

HTML <area><map>标签及在实际开发中的应用

这篇文章发布于 2017年05月6日, 星期六, 01:32, 归类于 [HTML](#) 相关。 阅读 37321 次, 今日 19 次 16 条评论

by zhangxinxu from <http://www.zhangxinxu.com/wordpress/?p=6115>

本文可全文转载, 但需得到原作者书面许可, 同时保留原作者和出处, 摘要引流则随意。

之前, 我一直以为HTML `<area>` 是一个鸡肋HTML, 估计到了HTML5时代会被废弃的命。但是, 最近一查资料, 乖乖了个咚, 不仅没被废弃, 反而发展了, 新增了一些标签属性, 例如 `rel`, `media`, `hreflang` 等。

然后进一步研究发现, 尼玛这货居然还可以用来解决一个困扰我很多年的棘手问题, 形象立马高大上了起来。

一、HTML <area>原本作用

HTML `<area>` 原本作用是给 `` 图像标记热点区域用的。例如一张中国省份地图, 点击对应省份显示对应省份介绍的页面; 又或者旭旭影院或者旭旭餐厅的座位平面图, 点击对应座位选座, 都是可以使用 `<area>` 标签的。

言语苍白, 实例说话, 您可以狠狠地点击这里: [HTML <area>基本使用demo](#)

HTML代码如下:

```

<map id="MM" name="MM">
  <area shape="rect" coords="20,20,80,80" href="#rect" alt="矩形">
  <area shape="circle" coords="200,50,50" href="#circle" alt="圆形">
  <area shape="poly" coords="150,100,200,120,180,130,190,180,150,150,100,160,140,120,100,110" href="#poly" alt="多边形">
</map>
```

此时图片上就有3个热区, 我们使用Tab键索引高亮就可以窥见其轮廓范围, 如下图片3连击:



图片和热点区域元素关联是使用图片的 `usemap` 属性, 其值对应 `<map>` 的 `id` 或者 `name` 值 (Chrome 浏览器只支持 `name` 属性值关联)。

这里出现了两个HTML标签, 一个是 `<area>` 还有一个是 `<map>`, 这些都是从很早就支持的HTML标签, 所以不必担心兼容性问题。其中就闭合特性来看, `<area>` 类似 ``, 是无法有子元素或其他子内容的。

这里出现了几个属性:

shape

`shape` 表示点击热点区域的形状, 支持矩形 `rect`, 圆形 `circle` 以及多边形 `poly`。

coords

`coords` 表示点击热点区域形状的坐标。坐标点 `0,0` 表示图片的左上角。其中矩形 `rect` 支持 4 个数值, 2 个坐标, 分别是矩形左上角坐标和矩形右下角坐标。例如, `coords="20,20,80,`

80" 生成的就是一个左上角坐标 20,20 宽高都是 60px 的矩形区域。圆形 `circle` 支持 3 个数值, 前 2 个值为圆心坐标, 第 3 个值为圆的半径大小。多边形 `poly` 就更简单了, 每两个数值组合表示一个坐标点, 依次连线形成的区域就是最终的热点区域。

href

`href` 和 `<a>` 元素的 `href` 是一样的东西, 直接跳转地址, 或者锚点等。也同样支持 `target` 属性和 `rel` 属性。也就是说 `<area>` 可以看成是半个 `<a>` 元素。

alt

`alt` 同 `` 元素的 `alt`, 表示热点区域图片的描述信息。

如果 `<area>` 要想和图片热点关联, 祖先元素 `<map>` 是不可缺少的, 但是不一定非得直接父子关系。

`<map>` 和 `<area>` 之间再嵌套个 `<div>` 什么的功能也是正常的。

如果单从 `<area>` 原本作用来讲, 这应该是一个非常有实用价值的属性, 例如, 社交站点的图片往往都会有人类识别标注功能, 类似这样:



跟 `<area>` 的热点看上去就是一个东西, 然而, 实际上, 我们很少很少使用 `<area>` 元素, 为什么呢?

因为图片上热点区域的样式根本无法自定义, 除了可以显示 `outline` 轮廓意外, 什么边框背景色都不行, 对 `area` 进行CSS设置, 也仅仅是改变这个标签元素本身的样式, 而不是图片上热点点击区域的样式。于是, 最终, 图片上的标注或者热点, 我们常常是使用其他标签, 如 `<div>`, 或者 `<a>` 标签进行覆盖进行模拟。

也就是说, HTML `<area>` 只能作用于对点击热点没有任何样式要求的场景。考虑到这种场景使用 `<a>` 标签模拟也是可以的, 进一步压缩了HTML `<area>` 的实用价值。除非是不规则的点击热点, 因为这种不规则区域包括IE8这种不支持CSS3的浏览器也是支持的, 如下图:



所以, 这么多年下来, 从没使用过 `<area>`, 也没有什么异样, 花儿还是那么香, 雾霾还是那么浓。难道, HTML `<area>` 真的一无是处, 只能在某些罕见场景下才有存在价值吗?

如果真是这样, 我也不会写这篇文章了, 实际上, 偶然间, 我发现了关于 `<area>` 一个隐匿的特性, 这

个特性可能会让 `<area>` 远离雾霾，闻到花香。

二、HTML `<area>` 衍生的更有价值的作用

正如前文提到的，`<area>` 元素可以看出半个 `<a>` 元素，这句话很关键，换句话说，有时候，我们可以使用 `<area>` 元素代替 `<a>` 元素。

于是，我们的思路就变成：`<a>` 元素有什么局限性，导致有时候我们需要使用 `<area>` 代替 `<a>` 元素呢？

答案就是：`<a>` 不支持嵌套。例如：

```
<a href="#1">1111111111<a href="#2">2222222222</a></a>
```

浏览器会将其解析为相邻兄弟关系，而不是父子关系，如下截图示意：



所以，如果我们有链接嵌套的需求，就可以试试使用 `<area>` 元素，举个例子，如下图标注：

上图黄色高亮区域点击是去“书籍详细页”，而点击列表其它任意区域是去“书籍阅读页”，请问该如何实现？

传统方法，比较好是改变视觉DOM结构和顺序，使链接内容变成相邻关系，再通过CSS重新布局定位，虽然麻烦了点，但至少语义和无障碍访问这一块还是很好的；比较次的方法是使用JS，点击目标区域 `preventDefault` 阻止默认事件，然后再 `location.href` 跳转之类。

实际上，还有一种更好的做法，就是 `<a>` 元素中嵌套 `<area>` 元素，可以保证DOM结构符合视觉呈现，又无需JS辅助。

眼见为实，您可以狠狠地点击这里：[借助area元素实现的链接嵌套demo](#)

demo截图如下：



在Chrome和IE9+浏览器下，点击图片以及标题会跳转到“书籍详情页”，而点击其他区域就会跳到“书籍阅读页”，链接嵌套功能就这么实现了。

其中，demo中，图片的链接嵌套和标题文字的实现原理有所不同。

图片的链接嵌套用法更加规范，也就是借助 `<map>` 和 `<area>` 元素在图片上创造点击热点，相关HTML代码如下：

```

<map id="bookCover" name="bookCover">
  <area shape="rect" coords="0,0,54,72" href="book/1003477570" alt="都市猎人 限时免费" target="_blank">
</map>
```

也就是把整个图片区域都作为点击热点，充分发挥了 `<area>` 标签本身的特性，没有任何兼容性问题，键盘访问等都非常良好，又满足我们链接嵌套功能，`<area>` 元素的实用价值大大地得以体现。

对于标题文字“都市猎人”的链接内嵌处理则要麻烦一些。demo实例页面的做法是，在文字前面直接放置了一个裸的 `<area>` 元素，通过CSS设置覆盖文字实现的，相关HTML和CSS代码如下：

```
<h4 class="book-title">
  <area class="area" href="/book/1003477570" target="_blank">
    都市猎人
</h4>
```

```
.book-title {
  position: relative;
}
.book-title > .area {
  position: absolute;
  left: 0; top: 0;
  width: 100%; height: 100%;
}
```

脱离 `<map>` 裸露的 `<area>` 元素实际上也是可以点击跳转了，就像一个 `<a>` 链接元素一样，但是 `<area>` 元素不支持子元素，所以只能覆盖。但是，却支持伪元素，也就是，如果单看视觉表现，你也可以让标题文字借助 `:before` 等伪元素实现。

然而，虽然主观功能都OK了，代码也很精炼，但是，这种直接使用 `<area>` 元素覆盖的方法还有一些局限，一是Firefox浏览器不支持（Firefox的 `<area>` 元素默认 `display:none`，且无法重置）；另外一个问题就是无法使用键盘Tab索引访问，如果没有外面的 `<a>` 元素是可以的，有了之后，这种取巧的做法就不行了。

所以，如果你的项目兼容性要求比较高，对无障碍访问要求也比较高，则建议使用透明图片覆盖的方式实现文字的链接嵌套，CSS不变，只要HTML稍微调整下就可以了，如下：

```
<h4 class="book-title">
  
    都市猎人
</h4>
<map id="mapTitle" name="mapTitle">
  <area shape="rect" coords="0,0,200,21" href="/book/1003477570" target="_blank">
</map>
```

也就是把原本裸露的 `<area>` 换成 ``，然后再使用 `<area>` 元素创建图片热点。于是，各大浏览器全兼容版本就这么大功告成了！

三、结束语

对于Chrome以及IE9+浏览器可以对 `<area>` 元素进行样式设置，而Firefox4之后就不可以，我查找网上资料，发现有人认为Firefox浏览器的做法是正确的，包括10年就有人向Mozilla反馈这个bug，但是，似乎Mozilla开发者坚持认为自己是正确的，也就是Firefox浏览器的 `<area>` 永远都是 `display:none`。

下面就是佐证Firefox浏览器的做法是正确的相关内容，引用了大量规范内容：

OK, so let's walk through the specs here. <https://drafts.csswg.org/cssom-view/#dom-element-getClientRects> says:

If the element on which it was invoked does not have an associated layout box return an empty sequence and stop this algorithm.

where “layout box” links to <https://drafts.csswg.org/cssom-view/#layout-box> which says that a “layout box” is either a CSS layout box or an SVG box. <https://drafts.csswg.org/cssom-view/#css-layout-box> then says that this is the same term the CSS specs use.

Going to the HTML spec, <https://html.spec.whatwg.org/multipage/embedded-content.html#the-area-element> doesn't seem to say anything useful on its own (which is not surprising; rendering is described elsewhere in HTML). <https://html.spec.whatwg.org/multipage/embedded-content.html#image-maps> describes association of `<area>` elements with layered shapes on an image, which are not “CSS layout boxes” (not least because they're not actually box-like in any way; they can have pretty arbitrary shapes). There's no SVG involved here, of course.

OK, on to <https://html.spec.whatwg.org/multipage/rendering.html> which is where HTML describes rendering. This says, in <https://html.spec.whatwg.org/multipage/rendering.html#hidden-elements>:

```
area { display: none; }
```

(the actual selector includes other things). Great, that's something to work with. <https://www.w3.org/TR/CSS21/visuren.html#display-prop> says:

none

... Please note that a display of ‘none’ does not create an invisible box; it creates no box at all.

The current css-display draft at <https://drafts.csswg.org/css-display/#valdef-display-none> likewise says:

The element and its descendants generates no boxes.

So as far as I can tell the Firefox behavior is exactly following the specs here: `<area>` elements have no layout boxes, and hence `getClientRects` returns an empty list.

孰对孰错其实没多大意义，我等只要知道除了区域热点，`<area>` 在链接嵌套功能上更有价值皆可安好！



《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

- ES5中新增的Array方法详细说明
- JavaScript实现http地址自动检测并添加URL链接
- 关于锚点跳转及jQuery下相关操作与插件
- 页面可用性之img标签longdesc属性与HTML5
- HTML5块状元素链接以及可用性探讨
- 小tip: 外链地址网站标志图标API应用
- 基于VoiceOver的移动web站无障碍访问实战
- HTML tabindex属性与web网页键盘无障碍访问
- HTML accesskey属性与web自定义键盘快捷访问
- WAI-ARIA无障碍网页应用属性完全展示
- 了解HTML/HTML5中的download属性

分享到:        0

标签: [area](#), [html](#), [map](#), [无障碍网页应用](#), [链接](#)

发表评论（目前16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. civen说道:

2018年09月27日 08:46

你好, 请教下, 为什么在苹果浏览器里面map热点就点不了啦?

[回复](#)



2. tenado说道:

2018年06月5日 17:29

看到你图片是没有后缀的, 请问下是怎么实现的呢? 浏览器是如何识别这是一张图片呢? 搜索了一些资料, 但是关于这个的比较少, 帮解答下, 万分感谢!!!

[回复](#)



张鑫旭说道:

2018年06月5日 17:30

应该是服务器输出文件时候返回contentType类型为图片, 如image/jpeg。

[回复](#)



3. **wayne**说道:

2018年01月23日 21:08

你好~ 最近正在研究HTML的area和map, 请问area或map这两个标签能够发出被点击、鼠标进入之类的页面事件么? 试了一下好像没成功

[回复](#)



4. **jiafu**说道:

2017年12月7日 16:39

大神, 刚试了一下, a标签好像可以嵌套的啊

[回复](#)



5. **君哥哥**说道:

2017年06月14日 12:41

这个用途其实很广泛, 一般用于机器自动生成的页引流页面, 比如邮件营销, 广告策划活动页面, 典型的电商造节的专题页面, 一张鲜明的图片, 有几个链接, 这些需求大, 迭代快速的页面, 要的就是速度与高效, 热力区块刚好够用, 不用人工手动写页面, 市场部做好计划, 设计好稿子, 运营同事直接导入图片, 按照需求划热力区, 自动生成页面, 全程都不需要技术部去投入, 也完全没必要, 早前dw, fireworks都有这功能, 只不过不完善垃圾代码太多, 现在那些电商公司的系统先进多了, 都稳定运行很多年, 相当成熟, 你所看到618, web, app, 内嵌广告, 无数这种页面, 可以抓包看一下, 更多的是混合式的, 部分手写部分机器生成。

[回复](#)



6. **菜菜小白**说道:

2017年05月24日 15:15

前段时间在看 w3c 学习html 看到这个标签 但是解析不详细 只是拿图片练习过跳转和做标签, 解析有点模糊, 这下好了有大神写这篇文章解析这么详细。

[回复](#)



7. **刘瑜**说道:

2017年05月22日 17:06

大神, 月色真美

[回复](#)



8. **bestRenekton**说道:

2017年05月12日 18:00

思路不错

[回复](#)



9. **fuzhiyu**说道:

2017年05月9日 17:39

旭哥6666

[回复](#)



10. **逆风**说道:

2017年05月8日 16:04

大神, 之前没遇到过a链接嵌套, 看完你的这篇文章自己试了一下, a链接可以嵌套啊。个人觉得area还是用在map映射更能发挥作用。把外面的a链接设置成块元素当盒子用, 里面的a设置相对于外面的a绝对定位, 我的能出的来效果。班门弄斧了, 望大神不吝赐教。

[回复](#)



张鑫旭说道:

2017年05月13日 10:05

不能嵌套哦，会总嵌套关系变成非嵌套的，浏览器在解析HTML的时候。

[回复](#)



逆风说道:

2017年05月17日 17:48

感谢大神回复，我那样做就是把外面的a标签单纯的看做一个盒子，这样的话不会影响点击。而area是个单标签，要给文字或者图片附链接的话不大方便

[回复](#)



11. kin说道:

2017年05月8日 11:19

在做网站聊天表情的时候 也用到了map,area,100个表情 每张单独使用小图片，打开表情框之后，请求数量不能直视啊。于是做了一个表情map图，使用area定位到某个表情。后来干脆，表情显示的时候 也不用单张了，直接用了这张map表情图 使用背景图片，这样效率就更好了。不过前提是表情需是静态的。

[回复](#)



mfk说道:

2017年05月8日 16:01

为什么不用css sprite?

[回复](#)



12. 成为海贼王的男人说道:

2017年05月6日 13:12

感觉应用的场景不是很多，作为一个技术点还是可以深藏于心的。

[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](#) ⁽¹⁹⁰⁾
- » [未来必热: SVG Sprite技术介绍](#) ⁽¹¹⁹⁾
- » [粉丝群第1期CSS小测点评与答疑](#) ⁽¹¹⁵⁾
- » [HTML5终极备忘大全 \(图片版+文字版\)](#) ⁽⁹³⁾
- » [让所有浏览器支持HTML5 video视频标签](#) ⁽⁸⁶⁾
- » [Selectivizr-让IE6~8支持CSS3伪类和属性选择器](#) ⁽⁸²⁾
- » [CSS3下的147个颜色名称及对应颜色值](#) ⁽⁷⁹⁾
- » [视区相关单位vw, vh..简介以及可实际应用场景](#) ⁽⁷⁶⁾
- » [写给自己看的display: flex布局教程](#) ⁽⁷⁶⁾
- » [小tips: 纯CSS实现打字动画效果](#) ⁽⁷⁶⁾



今年热议

- » 《CSS世界》女主角诚寻靠谱一起奋斗之人 ⁽⁷⁶⁾
- » 不借助Echarts等图形框架原生JS快速实现折线图效果 ⁽⁶⁴⁾
- » 看，for..in和for..of在那里吵架！ ⁽⁶⁰⁾
- » 是时候好好安利下LuLu UI框架了！ ⁽⁴⁷⁾
- » 原来浏览器原生支持JS Base64编码解码 ⁽³⁵⁾
- » 妙法攻略：渐变虚框及边框滚动动画的纯CSS实现 ⁽³³⁾
- » 炫酷H5中序列图片视频化播放的高性能实现 ⁽³¹⁾
- » CSS scroll-behavior和JS scrollIntoView让页面滚动平滑 ⁽³⁰⁾
- » windows系统下批量删除OS X系统.DS_Store文件 ⁽²⁶⁾
- » 写给自己看的display: flex布局教程 ⁽²⁶⁾

猜你喜欢

- ES5中新增的Array方法详细说明
- JavaScript实现http地址自动检测并添加URL链接
- 关于锚点跳转及jQuery下相关操作与插件
- 页面可用性之img标签longdesc属性与HTML5
- HTML5块状元素链接以及可用性探讨
- 小tip: 外链地址网站标志图标API应用
- 基于VoiceOver的移动web站无障碍访问实战
- HTML tabindex属性与web网页键盘无障碍访问
- HTML accesskey属性与web自定义键盘快捷访问
- WAI-ARIA无障碍网页应用属性完全展示
- 了解HTML/HTML5中的download属性