



从前端小工到中级工程师的必备技能

课前检验讲解

先过一下作业

1. 如何实现一个三栏布局，要求两边固定宽度，中间自适应

主要用一道css题考查下大家css用的咋样

1 圣杯布局

2 双飞翼布局

3 flex 布局

2. 说一下自己对 Javascript 变量提升的理解

```
1  function fn(){
2      var x = 1
3      var y = 2
4  }
5  fn()
```

1. 进入 fn，为 fn 创建一个环境。
2. 找到 fn 中所有用 var 声明的变量，在这个环境中「创建」这些变量（即 x 和 y）。
3. 将这些变量「初始化」为 undefined。
4. 开始执行代码
5. x = 1 将 x 变量「赋值」为 1
6. y = 2 将 y 变量「赋值」为 2

```
1    fn2()  
2  
3    function fn2(){  
4        |    console.log(2)  
5    }
```

1. 找到所有用 function 声明的变量，在环境中「创建」这些变量。
2. 将这些变量「初始化」并「赋值」为 function(){ console.log(2) }。
3. 开始执行代码 fn2()

```
1  {  
2    let x = 1  
3    x = 2  
4  }
```

1. 找到所有用 let 声明的变量，在环境中「创建」这些变量
2. 开始执行代码（注意现在还没有初始化）
3. 执行 `x = 1`，将 `x` 「初始化」为 1（这并不是一次赋值，如果代码是 `let x`，就将 `x` 初始化为 `undefined`）
4. 执行 `x = 2`，对 `x` 进行「赋值」


```
1  let x = 'global'
2  {
3    console.log(x) // Uncaught ReferenceError: x is not defined
4    let x = 1
5  }
```

- console.log(x) 中的 x 指的是下面的 x，而不是全局的 x
- 执行 log 时 x 还没「初始化」，所以不能使用（也就是所谓的暂时死区）

《我用了两个小时才理解let》

4. 下面代码会打印什么？

```
1  function A() {  
2      this.name = 'Daniel'  
3  }  
4  A.prototype.test = function () {  
5      setTimeout(function () {  
6          console.log(this.name)  
7      }, 1000)  
8  }  
9  var a = new A()  
10 a.test()
```

this永远指向函数运行时所在的对象！
而不是函数被创建时所在的对象。
也即：谁调用，指向谁。

```
1  var name = '全局'
2  function getName() {
3      var name = '局部'
4      return this.name
5  };
6  alert(getName())
```

没有明确的当前对象时的this

```
1  var name = 'Bob'
2  var nameObj = {
3      name: 'Tom',
4      showName: function () {
5          alert(this.name)
6      },
7      waitShowName: function () {
8          setTimeout(this.showName, 1000)
9      }
10 };
11
12 nameObj.waitShowName()
```

← → ↻ 安全 <https://www.baidu.com> ☆ 扩展 广告 更多

新闻 hao123 地图 视频 贴吧 学术 登录 设置 更多产品

Baidu 百度

百度一下

Memory Elements Console Network Sources Performance Security Application >> ⋮ ×

top ▼ Filter Info ▼ ⚙

```
> window.name
< ""

> window.name = 'baidu'
< "baidu"

>
```



课程 ▾

搜索课程



IT · 互联网 前端开发 Java >

设计 · 创作 平面设计 UI设计 >

语言 · 留学 实用英语 雅思 >

职业 · 考证 公务员 教师考试 >

跟世界500强CEO林正刚

学职场晋升之道

—— 职业经理人的能力框架



正在连接...

🔍 📄 | Memory Elements **Console** Network Sources Performance Security Application >> | ⋮ ×

🔇 top ▼ | Filter Info ▼ | ⚙️

> window.name

< "baidu"

>

这个神奇的特性，被用来解决跨域数据传递

5. 如何实现类似 jQuery 的链式调用?

```
1  class Core {  
2      constructor() {}  
3      say(word) {  
4          console.log(word)  
5          return this  
6      }  
7  }  
8  
9  const core = new Core  
10  
11  core.say('hello')  
12  |   .say('world')
```

6. Yslow 和 PageSpeed 你知道怎么用吗?
你记得其中多少规则?

7. HTTP请求头中，状态码分别是：200、301、302、304 分别代表什么意思？

8. PNG 和 JPG 有什么差别?

图片格式	支持透明	动画支持	压缩方式	浏览器支持	相对原图大小	适应场景
baseline-jpeg	不支持	不支持	有损	所有	由画质决定	所有通用场景
progressive-jpeg	不支持	不支持	有损	所有	由画质决定	所有通用场景, 渐进式加载
gif	支持	支持	无损	所有	由帧数和每帧图片大小决定	简单颜色, 动画
png	支持	不支持	无损	所有	由png色值位数决定	需要透明时
webp	支持	不支持	有损	Chrome、Opera	由压缩率决定	复杂颜色及形状, 浏览器平台可预知
apng	支持	支持	无损	Firefox、Safari、iOS Safari	由每帧图片决定	需要半透明效果的动画
svg	支持	支持	无损	所有(IE8以上)	由内容和特效复杂度决定	简单图形, 需要良好的放缩体验, 需要动态控制图片特效
bpg	支持	支持	有损	不支持, 需要js解码	由画质决定	jpeg上需要极限优化的场景

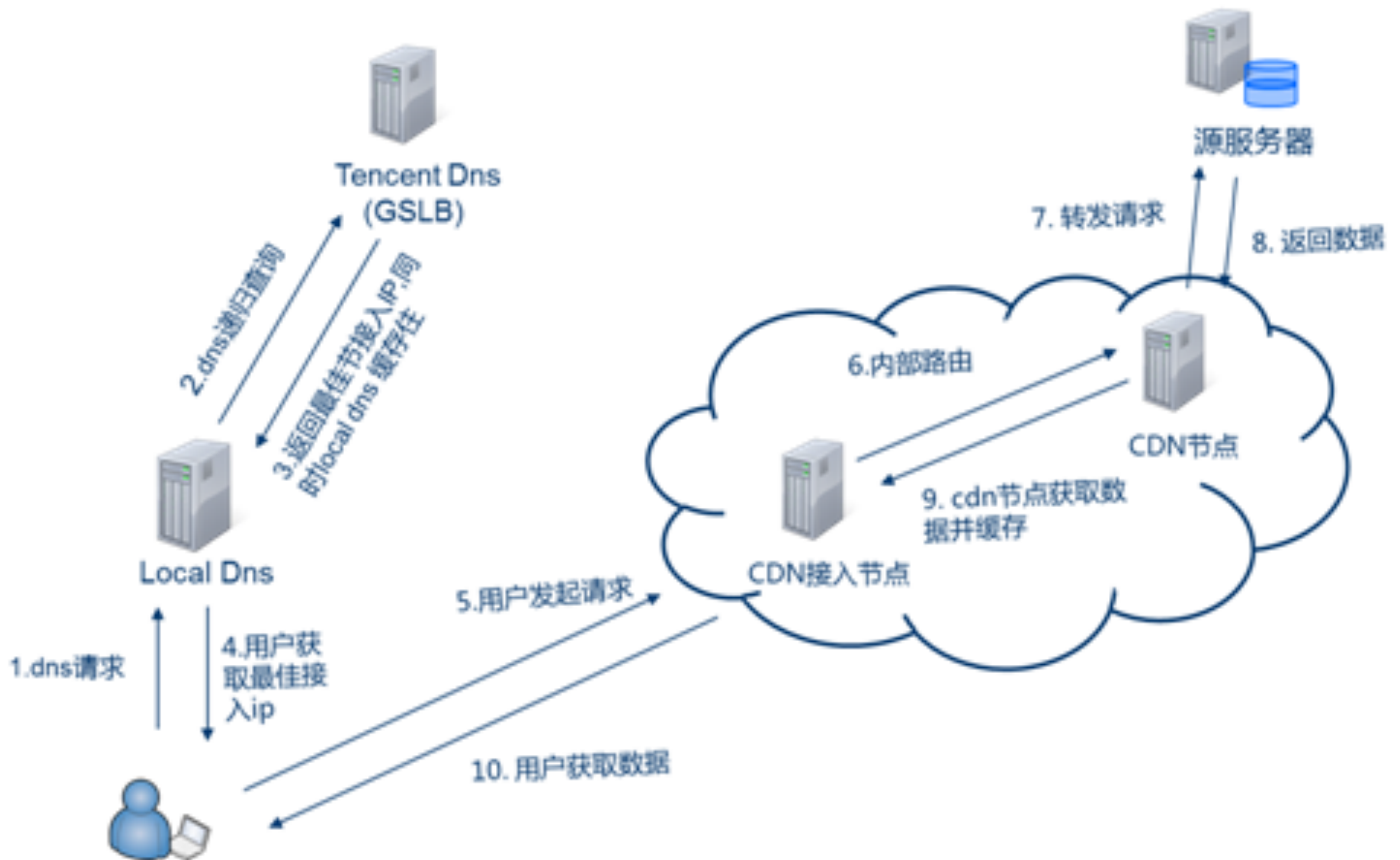
《前端图片优化机制》

9. 事件冒泡和事件捕获有什么区别?

10. 你遇到过兼容性问题吗？
最令你印象深刻的兼容问题是什么？

11. 你使用过构建工具来编译前端代码吗？你使用过 Grunt、Gulp、webpack、FIS 其中之一吗？

12. CDN是什么？为什么能资源加速？

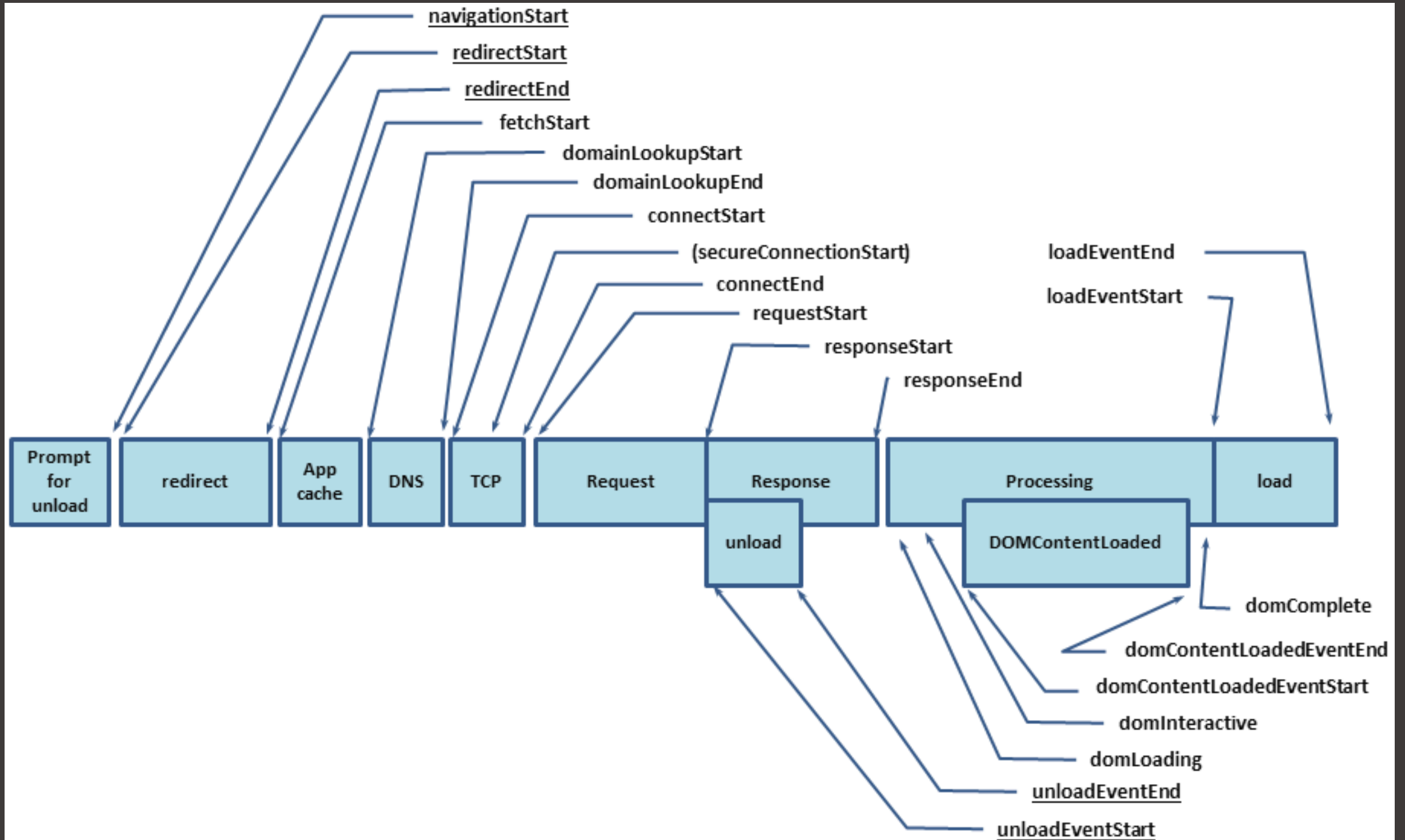


话说，其实CDN能做动态加速！

13. 你使用过React、Vue、Angular其中之一吗?

14. 你知道前端怎么样能拿到DNS的查询时间吗?

performance.timing



多普勒测速

$t_1 - t_2 = \text{DNS}$

$t_2 - t_3 = \text{TCP/IP}$

$T_3 = \text{RTT}$

多普勒测速方案：

- http://a-doppler.facebook.com/test_pixel?HTTP1.0&t=1&size=0k

$t_1 = \text{DNS} + \text{TCP/IP} + \text{RTT}$

- http://a-doppler.facebook.com/test_pixel?HTTP1.1&t=2&size=0k

$t_2 = \text{TCP/IP} + \text{RTT}$

- http://a-doppler.facebook.com/test_pixel?HTTP1.1&t=3&size=0k

$t_3 = \text{RTT}$

- http://a-doppler.facebook.com/test_pixel?HTTP1.1&t=4&size=10k

$10k / (t_4 - t_3) \approx \text{bandwidth}$

15. 你能拿 localStorage 来做些什么？

注意 localStorage 是不可信的，
数据可能损坏丢失，所以不要把重要的数据存进去，
并且做好容错



THANKS!



扫码了解更多