

拉勾教育

— 互联网人实战大学 —

# 《前端高手进阶》

朱德龙 前中兴软创主任工程师

— 拉勾教育出品 —

## 第23讲：谈性能优化到底在谈什么？

**性能**是前端领域关注度非常高的话题

页面性能的好坏会直接影响用户体验

前端工程师往往会对页面性能不断改进

这个改进的过程就叫**性能优化**

# 性能指标

拉勾教育

— 互联网人实战大学 —

所谓**性能**是指程序的运行速度

而前端性能是指页面的响应速度

性能指标都是以时间为单位来测量的



## 衡量前端性能的指标

- 是否可以通过浏览器采集上报
- 是否由权威组织或大型公司提出
- 以及是否严重影响用户体验



# 首屏绘制 (First Paint, FP)

拉勾教育

— 互联网人实战大学 —

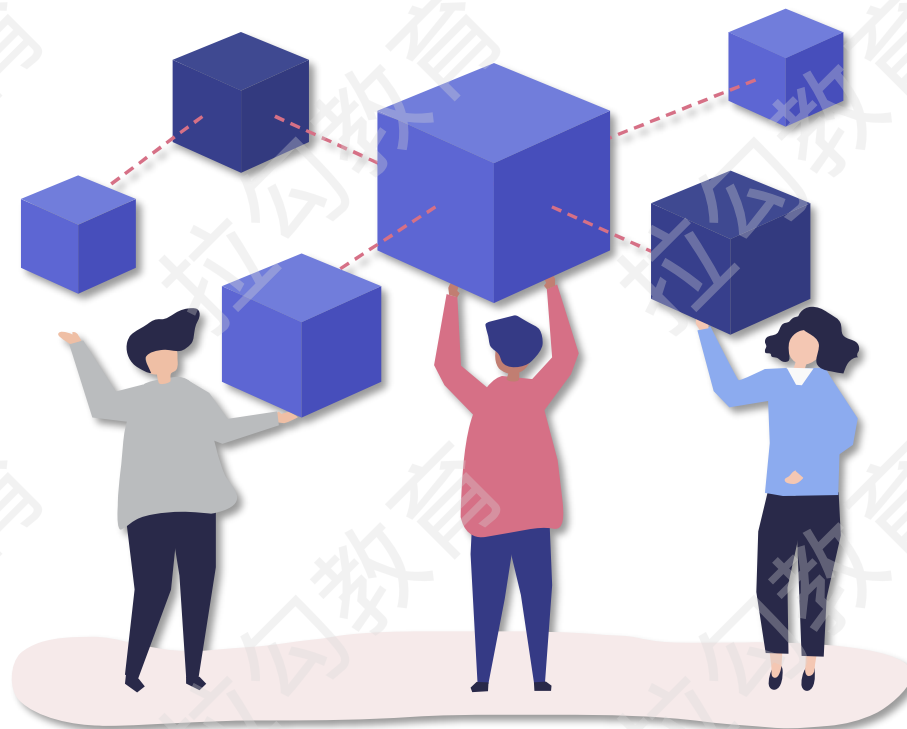
首屏绘制由 W3C 标准 **Paint Timing** 中提出

首屏绘制时间是指

从开始加载到浏览器首次绘制像素到屏幕上的时间

也就是页面在屏幕上首次发生视觉变化的时间

俗称 **“白屏时间”**



# 首屏绘制 (First Paint, FP)

拉勾教育

— 互联网人实战大学 —

```
performance.getEntriesByType('paint')[0]
/*
{
  duration: 0,
  entryType: "paint",
  name: "first-paint",
  startTime: 197.58499998715706,
}
*/
```

# 首屏内容绘制（First Contentful Paint, FCP）

拉勾教育

— 互联网人实战大学 —

首屏内容绘制由 W3C 标准 Paint Timing 中提出

浏览器首次绘制来自 DOM 的内容时间

这个内容可以是文字、图片（也包括背景图片）、非空白的 canvas 和 svg





# 首屏内容绘制 (First Contentful Paint, FCP)

拉勾教育

— 互联网人实战大学 —

```
performance.getEntriesByType('paint')[1]
/*
{
  duration: 0,
  entryType: "paint",
  name: "first-contentful-paint",
  startTime: 797.8649999859044
}
*/
```

# 可交互时间 (Time to Interactive, TTI)

拉勾教育

— 互联网人实战大学 —

可交互时间由 **Web 孵化器社区组 (WICG)** 提出

是指网页在视觉上都已渲染出了

浏览器可以响应用户的操作



# 可交互时间 (Time to Interactive, TTI)

拉勾教育

— 互联网人实战大学 —

- 主线程的长任务（长任务是指耗时超过 50 ms）执行完成后
- 随后网络静默时间达到 5 秒

## 静默时间

指请求数不超过 2 个

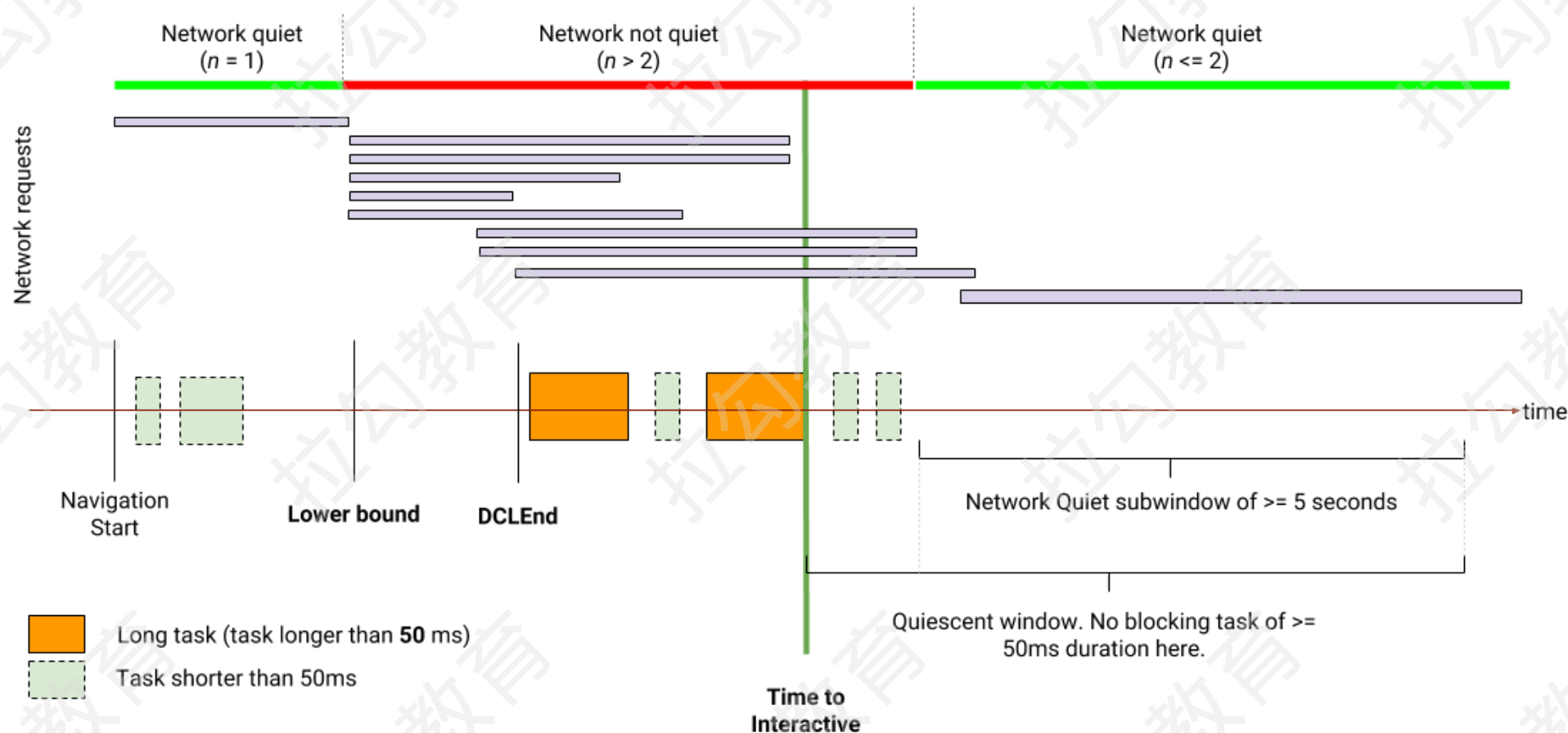
排除失败的资源请求和未使用 GET 方法进行的网络请求



# 可交互时间 (Time to Interactive, TTI)

拉勾教育

— 互联网人实战大学 —



TTI 示意图

# 可交互时间 (Time to Interactive, TTI)

拉勾教育

— 互联网人实战大学 —

```
import ttiPolyfill from 'tti-polyfill';

ttiPolyfill.getFirstConsistentlyInteractive(opts).then((tti) => {
  ...
});
```

# 总阻塞时间 (Total Blocking Time, TBT)

拉勾教育

— 互联网人实战大学 —

总阻塞时间由 W3C 标准 **Long Tasks API 1** 提出

指阻塞用户响应（比如键盘输入、鼠标点击）的所有时间

**指标值**是将 FCP 之后一直到 TTI 这段时间内的阻塞部分时间总和

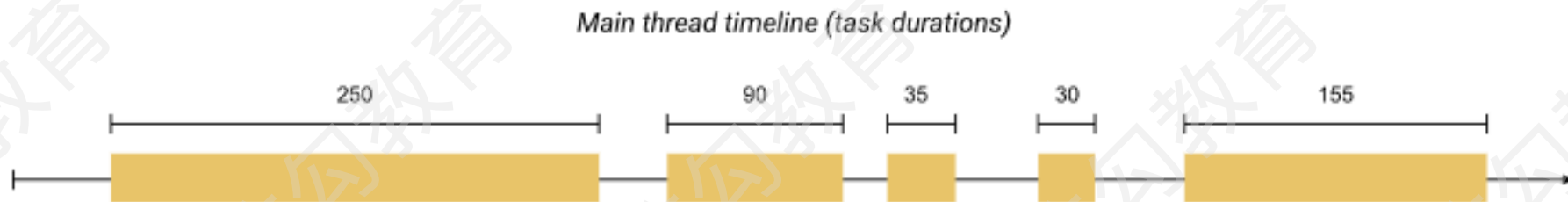
**阻塞部分**是指长任务执行时间减去 50 毫秒



# 总阻塞时间 (Total Blocking Time, TBT)

拉勾教育

— 互联网人实战大学 —



# 总阻塞时间 (Total Blocking Time, TBT)

拉勾教育

— 互联网人实战大学 —

```
var observer = new PerformanceObserver(function (list) {  
  var perfEntries = list.getEntries();  
  for (var i = 0; i < perfEntries.length; i++) {  
    console.log(perfEntries[i].toJSON())  
  }  
  /*  
  {  
    attribution: [TaskAttributionTiming]?  
    duration: 6047.7700000004675?  
    entryType: "longtask"?  
    name: "self"?  
    startTime: 22.4449999995059334
```



# 总阻塞时间 (Total Blocking Time, TBT)

拉勾教育

— 互联网人实战大学 —

```
duration: 6047.770000004675?  
entryType: "longtask"?  
name: "self"?  
startTime: 22.444999995059334  
}  
*/  
}  
});  
  
observer.observe({  
  entryTypes: ["longtask"]  
});
```

# 最大内容绘制 (Largest Contentful Paint, LCP)

拉勾教育

— 互联网人实战大学 —

最大内容绘画指的是视口内可见的最大图像或文本块的绘制时间

测量这个指标的值和 TBT 相似

不同的是将实体类型改为 “largest-contentful-paint”



# 最大内容绘制 (Largest Contentful Paint, LCP)

拉勾教育

— 互联网人实战大学 —

```
var observer = new PerformanceObserver(function (list) {  
  var perfEntries = list.getEntries();  
  for (var i = 0; i < perfEntries.length; i++) {  
    console.log(perfEntries[i].toJSON())  
    /*  
    {  
      duration: 0,  
      element: img,  
      entryType: "largest-contentful-paint",  
      id: "",  
      loadTime: 274.864,  
      name: "img",  
      startTime: 274.864,  
      target: "img",  
      type: "Image",  
      value: 0,  
      weight: 1  
    }  
    */  
  }  
})
```

# 最大内容绘制 (Largest Contentful Paint, LCP)

拉勾教育

— 互联网人实战大学 —

```
loadTime: 274.864,  
name: "",  
renderTime: 0,  
size: 2502,  
startTime: 274.864,  
url:  
"https://www.google.com/images/branding/googlelogo/1x/googlelogo_color_272x92dp.png"  
}  
*/  
}
```

# 最大内容绘制 (Largest Contentful Paint, LCP)

拉勾教育

— 互联网人实战大学 —

```
renderTime: 0,  
size: 2502,  
startTime: 274.864,  
  url:  
    "https://www.google.com/images/branding/googlelogo/1x/googlelogo_color_272x92dp.png"  
}  
*/  
}  
});  
observer.observe({entryTypes: ['largest-contentful-paint']});
```

不同的用户、不同的环境采集同一指标值会有所差异  
通常需要对大量采集的性能指标数据进行统计才能用来量化

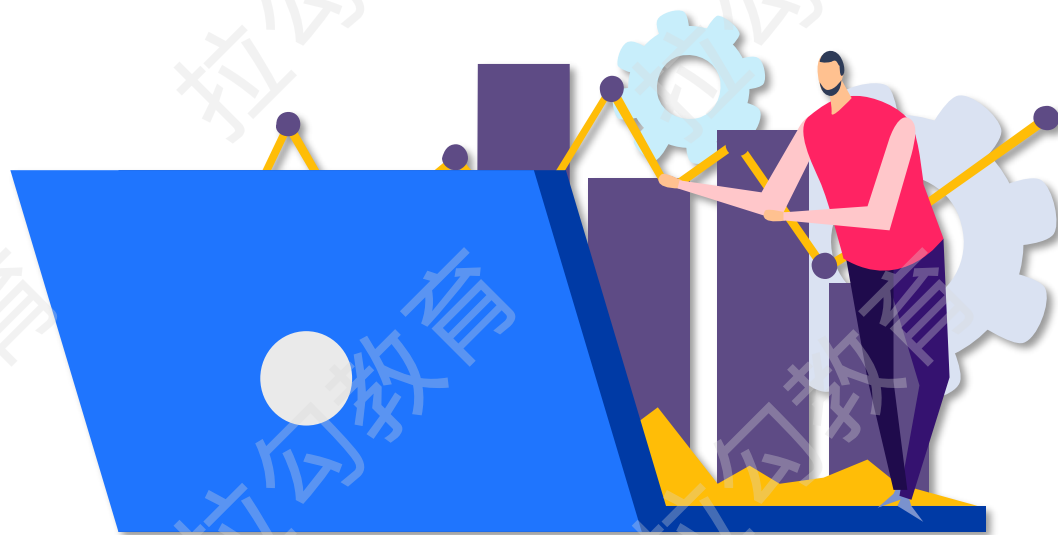
# 平均值统计

拉勾教育

— 互联网人实战大学 —

将所有用户产生的性能指标值收集起来  
然后对这些数据取平均值  
最终得到平均耗时数据

最大的问题就是**容易受极值影响**



# 百分位数统计

拉勾教育

— 互联网人实战大学 —

百分位数统计可以解决极值问题

百分位数是对应于百分位的实际数值





# 百分位数统计

拉勾教育

— 互联网人实战大学 —

比如**第 70 百分位数**:

将数据从小到大排列

处于第 70% 的数据称为 70 分位数

表示 70% 的性能数据均小于等于该值

那剩下的 30% 的数据均大于等于该值了



# 百分位数统计

拉勾教育

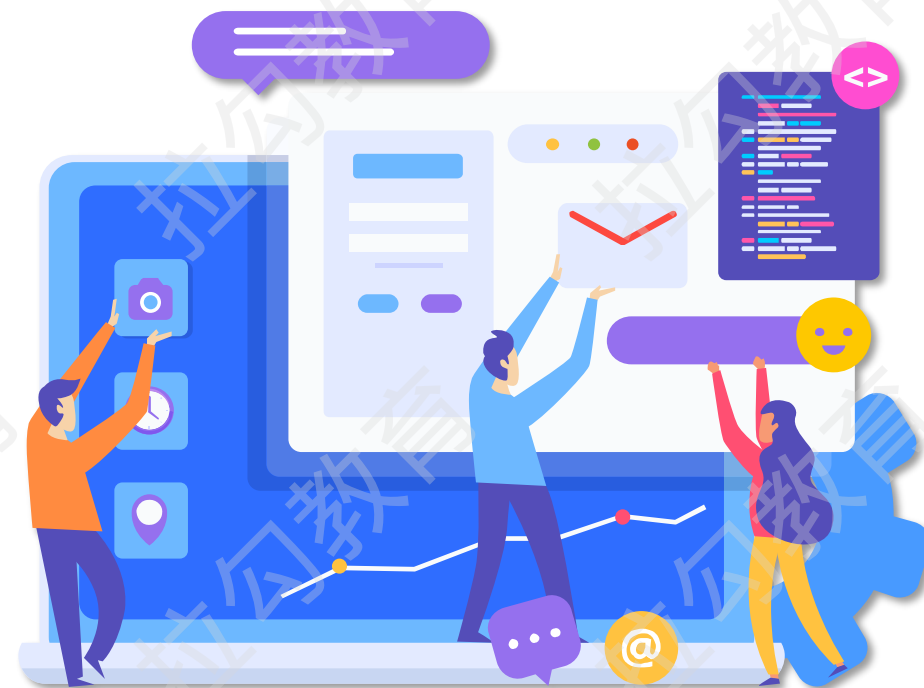
— 互联网人实战大学 —

## 百分位数的好处：

对于性能需求不同的页面或应用

可以设置不同的百分位数

对性能要求越高使用越大的百分位数



前端性能优化一般分为**加载性能优化**和**渲染性能优化**

- **做减法**是直接减少耗时操作或资源体积
- **做除法**是在耗时操作和资源体积无法减少的情况下  
对其进行拆分处理或者对不可拆分的内容进行顺序调换



## 加载性能的优化手段中做减法

- 采用 gzip 压缩
- 使用缓存
- 使用雪碧图



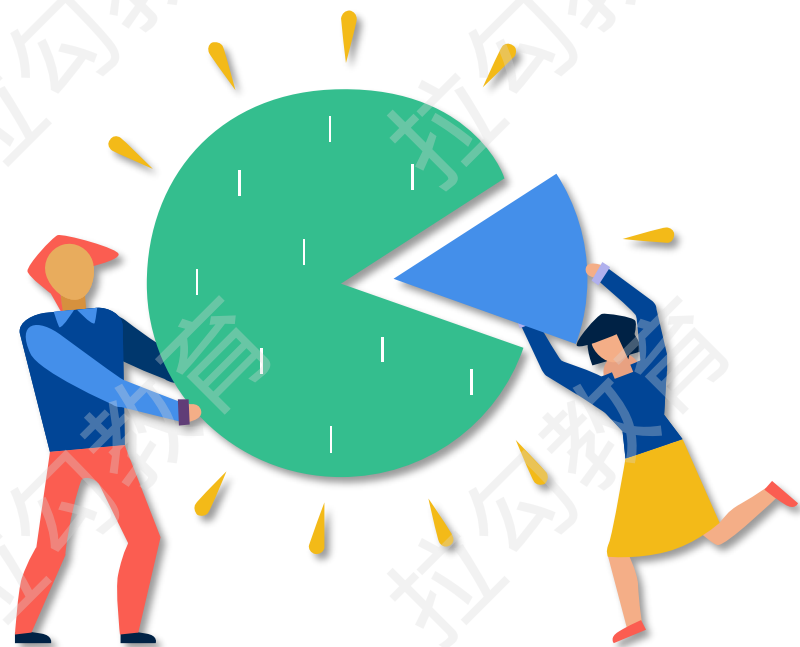
## 加载性能的优化手段中做除法

- HTTP2 多路复用
- 懒加载
- 把 script 标签放到 body 底部



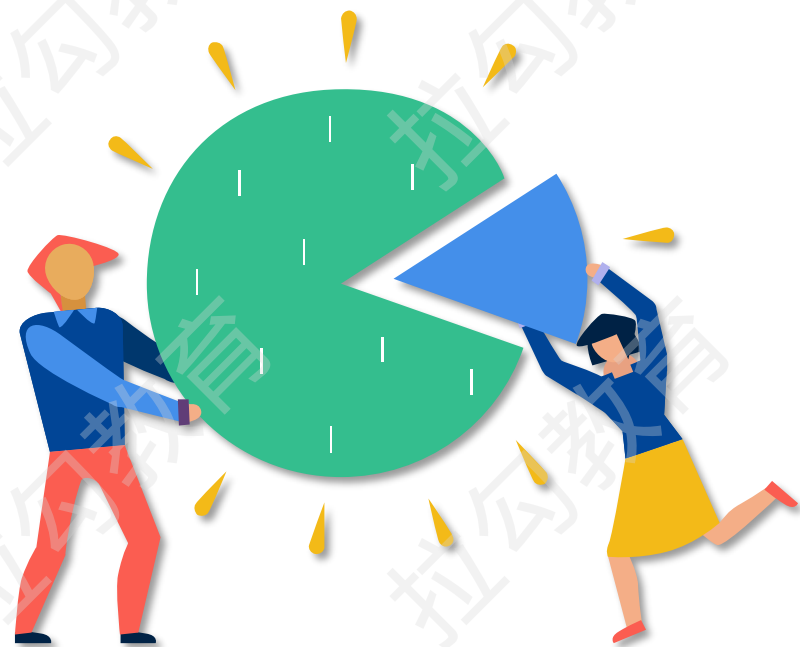
## 渲染性能优化手段中做减法

- 避免重排与重绘
- 防抖操作
- 减少 DOM 操作



## 渲染性能优化手段中做除法

- 骨架屏
- 使用 Web Worker
- React Fiber



前端性能优化实际上包括**量化**和**优化**

本课时提到了 5 个比较重要的指标

包括首屏绘制、首屏内容绘制、可交互时间、总阻塞时间、最大内容绘制





对不同用户产生的指标值进行统计

这里推荐使用百分位数统计法

对于不同性能需求的页面设置不同的百分位数



**优化**过程中要根据性能指标统计结果进行优化

可通过做减法和做除法的思路分别对加载性能和渲染性能进行优化



你还使用过哪些性能优化的指标 ?



Next：第24讲：《你的代码是怎么成为黑客工具的》

# 拉勾教育

— 互联网人实战大学 —



下载「拉勾教育App」  
获取更多内容