

html文档页面重绘和重新布局

当浏览器下载完所有页面HTML 标记，JavaScript，CSS，图片之后，它解析文件并创建两个内部数据结构：一棵DOM树表示页面结构，一棵渲染树表示DOM节点如何显示。在对html页面的操作过程中会改变页面的结构或者页面的外观或者同时改变页面的结构和外观。在过程中,文档可见外观改变很小(如：改变了某些元素的颜色,或者改变了某些元素的可见性,但这种改变不会影响到页面的布局),这种行为称为“重绘”,也就是浏览器对文档进行了重新绘制。如果页面外观发生了显著变化并且影响到文档的实际布局(如一个<div>元素的大小发生了改变),这时浏览器将会进行重新布局，也就是浏览器将需要重新计算并重新绘制改div元素的所有子元素。

重绘和重新布局都是非常耗性能的一种操作。发生重绘时不一定会发生重新布局,但发生重新布局一定会发生重绘。那么哪些页面的操作会发生重绘与重新布局呢。

正如前面所提到的，当布局和几何改变时需要重新布局。在下述情况中会发生重新布局：

- (1)添加或删除可见的DOM 元素
- (2)元素位置改变
- (3)元素尺寸改变（因为边距，填充，边框宽度，宽度，高度等属性改变）
- (4)内容改变，例如，文本改变或图片被另一个不同尺寸的所替代最初的页面渲染
- (5)浏览器窗口改变尺寸

在使用javascript对页面的操作时会发生重绘或者重新布局,虽然不能避免这2种行为,但可以减少触发这两种行为的发生。如：

```
div.style.width='200px',  
div.style.height='300px',  
div.style.color='#FFF',  
div.style.fontSize='16px'
```

这样浏览器会对页面进行4次计算与绘制，

我们可以把这些改变放到一个类里面,如：

```
.change{width:200px;height:300px;color:#FFF;font-size:16px;}
```

然后将div的类设置为change

```
div.class='change'
```

这样浏览器对页面的计算和绘制就减少到一次了。

有时会给页面添加多个节点时，有可能触发多次重新布局文档的操作。如：

```
var i=0;  
for(;i<10;i++){  
    var newP=document.createElement('p'),  
        newText='new p elements ---'+i;  
    newP.innerText=newText;
```

```
document.body.appendChild(newP);  
}
```

这会触发10重新布局。这时使用文档片段就会将重新布局减少到一次。如：

```
var i=0 , pFragment=document.createDocumentFragment();  
for(;i<10;i++){  
    var newP=document.createElement('p'),  
        newText='new p elements ---'+i;  
    newP.innerText=newText;  
    pFragment.appendChild(newP);  
}  
document.body.appendChild(pFragment);  
javascript对文档性能的影响主要都是在对DOM的操作上。
```