



分类：[IT技术](#)

上一篇：[小企业的生存之道](#)

下一篇：[对于Rails Rumble 200](#)

## 用 Javascript 获取页面元素的位置

作者：阮一峰

日期：2009年9月14日

制作网页的过程中，你有时候需要知道某个元素在网页上的确切位置。

下面的教程总结了Javascript在网页定位方面的相关知识。

### 一、网页的大小和浏览器窗口的大小

首先，要明确两个基本概念。

一张网页的全部面积，就是它的大小。通常情况下，网页的大小由内容和CSS样式表决

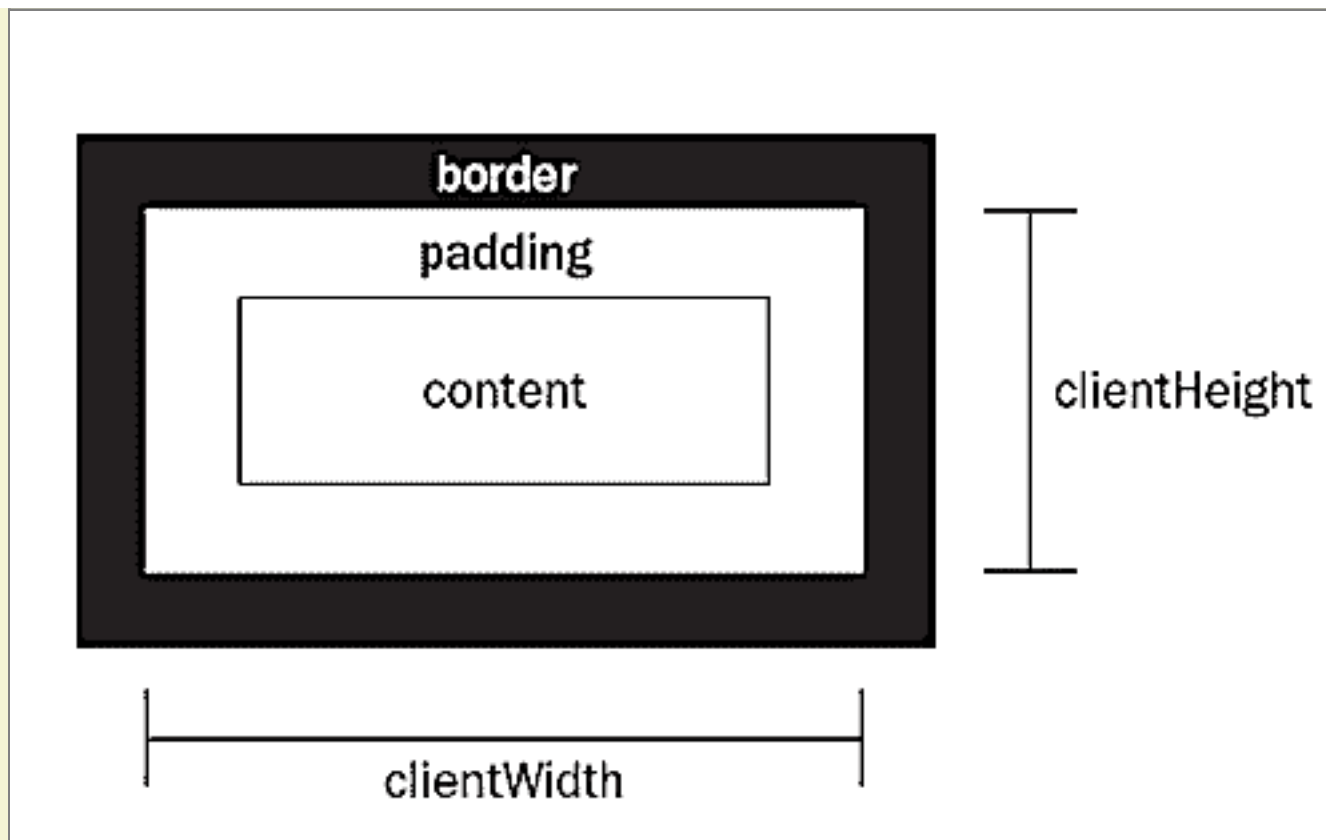
定。

浏览器窗口的大小，则是指在浏览器窗口中看到的那部分网页面积，又叫做viewport（视口）。

很显然，如果网页的内容能够在浏览器窗口中全部显示（也就是不出现滚动条），那么网页的大小和浏览器窗口的大小是相等的。如果不能全部显示，则滚动浏览器窗口，可以显示出网页的各个部分。

## 二、获取网页的大小

网页上的每个元素，都有clientHeight和clientWidth属性。这两个属性指元素的内容部分再加上padding的所占据的视觉面积，不包括border和滚动条占用的空间。



（图一 clientHeight和clientWidth属性）

因此，document元素的clientHeight和clientWidth属性，就代表了网页的大小。

```
function getViewport(){  
    if (document.compatMode == "BackCompat"){  
        return {  
            width: document.body.clientWidth,  
            height: document.body.clientHeight
```

```
    }  
  } else {  
    return {  
      width: document.documentElement.clientWidth,  
      height: document.documentElement.clientHeight  
    }  
  }  
}
```

上面的getViewport函数就可以返回浏览器窗口的高和宽。使用的时候，有三个地方需要注意：

- 1) 这个函数必须在页面加载完成后才能运行，否则document对象还没生成，浏览器会报错。
- 2) 大多数情况下，都是document.documentElement.clientWidth返回正确值。但是，在IE6的quirks模式中，document.body.clientWidth返回正确的值，因此函数中加入了对文档模式的判断。
- 3) clientWidth和clientHeight都是只读属性，不能对它们赋值。

### 三、获取网页大小的另一种方法

网页上的每个元素还有scrollHeight和scrollWidth属性，指包含滚动条在内的该元素的视觉面积。

那么，document对象的scrollHeight和scrollWidth属性就是网页的大小，意思就是滚动条滚过的所有长度和宽度。

仿照getViewport()函数，可以写出getPagearea()函数。

```
function getPagearea(){
    if (document.compatMode == "BackCompat"){
        return {
            width: document.body.scrollWidth,
            height: document.body.scrollHeight
        }
    } else {
        return {
            width: document.documentElement.scrollWidth,
            height: document.documentElement.scrollHeight
        }
    }
}
```

```
}
```

但是，这个函数有一个问题。如果网页内容能够在浏览器窗口中全部显示，不出现滚动条，那么网页的`clientWidth`和`scrollWidth`应该相等。但是实际上，不同浏览器有不同的处理，这两个值未必相等。所以，我们需要取它们之中较大的那个值，因此要对`getPagearea()`函数进行改写。

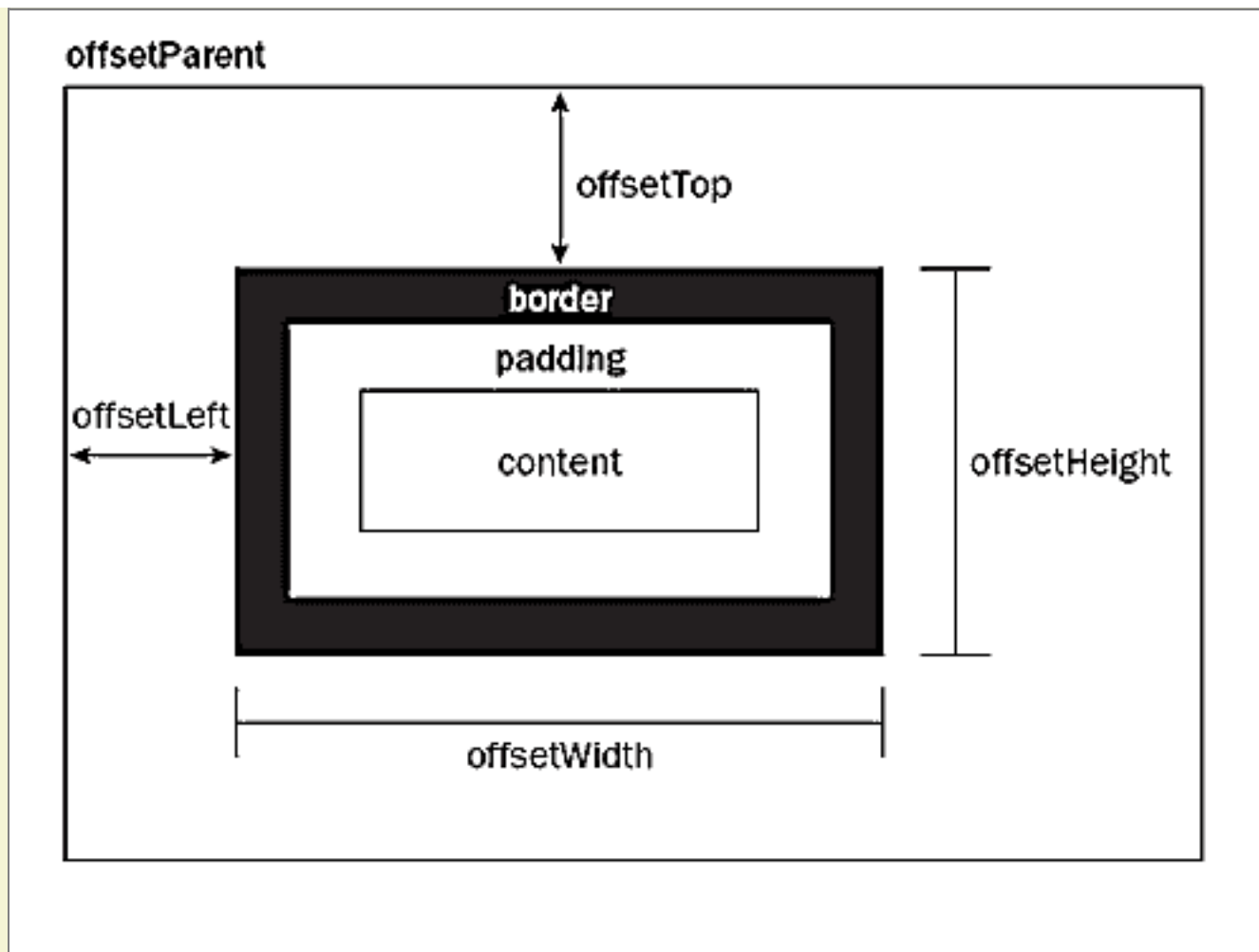
```
function getPagearea(){
    if (document.compatMode == "BackCompat"){
        return {
            width: Math.max(document.body.scrollWidth,
                           document.body.clientWidth),
            height: Math.max(document.body.scrollHeight,
                             document.body.clientHeight)
        }
    } else {
        return {
            width: Math.max(document.documentElement.scrollWidth,
                           document.documentElement.clientWidth),
            height: Math.max(document.documentElement.scrollHeight,
                             document.documentElement.clientHeight)
        }
    }
}
```

```
    }  
  }  
}
```

#### 四、获取网页元素的绝对位置

网页元素的绝对位置，指该元素的左上角相对于整张网页左上角的坐标。这个绝对位置要通过计算才能得到。

首先，每个元素都有offsetTop和offsetLeft属性，表示该元素的左上角与父容器（offsetParent对象）左上角的距离。所以，只需要将这两个值进行累加，就可以得到该元素的绝对坐标。



(图二 offsetTop和offsetLeft属性)

下面两个函数可以用来获取绝对位置的横坐标和纵坐标。

```
function getElementLeft(element){  
    var actualLeft = element.offsetLeft;  
    var current = element.offsetParent;
```



```
while (current !== null){
    actualLeft += current.offsetLeft;
    current = current.offsetParent;
}

return actualLeft;
}

function getElementTop(element){
    var actualTop = element.offsetTop;
    var current = element.offsetParent;

    while (current !== null){
        actualTop += current.offsetTop;
        current = current.offsetParent;
    }

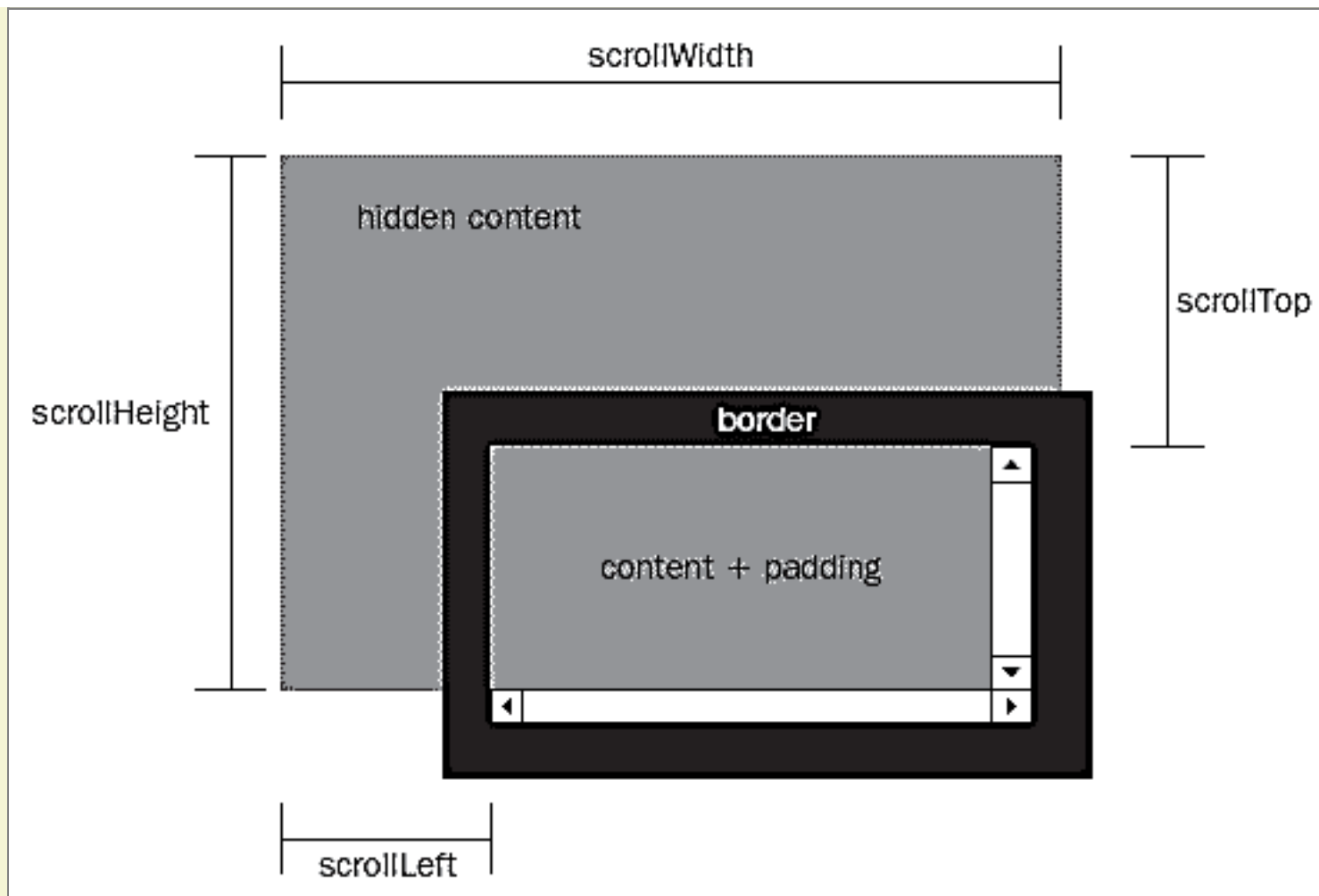
    return actualTop;
}
```

由于在表格和iframe中，offsetParent对象未必等于父容器，所以上面的函数对于表格和iframe中的元素不适用。

## 五、获取网页元素的相对位置

网页元素的相对位置，指该元素左上角相对于浏览器窗口左上角的坐标。

有了绝对位置以后，获得相对位置就很容易了，只要将绝对坐标减去页面的滚动条滚动的距离就可以了。滚动条滚动的垂直距离，是document对象的scrollTop属性；滚动条滚动的水平距离是document对象的scrollLeft属性。



(图三 scrollTop和scrollLeft属性)

对上一节中的两个函数进行相应的改写：

```
function getElementViewLeft(element){  
    var actualLeft = element.offsetLeft;  
    var current = element.offsetParent;
```

```
while (current !== null){
    actualLeft += current.offsetLeft;
    current = current.offsetParent;
}

if (document.compatMode == "BackCompat"){
    var elementScrollLeft=document.body.scrollLeft;
} else {
    var elementScrollLeft=document.documentElement.scrollLeft;
}

return actualLeft-elementScrollLeft;
}

function getElementViewTop(element){
    var actualTop = element.offsetTop;
    var current = element.offsetParent;

    while (current !== null){
        actualTop += current. offsetTop;
        current = current.offsetParent;
    }
}
```

```
    }

    if (document.compatMode == "BackCompat"){
        var elementScrollTop=document.body.scrollTop;
    } else {
        var elementScrollTop=document.documentElement.scrollTop;
    }

    return actualTop-elementScrollTop;
}
```

scrollTop和scrollLeft属性是可以赋值的，并且会立即自动滚动网页到相应位置，因此可以利用它们改变网页元素的相对位置。另外，element.scrollIntoView()方法也有类似作用，可以使网页元素出现在浏览器窗口的左上角。

## 六、获取元素位置的快速方法

除了上面的函数以外，还有一种快速方法，可以立刻获得网页元素的位置。

那就是使用getBoundingClientRect()方法。它返回一个对象，其中包含了left、right、top、bottom四个属性，分别对应了该元素的左上角和右下角相对于浏览器窗口（viewport）左上角的距离。

所以，网页元素的相对位置就是

```
var X= this.getBoundingClientRect().left;
```

```
var Y =this.getBoundingClientRect().top;
```

再加上滚动距离，就可以得到绝对位置

```
var X= this.getBoundingClientRect().left+document.documentElement.scrollLeft;
```

```
var Y =this.getBoundingClientRect().top+document.documentElement.scrollTop;
```

目前，IE、Firefox 3.0+、Opera 9.5+都支持该方法，而Firefox 2.x、Safari、Chrome、Konqueror不支持。

(完)

## 文档信息

---

- 版权声明：自由转载-非商用-非衍生-保持署名 | Creative Commons BY-NC-ND 3.0
- 原文网

■ 最后修改时间：2010年5月27日 19:14



## 相关文章

■ **2010.12.27:** [PHP最佳实践](#)

今天下午，我在读下面这篇文章。

■ **2010.12.07:** [《黑客英雄》书摘](#)

《黑客与画家》翻译完成后，我还欠出版社一篇译者序。

## 功能链接

- 前一篇：[小企业的生存之道](#)
- 后一篇：[对于Rails Rumble 2009的一点感想](#)
- 更多内容请访问：[首页](#) » [档案](#) » [IT技术](#)
- 站内搜索：
- **Feed**订阅：

## 广告

---

留言（**6**条）



**dylanklc** 说：

JS 不用prototype 手写还是很烦的.....

2009年9月14日 17:43 | 档案 | 引用

---

**joiho** 说：

阮兄真是什么都学呀

不少人都很期望你学习经历、感悟相关的文章，好像也没正面说过会不会写

2009年9月14日 20:07 | 档案 | 引用

---

**3班袁春阳** 说：

最后一种方法比较好用,只有快感,但没有成就感

2009年9月15日 09:30 | 档案 | 引用

---

老犁 说：

```
function getPagearea(){  
    if (document.compatMode == "BackCompat"){  
        return {  
            width: document.body.scrollWidth,
```

```
        height: document.body.scrollHeight
    }
} else {
    return {
        width: document.documentElement.scrollWidth,
        height: document.documentElement.scrollHeight
    }
}
}
```

-----

这代码写得有点啰嗦，可以改成这样。

```
function my_function() {
    if ( xxxx ) {
        return xxxx;
    }
    return xxxx;
}
```

或者这样：

```
function my_function() {
    var ret_value = null;
```

```
if ( xxxx ) {  
    ret_value = xxxx;  
}  
return ret_value;  
}
```

2009年9月15日 09:32 | 档案 | 引用

---

**tzwlss** 说：

文字代码都很普通，图片很赞啊，对应理解这些一看就知道了，赞！

2009年9月21日 20:38 | 档案 | 引用

---

**harley** 说：

受益匪浅

2010年2月25日 09:43 | 档案 | 引用

---

## 我要发表看法

您的留言（HTML标签部分可用）

您的大名：

«- 必填

电子邮件：

«- 必填，不公开

个人网址：

«- 我信任你，不会填写广告链接

记住个人信息？☐

发表

«- 点击按钮

