

目录	
第1章 优化的概念	
1 开篇词：你的前端性能还能再抢救一下	
2 解读雅虎35条军规（上）	
3 解读雅虎35条军规（下）	
4 你要不要看这些优化指标?	最近阅读
第2章 性能工具介绍	
5 性能优化百宝箱（上）	
6 性能优化百宝箱（下）	
第3章 网络部分	
7 聊聊 DNS Prefetch	
8 Webpack 性能优化两三事	
9 图片加载优化（上）	
10 图片加载优化（下）	
第4章 缓存部分	
11 十八般缓存	
12 CDN 缓存	
13 本地缓存（Web Storage）	
14 浏览器缓存（上）	
15 浏览器缓存（下）	
第5章 渲染部分	
16 渲染原理与性能优化	
17 如何应对首屏“一片空白”（上）	
18 如何应对首屏“一片空白”（下）	
19 不容小觑的 DOM 性能优化	

4 你要不要看这些优化指标？

更新时间：2019-11-26 09:48:59



“困难只能吓倒懦夫懒汉，而胜利永远属于敢于等科学高峰的人。”——茅以升

相信大家平时都会去阅读一些性能优化的文章。很多文章一开始就开始讲如何优化性能，却未能给出一个可计算，可采集的性能量化标准。还有一些文章，提到做了性能优化之后页面加载速度提升了多少多少，但是当你去问他你怎么得到这些量化指标的时候，却并不能给出行之有效的方法。

那么如何得到准确的性能优化数据呢？这就引出了我们今天要讲的性能优化指标，任何事情都有衡量的标准，当然前端性能优化也有衡量它对应的指标。可能有的同学对一些专业名词嗤之以鼻，认为记忆这些指标没什么用，的确有的指标确实用到的地方非常少，但是对于一些比较重要的指标还是有必要掌握的，本小节主要介绍一些常用的衡量指标，利用这些指标能够更好地对网站存在的性能问题进行优化，做到有的放矢。

Performance API

提到性能优化指标就不得不说 Performance API，这个 API 是 HTML5 新增的特性。过去我们要统计脚本的运行时间，会使用 Date.getTime() 去获取对应的时间；如果要获取白屏时间是在 head 末尾插入一段获取时间戳的代码，然后用这个时间戳减去开始接收数据的那个时间戳，得出的结果为白屏时间。这样的方法无疑是笨重的，而且获取的时间无法更精确(只能到ms级别)，而且一些后台比较关注的时间根本无法获取。W3C 为了解决这个问题，在 HTML5 推出的时候，新增了这个 API。

首先来看 MDN 上关于这个 API 的介绍：

Performance 接口可以获取到当前页面中与性能相关的信息。它是 High Resolution Time API 的一部分，同时也融合了 Performance Timeline API、Navigation Timing API、User

← 慕课专栏	≡ 你不知道的前端性能优化技巧 / 4 你要不要看这些优化指标?
目录	
第1章 优化的概念	
1 开篇词：你的前端性能还能再抢救一下	
2 解读雅虎35条军规（上）	
3 解读雅虎35条军规（下）	
4 你要不要看这些优化指标? 最近阅读	
第2章 性能工具介绍	
5 性能优化百宝箱（上）	
6 性能优化百宝箱（下）	
第3章 网络部分	
7 聊聊 DNS Prefetch	
8 Webpack 性能优化两三事	
9 图片加载优化（上）	
10 图片加载优化（下）	
第4章 缓存部分	
11 十八般缓存	
12 CDN 缓存	
13 本地缓存（Web Storage）	
14 浏览器缓存（上）	
15 浏览器缓存（下）	
第5章 渲染部分	
16 渲染原理与性能优化	
17 如何应对首屏“一片空白”（上）	
18 如何应对首屏“一片空白”（下）	
19 不容小觑的 DOM 性能优化	

上面提到我们可以通过window.performance来获取网站的各项指标信息，我们现在打开慕课网，然后在控制台输入window.performance，看看是否能够获取到相关信息，如下图：



可以看到我们获取到了相关信息，如图所以，performance包括了五个属性，其中**timing**是我们需要重点关注的，timing是一个map数据结构，其中key值是性能优化指标，value值是对应的时间戳。其中这些时间戳与页面整个加载过程中的关键节点是有着一一对应的关系，这里我们通过谷歌开发者网站的一张图来说明：



从图中可以看到很多指标都是成对出现，这里我们直接作差就可以求出对应页面加载过程中关键节点的时间，这里我们介绍几个比较常用的，比如：

```
const timingInfo = window.performance.timing;

// TCP连接耗时
timingInfo.connectEnd - timingInfo.connectStart

// DNS查询耗时
timingInfo.domainLookupEnd - timingInfo.domainLookupStart;

// 获得首字节耗费时间，也叫TTFB
timingInfo.responseStart - timingInfo.navigationStart

// domReady时间(与前面提到的DomContentLoaded事件对应)
timingInfo.domContentLoadedEventStart - timingInfo.navigationStart

// DOM资源下载
timingInfo.responseEnd - timingInfo.responseStart
```

上述几个就是比较常用的指标，这些指标大家也可以去Chrome浏览器的network面板中的Timing下获取，如图：

<div><div>← 慕课专栏</div><div>≡ 你不知道的前端性能优化技巧 / 4 你要不要看这些优化指标?</div></div>	
目录	
第1章 优化的概念	
1 开篇词：你的前端性能还能再抢救一下	
2 解读雅虎35条军规（上）	
3 解读雅虎35条军规（下）	
4 你要不要看这些优化指标? 最近阅读	
第2章 性能工具介绍	
5 性能优化百宝箱（上）	
6 性能优化百宝箱（下）	
第3章 网络部分	
7 聊聊 DNS Prefetch	
8 Webpack 性能优化两三事	
9 图片加载优化（上）	
10 图片加载优化（下）	
第4章 缓存部分	
11 十八般缓存	
12 CDN 缓存	
13 本地缓存（Web Storage）	
14 浏览器缓存（上）	
15 浏览器缓存（下）	
第5章 渲染部分	
16 渲染原理与性能优化	
17 如何应对首屏“一片空白”（上）	
18 如何应对首屏“一片空白”（下）	
19 不容小觑的 DOM 性能优化	



以上的瀑布图就是这些指标更为直观的展示。

感官性能优化指标

大家可能也注意到了我们上面介绍的一些指标都属于非视觉指标，是优化体验的常规指标。但是这些指标与用户关系并不大，因为用户根本无法感知到这几个时间，也无法得到用户的感官性能。

下面要介绍的就是与普通用户感官体验联系更加紧密的一些指标：

- First Paint(简称FP)：表示文档中任一元素首次渲染时间。
- First Contentful Paint(简称FCP)：当浏览器首次渲染任何文本，图像（包括背景图像），非白色画布或SVG时。这个指标就是我们日常说的白屏时间。
- First Meaningful Paint(简称FMP)：首次有意义的绘制，这个指标反映的是主要内容出现在页面上所需要的时间，如果FMP时间过长的话，这里就要考虑是不是静态文件阻塞了主线程。
- Time To Interactive(TTI):可交互时间，等到服务器通过HTTP协议将响应全部返回之后，便开始DOM Tree 的构建，完成之后，网页变成可交互状态，到此为止便是网页的可交互时间。用户可以进行正常的事件输入交互操作，这个指标是最重要的用户体验指标，用户最关心的就是什么时候可以进行交互，所以通常这个指标是我们优化的重点。

这里为了让大家更加直观的理解这4个指标，这里我给出一张图，相信看了这张图之后会对这些指标有更加直观的认识。



tips：上面介绍的4个指标，其中First Paint和First Contentful Paint是有对应的API来直接进行获取的，我们可以在控制台输入window.performance.getEntriesByType('paint')获取，另外2个指标我们可以通过性能优化工具Lighthouse实时测量进行获取，这部分下一节我们会进行介绍。

小结

<div><div>← 慕课专栏</div><div>你想知道的前端性能优化技巧 / 4 你要不要看这些优化指标？</div></div>		
解这些指标之后，我们才能更好的结合下一节我们要讲的性能优化工具进行相应的实践。		
目录		
第1章 优化的概念	← 3 解读雅虎35条军规（下）	5 性能优化百宝箱（上） →
1 开篇词：你的前端性能还能再抢救一下		
2 解读雅虎35条军规（上）	精选留言 6	
3 解读雅虎35条军规（下）	欢迎在这里发表留言，作者筛选后可公开显示	
4 你要不要看这些优化指标？ 最近阅读		
第2章 性能工具介绍		
5 性能优化百宝箱（上）		
6 性能优化百宝箱（下）		
第3章 网络部分		
7 聊聊 DNS Prefetch		
8 Webpack 性能优化两三事		
9 图片加载优化（上）		
10 图片加载优化（下）		
第4章 缓存部分		
11 十八般缓存		
12 CDN 缓存		
13 本地缓存（Web Storage）		
14 浏览器缓存（上）		
15 浏览器缓存（下）		
第5章 渲染部分		
16 渲染原理与性能优化		
17 如何应对首屏“一片空白”（上）		
18 如何应对首屏“一片空白”（下）		
19 不容小觑的 DOM 性能优化		

<div><div>← 慕课专栏</div><div>☰ 你不知道的前端性能优化技巧 / 4 你要不要看这些优化指标?</div></div>		
目录	<div><div>👍 0</div><div>回复</div></div>	2019-07-19
<div><div>第1章 优化的概念</div><div>1 开篇词：你的前端性能还能再抢救一下</div><div>2 解读雅虎35条军规（上）</div><div>3 解读雅虎35条军规（下）</div><div>4 你要不要看这些优化指标? <div>最近阅读</div></div><div>第2章 性能工具介绍</div><div>5 性能优化百宝箱（上）</div><div>6 性能优化百宝箱（下）</div><div>第3章 网络部分</div><div>7 聊聊 DNS Prefetch</div><div>8 Webpack 性能优化两三事</div><div>9 图片加载优化（上）</div><div>10 图片加载优化（下）</div><div>第4章 缓存部分</div><div>11 十八般缓存</div><div>12 CDN 缓存</div><div>13 本地缓存（Web Storage）</div><div>14 浏览器缓存（上）</div><div>15 浏览器缓存（下）</div><div>第5章 渲染部分</div><div>16 渲染原理与性能优化</div><div>17 如何应对首屏“一片空白”（上）</div><div>18 如何应对首屏“一片空白”（下）</div><div>19 不容小觑的 DOM 性能优化</div></div>		
<div><div>慕侠2389804</div><div>跟着老师一步步积累，一定能成为前端优化的高手</div><div><div>👍 0</div><div>回复</div></div></div>		
<div>千学不如一看，千看不如一练</div>		
<div>www.imoooc.com/read/41/article/642</div>		