

 稀土掘金

课程

三 前端面试之道



26 UDP

已学完 学习时长: 3分11秒

27 TCP

已学完 学习时长: 10分1秒

28 HTTP 及 TLS

已学完 学习时长: 10分38秒

29 HTTP/2 及 HTTP/3

已学完 学习时长: 5分18秒

30 输入 URL 到页面渲染的整个流程

已学完 学习时长: 2分48秒

31 设计模式

已学完 学习时长: 18分36秒

32 常见数据结构

已学完 学习时长: 100分51秒

33 常考算法题解析

学习时长: 65分31秒

34 如何写好一封简历

已学完 学习时长: 3分29秒

35 面试常用技巧

已学完 学习时长: 3分1秒

输入 URL 到页面渲染的整个流程

之前我们学了那么多章节的内容，是时候找个时间将它们再次复习消化了。就借用这道经典面试题，将之前学习到的浏览器以及网络几章节的知识联系起来。

首先是 DNS 查询，如果这一步做了智能 DNS 解析的话，会提供访问速度最快的 IP 地址回来，这部分的内容之前没有写过，所以就在这里讲解下。

DNS

DNS 的作用就是通过域名查询到具体的 IP。

因为 IP 存在数字和英文的组合（IPv6），很不利于人类记忆，所以就出现了域名。你可以把域名看成是某个 IP 的别名，DNS 就是去查询这个别名的真正名称是什么。

在 TCP 握手之前就已经进行了 DNS 查询，这个查询是操作系统自己做的。当你在浏览器中想访问 `www.google.com` 时，会进行一下操作：

- 操作系统会首先在本地缓存中查询 IP
- 没有的话会去系统配置的 DNS 服务器中查询
- 如果这时候还没得话，会直接去 DNS 根服务器查询，这一步查询会找出负责 `com` 这个一级域名的服务器
- 然后去该服务器查询 `google` 这个二级域名
- 接下来三级域名的查询其实是我们配置的，你可以给 `www` 这个域名配置一个 IP，然后还可以给别的三级域名配置一个 IP

以上介绍的是 DNS 迭代查询，还有种是递归查询，区别就是前者是由客户端去做请求，后者是由系统配置的 DNS 服务器做请求，得到结果后将数据返回给客户端。

PS：DNS 是基于 UDP 做的查询，大家也可以考虑下为什么之前不考虑使用 TCP 去实现。

接下来是 TCP 握手，应用层会下发数据给传输层，这里 TCP 协议会指明两端的端口号，然后下发给网络层。网络层中的 IP 协议会确定 IP 地址，并且指示了数据传输中如何跳转路由器。然后包会再被封装到数据链路层的数据帧结构中，最后就是物理层面的传输了。

在这一部分中，可以详细说下 TCP 的握手情况以及 TCP 的一些特性。

当 TCP 握手结束后就会进行 TLS 握手，然后就开始正式的传输数据。

在这一部分中，可以详细说下 TLS 的握手情况以及两种加密方式的内容。

数据在进入服务端之前，可能还会先经过负责负载均衡的服务器，它的作用就是将请求合理的分发到多台服务器上，这时假设服务端会响应一个 HTML 文件。

首先浏览器会判断状态码是什么，如果是 200 那就继续解析，如果 400 或 500 的话就会报错，如果 300 的话会进行重定向，这里会有个重定向计数器，避免过多次的重定向，超过次数也会报错。

浏览器开始解析文件，如果是 gzip 格式的话会先解压一下，然后通过文件的编码格式知道该如何去解码文件。

文件解码成功后会正式开始渲染流程，先会根据 HTML 构建 DOM 树，有 CSS 的话会去构建 CSSOM 树。如果遇到 script 标签的话，会判断是否存在 async 或者 defer，前者会并行进行下载并执行 JS，后者会先下载文件，然后等待 HTML 解析完成后顺序执行。

如果以上都没有，就会阻塞住渲染流程直到 JS 执行完毕。遇到文件下载的会去下载文件，这里如果使用 HTTP/2 协议的话会极大的提高多图的下载效率。

CSSOM 树和 DOM 树构建完成后会开始生成 Render 树，这一步就是确定页面元素的布局、样式等等诸多方面的东西

在生成 Render 树的过程中，浏览器就开始调用 GPU 绘制，合成图层，将内容显示在屏幕上了。

这一部分就是渲染原理中讲解到的内容，可以详细的说明下这一过程。并且在下载文件时，也可以说下通过 HTTP/2 协议可以解决队头阻塞的问题。

总的来说这一章节就是带着大家从 DNS 查询开始到渲染出画面完整的了解一遍过程，将之前学习到的内容连接起来。

当来这一过程远远不止这些内容，但是对于大部分人能答出这些内容已经很不错了，你如果想了解更加详细的过程，可以阅读这篇文章。

留言



输入评论（Enter换行，`⌘` + Enter发送）

发表评论

全部评论（22）



不会做蛋糕的程序员

前端开发 | 1月前

越来越水了，前面还讲讲原理，后面的都是点到为止

👍 2

🗨 回复



Adolph_lala

前端小菜鸟儿 | 2月前

小册介绍、开篇里边都说了这只是个check list，需要自己去发散

👍 点赞

🗨 回复



专一且有才华

切图仔 | 2月前

你是真的水

👍 1

🗨 回复



砂糖小刀

暂无 | 1年前

我寻思这篇小册就是用来面试的，又不是百科全书各种知识点都给你讲那么深，就是抛砖引玉让自己知道需要掌握哪些知识点根据自身情况查漏补缺而已，说白了小册能应付面试就行，作者也说过这篇小册是应对中小型公司的，中小型公司你能在面试的时候说出这些已经完全够了，觉得水，太简单，那说明你的水平和目标得是月薪上万的一线大厂了，那你还来看这种初级小册？就这么说，买来一本初中数学指责它里面不讲微积分不觉得搞笑吗？

👍 11

🗨 回复



下雨天DY

前端研发工程师 @ 小米 | 1年前

太水了

👍 1

🗨 回复



用户696469358...

前端开发 @ 美团 | 1年前

确实后面越来越水了

👍 点赞

🗨 回复



澜曦

前端开发 @ 500彩票 | 1年前

说句实话，后面整理的这些真的越来越水了，既然决定了写小册就要对内容负责，这样做只是坏了自己的名声！

👍 2

🗨 回复



最美大魔王

前端小菜鸡 @ 携程 | 2年前

没必要懵吧，全部写一下呗，前面这么具体，后面一下都没了。虎头蛇尾都算不上。

👍 3

🗨 回复



你若像风

前端开发工程师 | 2年前

讲得很好。能把这些讲清楚已经很好了。

👍 1

🗨 回复



lees

前端开发工程师 | 2年前

后面整理的这些真的越来越水了.....

👍 4

🗨 回复



Lstoryc君

| 2年前

DNS 是基于 UDP 做的查询，大家也可以考虑下为什么之前不考虑使用 TCP 去实现。现在也有些会使用 TCP

👍 1

🗨 回复



toored

| 2年前

条理性、层次感太差了，感觉很应付，重点也不突出

👍 点赞

🗨 回复



h5rtongqi

前端开发工程师 | 2年前

差评

👍 点赞

🗨 回复



别来无恙、

| 3年前

AST这个在什么时候出现的

👍 点赞

🗨 回复



jhongLjun

前端工程师 | 3年前

具体又不全面。

👍 1

🗨 回复



Genesis同学

| 3年前

就不能先把答案写出来再讲解吗 洋洋洒洒一大篇也不知道面试的时候怎么说答案比较好

👍 17

🗨 回复



李符聿

前端开发 | 4年前

如果能有序排列一下更好，总感觉像是读作文。条理性不是很直观。

👍 18

🗨 回复



__RDY

字节跳动 | 4年前

这一过程涉及到浏览器缓存机制么。好像没有看到。

👍 3

🗨 回复



易水寒YSP

| 4年前

手动点赞~

👍 1

🗨 回复



ChrisSong

前端开发 | 4年前

这个写的太一般了，简单的都有，深入的都没有，只起了一个指导意义的作用

👍 18

🗨 回复

查看全部 22 条回复