

## 小tips: 滚动容器尺寸变化子元素视觉上位置不变JS实现

这篇文章发布于 2018年02月2日, 星期五, 01:08, 归类于 [JS实例](#)。阅读 12726 次, 今日 6 次 [6 条评论](#)

by zhangxinxu from <http://www.zhangxinxu.com/wordpress/?p=7351>

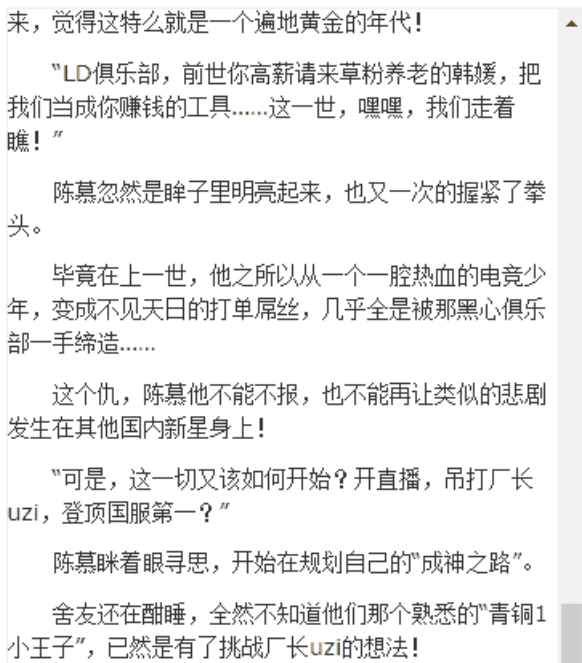
本文可全文转载, 但需要保留原作者和出处。

### 一、先从需求说起

对于一个宽度不固定的滚动容器, 如果里面内容已经滚动到了一定的高度, 这个时候滚动容器的宽度发生变化, 则里面内容的位置会进行重定位, 一不留神就不知道刚才的位置是哪里了。

尤其是看小说这种非常考验眼力的场景。

例如下面的GIF截图演示:



于是, 就有需求。

当滚动容器尺寸发生变化的时候 (如宽屏窄屏切换, 或者默认尺寸变全屏时候), 最上面元素位置要保持不变, 这样视觉体验就很好。不会因为突然的尺寸变化而不知道刚才看到哪里了。

那么该如何实现呢?

需要借助JavaScript手动修正滚动位置。

### 二、滚动容器尺寸变化子元素视觉上位置不变的JS处理

我们先看实现后的截屏gif效果:

一直没上锁的宿舍门，忽然被人推开。

“兄弟们，是时候走起了！”

有个浑厚的噪音响起.....不用看陈慕都知道，门口肯定是隔壁宿舍的大壮，一个同样痴迷英雄联盟的家伙。

“我擦，还没养足精神啊！”

“老哥你这嗓门太大了！”

“起床！”

这一个噪音响起之后，随着一个高大个堵住了门口，从冬子到超胜再到黄站长，107寝室三人全部醒来。

“咦，小慕你早醒来了？”

睡眼惺忪的冬子，一眼就看到了坐在床上面带微笑的陈慕。

“奇怪，你不是早上才睡的么？这么快就醒了。”

黄站长也斜了下脑袋，表示郁闷，毕竟陈慕

您可以狠狠地点击[这里](#)：[滚动容器尺寸变化时候最上方元素位置不变demo](#)

滚动容器到合适位置，然后改变浏览器窗口尺寸，可以体验到微丝不动的非常友好的交互体验效果。

## JS实现的原理

1. 获得最靠近滚动容器上边缘的元素；
2. 获得最靠近滚动容器上边缘的元素距离上边缘的距离；
3. 当滚动容器尺寸改变后，获得之前最靠近上边缘元素现在距离上边缘的距离，根据前后的差值修正此时的scrollTop大小；

原理第一步是最难点，如何获得最靠近滚动容器上边缘的元素呢？

我这里使用的是 `document.elementsFromPoint` 方法，语法如下：

```
var elements = document.elementsFromPoint(x, y);
```

表示返回所有距离浏览器可视窗口 `x, y` 坐标的DOM元素集合。因此，`elements` 是一个从最子元素开始，依次向上，一直到 `<body>`，`<html>` 元素的类数组集合。

在本例中，`elements[0]` 就是我们需要的元素。

假设滚动容器元素的 `id` 是 `box`，则使用JavaScript代码表示就是：

```
var box = document.getElementById('box');

// x就取滚动容器水平中心点
var x = box.getBoundingClientRect().left + box.clientWidth / 2;
// + 12为了让更靠下一点的元素作为边缘元素
var y = box.getBoundingClientRect().top + 12;
// target就是我们获取的最接近滚动容器上边缘的元素
var target = document.elementsFromPoint(x, y)[0];
```

然后 `target` 距离上边缘的距离也很好实现（容器 `scroll` 事件时候执行）：

```
var offsetTop = target.getBoundingClientRect().top - box.getBoundingClientRect().top
```

最后，根据滚动后的边缘距离进行滚动修正（窗体 `resize` 事件时候执行）：

```
var currentOffsetTop = box.getBoundingClientRect().top - target.getBoundingClientRect().top;
// 滚动修正，效果达成
box.scrollTop = box.scrollTop - currentOffsetTop - offsetTop;
```

更完整JS代码参见[demo页面](#)（就几十行），大家可以根据自己实际项目场景进行修改，这里不重复展示。

### 关于document.elementsFromPoint API的兼容性

根据MDN上的数据，`document.elementsFromPoint` IE10+才支持，而且需要 `ms` 私有前缀，因此，如果想要兼容IE浏览器，可以加一句：

```
if (!document.elementsFromPoint) {
    document.elementsFromPoint = document.msElementsFromPoint;
}
```

由于本文实例属于体验增强的功能，因此，就算浏览器不支持也无伤大雅，能够让90+%浏览器有更好体验已经很棒了，因此是可以放心大胆使用的一个技术案例。

### IE浏览器下更多细节

在IE浏览器下测试，`resize` 时候滚动内容有晃动，我猜测可能与边缘位置计算带小数有关，因此，`offsetTop` 的计算都是取整的，如下示意：

```
var currentOffsetTop = Math.round(box.getBoundingClientRect().top) - Math.round(target.getBoundingClientRect().top);
```

体验果然好了很多。

另外，在Chrome等浏览器下，`target` 的获取和偏移计算可以直接在 `window resize` 时候执行。但是IE浏览器不行，似乎IE浏览器先触发浏览器行为，再执行了 `resize` 事件，所以，综合下来，还是建议最上边缘元素获取放在 `scroll` 事件中处理。

## 三、结束语

一个JS体验优化实例，没什么其他好展开的。

感谢阅读！



《CSS世界》签名版独家发售，包邮，可指定寄语，点击显示购买码

（本篇完） // 想要打赏？点击[这里](#)。有话要说？点击[这里](#)。



### 猜你喜欢

- CSSOM视图模式(CSSOM View Module)相关整理
- 小tip: 子元素scroll父元素容器不跟随滚动JS实现
- 有意思: textarea resize属性下纯CSS交互效果
- 利用重绘解决IE下JS交互产生的定位重叠等棘手bug
- 翻译 - 逐渐消失的Flash网站
- jQuery-马化腾产品设计与用户体验的一些技术实现
- 翻译: 慢点更好-为何排序比速度更重要?
- 5分钟快速了解下CSS4 color-adjust属性
- jQuery图片文本滚动切换插件jCarousel中文翻译与详解
- 纯CSS实现易拉罐3D滚动效果
- 基于HTML模板和JSON数据的JavaScript交互

分享到:         { 0 }

标签: elementsFromPoint, getBoundingClientRect, resize, scrollTop, 交互, 体验, 滚动

### 发表评论（目前6条评论）

名称 (必须)

邮件地址(不会被公开) (必须)

网站

提交评论

1. 前辈杠杠滴说道:  
2018年05月29日 14:53

刚好在找，刚好找到，刚刚好

像前辈这样的男人，注定是不会单身的对不。

[回复](#)



2. half肥宅说道:  
2018年05月25日 16:58

这个功能真的很实用很人性化啊！超赞的！

[回复](#)



3. 仰慕前辈说道:

2018年05月3日 20:39

您这样不断的探索新的东西，你把前端真心才发挥到极致。佩服佩服前辈的精神

[回复](#)



4. 鎏金圣手火麒麟说道:

2018年02月5日 10:59

感觉自己的思维真的不能被项目的兼容性所限制，还是应该尽可能多地将新技术用到项目中，起码可以优化高级浏览器的体验效果

[回复](#)



5. mengkun说道:

2018年02月2日 21:16

强烈建议一些响应式前端框架的 wiki 文档加入这个！每次对照响应式效果时一调整浏览器大小就不知道滚动到哪去了~哈哈~

[回复](#)

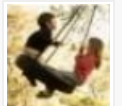


6. webster说道:

2018年02月2日 09:29

前端发挥到了极致啊，羡慕 佩服

[回复](#)



#### 最新文章

- » [常见的CSS图形绘制合集](#)
- » [粉丝群第1期CSS小测点评与答疑](#)
- » [分享三个纯CSS实现26个英文字母的案例](#)
- » [小tips: 纯CSS实现打字动画效果](#)
- » [CSS/CSS3 box-decoration-break属性简介](#)
- » [CSS :placeholder-shown伪类实现Material Design占位符交互效果](#)
- » [从天猫某活动视频不必要的3次请求说起](#)
- » [CSS vector-effect与SVG stroke描边缩放](#)
- » [CSS ::backdrop伪元素是干嘛用的?](#)
- » [周知: CSS -webkit-伪元素选择器不再导致整行无效](#)

#### 今日热门

- » [常见的CSS图形绘制合集](#) <sup>(178)</sup>
- » [粉丝群第1期CSS小测点评与答疑](#) <sup>(112)</sup>
- » [未来必热: SVG Sprite技术介绍](#) <sup>(111)</sup>
- » [HTML5终极备忘大全 \(图片版+文字版\)](#) <sup>(85)</sup>
- » [让所有浏览器支持HTML5 video视频标签](#) <sup>(83)</sup>
- » [Selectivizr-让IE6~8支持CSS3伪类和属性选择器](#) <sup>(80)</sup>
- » [CSS3下的147个颜色名称及对应颜色值](#) <sup>(78)</sup>
- » [小tips: 纯CSS实现打字动画效果](#) <sup>(72)</sup>
- » [写给自己看的display: flex布局教程](#) <sup>(69)</sup>
- » [分享三个纯CSS实现26个英文字母的案例](#) <sup>(69)</sup>

#### 今年热议

- » [《CSS世界》女主角诚寻靠谱一起奋斗之人](#) <sup>(76)</sup>
- » [不借助Echarts等图形框架原生JS快速实现折线图效果](#) <sup>(64)</sup>
- » [看, for..in和for..of在那里吵架!](#) <sup>(60)</sup>
- » [是时候好好安利下LuLu UI框架了!](#) <sup>(47)</sup>
- » [原来浏览器原生支持JS Base64编码解码](#) <sup>(35)</sup>
- » [妙法攻略: 渐变虚框及边框滚动动画的纯CSS实现](#) <sup>(33)</sup>
- » [炫酷H5中序列图片视频化播放的高性能实现](#) <sup>(31)</sup>

- » [CSS scroll-behavior和JS scrollIntoView让页面滚动平滑](#) <sup>(30)</sup>
- » [windows系统下批量删除OS X系统.DS\\_Store文件](#) <sup>(26)</sup>
- » [写给自己看的display: flex布局教程](#) <sup>(26)</sup>

## 猜你喜欢

- [CSSOM视图模式\(CSSOM View Module\)相关整理](#)
- [小tip: 子元素scroll父元素容器不跟随滚动JS实现](#)
- [有意思: textarea resize属性下纯CSS交互效果](#)
- [利用重绘解决IE下JS交互产生的定位重叠等棘手bug](#)
- [翻译 - 逐渐消失的Flash网站](#)
- [jQuery-马化腾产品设计与用户体验的一些技术实现](#)
- [翻译: 慢点更好-为何排序比速度更重要?](#)
- [5分钟快速了解下CSS4 color-adjust属性](#)
- [jQuery图片文本滚动切换插件jCarousel中文翻译与详解](#)
- [纯CSS实现易拉罐3D滚动效果](#)
- [基于HTML模板和JSON数据的JavaScript交互](#)