方法、URI、HTTP版本。首部字段传递重要信息，包括请求首部字段、通用首部字段和实体首部字段。我们可以从报文中看到发出的请求的具体信息。



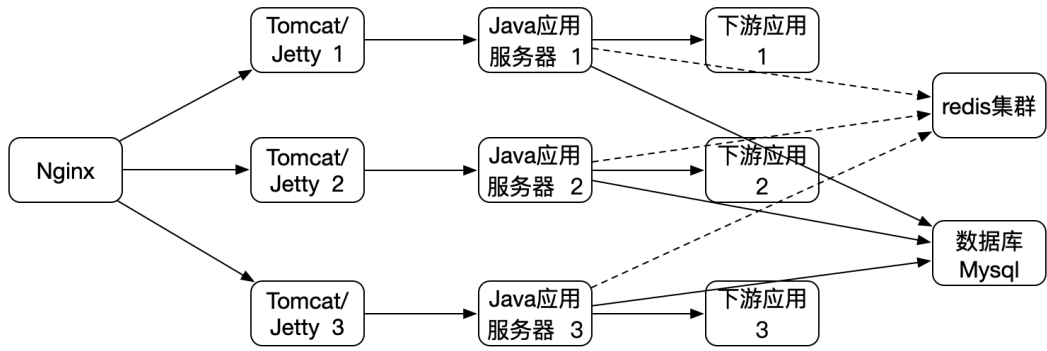
http原理: <https://note.youdao.com/s/d6gMbvYw>

4、第四步：服务器处理请求

如果请求的是静态资源，那么流量有可能到达 CDN 服务器，去取html、js

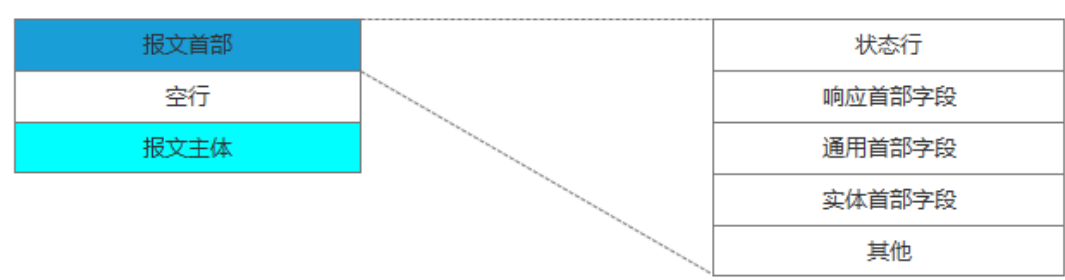
等静态资源渲染页面。

如果请求的是动态资源，那么情况就比较复杂，流量可能依次经过代理/网关、Web 服务器、应用服务器、数据库。

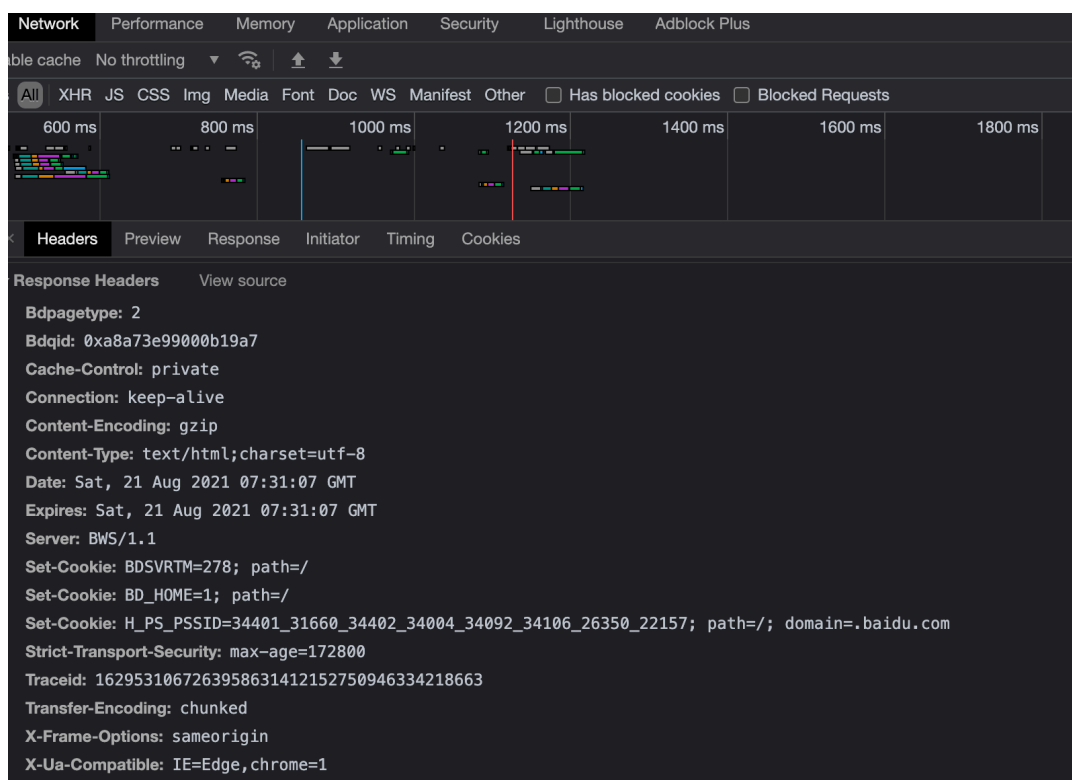


5、第五步：返回响应结果

在HTTP里，有请求就会有响应，哪怕是错误信息。这里看下响应报文的组成结构：



响应报文，在浏览器中查看response，以www.baidu.com为例：



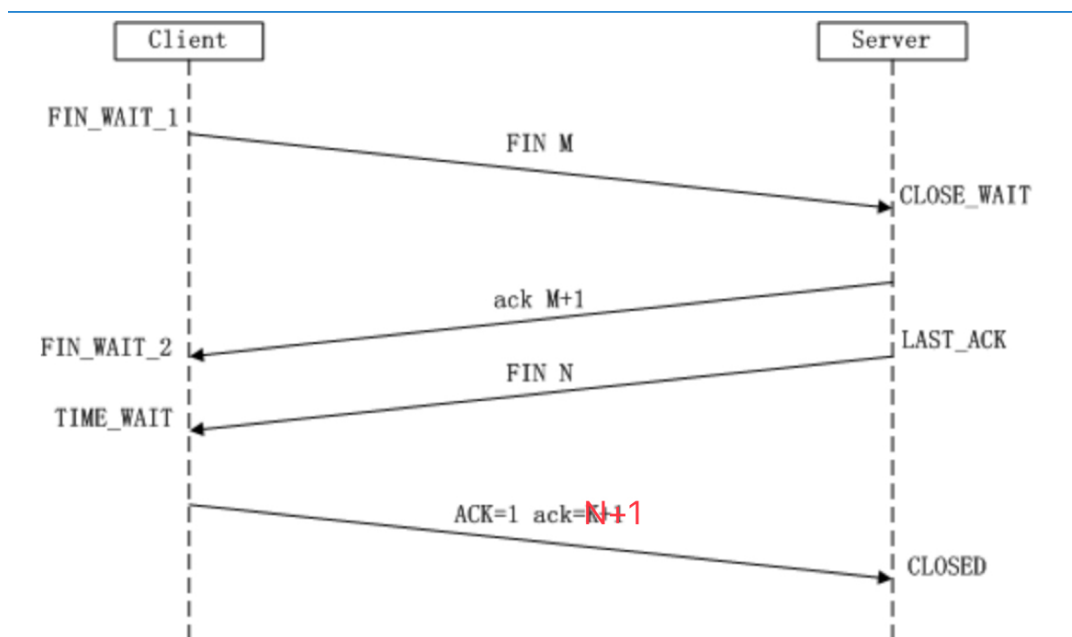
在响应结果中都会有个一个HTTP状态码，比如我们熟知的200、301、404、500等。通过这个状态码我们可以知道服务器端的处理是否正常，并能了解具体的错误。

状态码由3位数字和原因短语组成。根据首位数字，状态码可以分为五类：

	类别	原因短语
1xx	信息性状态码	接收的请求正在处理
2xx	成功状态码	请求正常处理完毕
3xx	重定向状态码	需要进行附加操作以完成请求
4xx	客户端错误状态码	服务器无法处理请求
5xx	服务器错误状态码	服务器处理请求出错

6、第六步：关闭TCP连接

当双方没有请求或响应传递时，任意一方都可以发起关闭请求。关闭TCP连接，需要4次，即四次挥手。

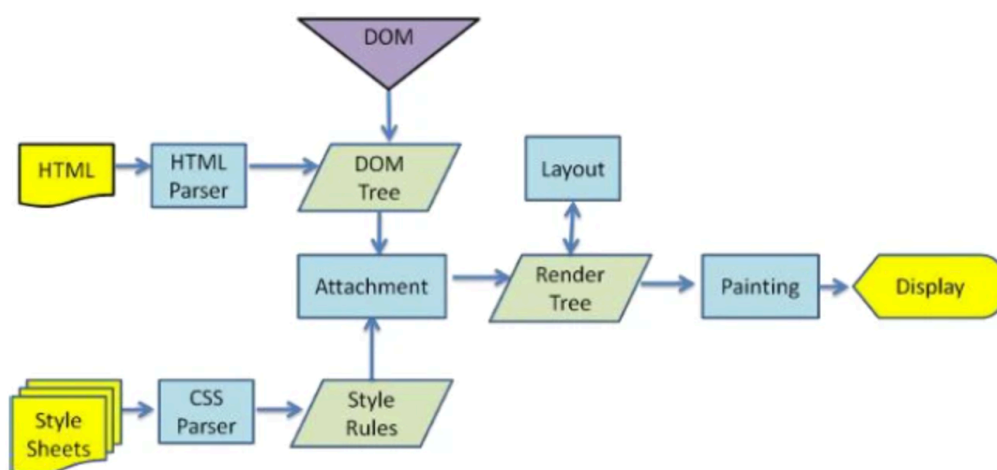


7、第七步：浏览器解析资源，渲染布局

1) 解析资源

浏览器需要加载解析HTML、CSS、JS。以及还要加载图片、视频等其他媒体资源。

浏览器通过解析HTML，生成DOM树，解析CSS，生成CSS规则树，然后通过DOM树和CSS规则树生成渲染树。渲染树与DOM树不同，渲染树中并没有head、display为none等不必显示的节点。



要注意的是，浏览器的解析过程并非是串连进行的，比如在解析CSS的同时，可以继续加载解析HTML，但在解析执行JS脚本时，会停止解析后续

HTML，这就会出现阻塞问题，关于JS阻塞相关问题，这里不过多阐述,后面会单独开篇讲解。

2) 渲染布局

根据渲染树布局，计算CSS样式，即每个节点在页面中的大小和位置等几何信息。HTML默认是流式布局的，CSS和JS会打破这种布局，改变DOM的外观样式以及大小和位置。这时就要提到两个重要概念：replaint和reflow。

- replaint：屏幕的一部分重画，不影响整体布局，比如某个CSS的背景色变了，但元素的几何尺寸和位置不变。
- reflow：意味着元件的几何尺寸变了，我们需要重新验证并计算渲染树。是渲染树的一部分或全部发生了变化。这就是Reflow，或是Layout。

所以我们应该尽量减少reflow和replaint，我想这也是为什么现在很少有用table布局的原因之一。

最后浏览器绘制各个节点，将页面展示给用户。

