



链滴

重绘和重排

作者: [cdq](#)

原文链接: <https://ld246.com/article/1646637297221>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

浏览器渲染过程

- 发生http请求，建立TCP连接。
- 处理 HTML 标记，并构造 DOM 树。
 - 在解析 HTML 时，遇到非阻塞资源，如图片资源，会请求这些资源并继续进行解析，css 文件是一样的，但是遇到了<script>标签的话，会阻塞渲染并且停止解析 HTML。
 - 浏览器构建 DOM 树时，这个过程占用了主线程，当这种情况发生是，预加载扫描器将解析可的内容并请求高优先级资源，如 CSS、JavaScript 和 Web 字体。有了预加载扫描器，我们就不用等解析器找到对外部资源的引用再来请求，它将在后台检索资源，以便在 HTML 解析器到达请求的资时，他们已经在运行或者已经下载完毕了。预加载扫描器减少了阻塞。
- 处理 CSS 并构建 CSSOM 树，CSSOM 和 DOM 树是相似的，但是他们是独立的数据结构，浏览将 CSS 规则转换为可以理解 and 使用的样式映射。浏览器根据遍历 CSS 的每个规则集，根据 CSS 选择创建具有父、子和兄弟关系的节点树。
- 将 CSSOM 和 DOM 树组合成一个 Render 树，计算样式树或渲染树从 DOM 树的根开始构建，历每个可见节点。
- 在渲染树上运行布局以计算每个节点的几何体。布局是确定呈现树中所有节点的高度、宽度和位置以确定页面上每个对象的大小和位置的过程。
- 最后一步是将各个节点绘制到屏幕上。
- 重绘
 - 概念：当一个元素的外观发生改变，但没有改变布局，浏览器将会将该元素重新绘制出来的过，叫做重绘。
 - 以下是会引发重绘的属性。

```
<table><thead><tr><th>属性: </th><th align="center">--</th><th align="right">--</th><th align="right">--</th></tr></thead><tbody><tr><td>color</td><td align="center">order-style</td><td align="right">visibility</td><td align="right">background</td></tr><tr><td>text-decoration</td><td align="center">background-image</td><td align="right">background-position</td><td align="right">background-repeat</td></tr><tr><td>outlin-color</td><td align="center">outline</td><td align="right">outline-style</td><td align="right">border-radius</td></tr><tr><td>outline-width</td><td align="center">box-shad w</td><td align="right">background-size</td><td align="right"></td></tr></tbody></table>
```

- 重排
 - 概念：当 DOM 的变化影响了元素的几何信息(元素的大小和位置)，浏览器需要重新计算元素的何属性，将其安放在正确的位置，这个过程叫重排，重排也叫回流。
 - 以下操作会引发重排：
 - 页面初始渲染，这是开销最大的一次重排
 - 添加/删除可见的DOM元素
 - 改变元素位置
 - 改变元素尺寸，比如边距、填充、边框、宽度和高度等
 - 改变元素内容，比如文字数量，图片大小等
 - 改变元素字体大小

- 改变浏览器窗口尺寸，比如resize事件发生时
 - 激活CSS伪类（例如：`:hover`）
 - 设置 style 属性的值，因为通过设置style属性改变结点样式的话，每一次设置都会触发一次re low
 - 查询某些属性或调用某些计算方法：`offsetWidth`、`offsetHeight`等，除此之外，当我们调用`etComputedStyle`方法，或者IE里的`currentStyle`时，也会触发重排，原理是一样的，都为求一个“时性”和“准确性”。

常见引起重排属性和方法

<code>width</code>	<code>height</code>	<code>margin</code>	<code>padding</code>
<code>display</code>	<code>border-width</code>	<code>border</code>	
<code>overflow</code>	<code>font-size</code>	<code>vertical-align</code>	
<code>clientWidth</code>	<code>clientHeight</code>	<code>clientTop</code>	
<code>offsetWidth</code>	<code>offsetHeight</code>	<code>offsetTop</code>	
<code>scrollWidth</code>	<code>scrollHeight</code>	<code>scrollTop</code>	
<code>scrollIntoView()</code>	<code>scrollTo()</code>	<code>getComputedStyle()</code>	
<code>getBoundingClientRect()</code>	<code>scrollIntoViewIfNeeded()</code>		

- 重排影响的范围又分为全局范围和局部范围
 - 全局范围：从根节点开始对整个渲染树进行重新布局。
 - 局部范围：对渲染树的某部分或者说是某一个渲染对象进行重排。
- 重排一定引起重绘，重绘不一定引起重排。