[点击穿透原理及解决](http://www.cnblogs.com/shytong/p/5463673.html)

**一、事件触发顺序**

　　PC网页上的大部分操作都是用鼠标的，即响应的是鼠标事件，包括mousedown、mouseup、mousemove和click事件。一次点击行为，可被拆解成：mousedown -> mouseup -> click 三步。

　　手机上没有鼠标，所以就用触摸事件去实现类似的功能。touch事件包含touchstart、touchmove、touchend，注意手机上并没有tap事件。手指触发触摸事件的过程为：touchstart -> touchmove -> touchend。

　　手机上没有鼠标，但不代表手机不能响应mouse事件（其实是借助touch去触发mouse事件）。也就是说在移动端的click事件可以拆解为：touchstart -> touchmove -> touchend -> click。

　　浏览器在 touchend 之后会等待约 300ms ，如果没有 tap 行为，则触发 click 事件。 而浏览器等待约 300ms 的原因是，判断用户是否是双击（double tap）行为，双击过程中就不适合触发 click 事件了。 由此可以看出 click 事件触发代表一轮触摸事件的结束。

　　上面说到原生事件中并没有 tap 事件，可以参考经典的 zepto.js 对 singleTap 事件的处理。[见源码 136-143 行](https://github.com/madrobby/zepto/blob/master/src/touch.js#L136-L143)

　　可以看出，singleTap 事件的触发时机 —— 在 touchend 事件响应 250ms 无操作后，触发singleTap。

**二、点击穿透场景及原因**

　　有了以上的基础，我们就可以理解为什么会出现点击穿透现象了。我们经常会看到“弹窗/浮层”这种东西，我做个了个demo。

　　整个容器里有一个底层元素的div，和一个弹出层div，为了让弹出层有模态框的效果，我又加了一个遮罩层。

[复制代码](javascript:void(0);)

<div class="container">

<div id="underLayer">底层元素</div>

<div id="popupLayer">

<div class="layer-title">弹出层</div>

<div class="layer-action">

<button class="btn" id="closePopup">关闭</button>

</div>

</div>

</div>

<div id="bgMask"></div>

[复制代码](javascript:void(0);)

　　然后为底层元素绑定 click 事件，而弹出层的关闭按钮绑定 tap 事件。

[复制代码](javascript:void(0);)

$('#closePopup').on('tap', function(e){

$('#popupLayer').hide();

$('#bgMask').hide();

});

$('#underLayer').on('click', function(){

alert('underLayer clicked');

});

[复制代码](javascript:void(0);)

　　点击关闭按钮，touchend首先触发tap，弹出层和遮罩就被隐藏了。touchend后继续等待300ms发现没有其他行为了，则继续触发click，由于这时弹出层已经消失，所以当前click事件的target就在底层元素上，于是就alert内容。整个事件触发过程为 touchend -> tap -> click。

　　而由于click事件的滞后性（300ms），在这300ms内上层元素隐藏或消失了，下层同样位置的DOM元素触发了click事件（如果是input框则会触发focus事件），看起来就像点击的target“穿透”到下层去了。

**因此，点击穿透的现象就容易理解了，在这 300ms 以内，因为上层元素隐藏或消失了，由于 click 事件的滞后性，同样位置的 DOM 元素触发了 click 事件（如果是 input 则触发了 focus 事件）。在代码中，给我们的感觉就是 target 发生了飘移。**

**三、解决**

**1. 触摸结束时** touchend 事件触发时，preventDefault()。看上去好像没有什么问题，但是，很遗憾的是不是所有的浏览器都支持。

**2. 禁止页面缩放** 通过设置meta标签，可以禁止页面缩放，**部分浏览器**不再需要等待 300ms，导致点击穿透。点击事件仍然会触发，但相对较快，所以 click 事件从某种意义上来说可以取代点击事件， 而代价是牺牲少数用户（click 事件触发仍然较慢）的体验。

<meta name="viewport" content="width=device-width, user-scalable=no">

IE 10可以用 CSS 取消点击穿透的延迟：

html {

-ms-touch-action: manipulation;

touch-action: manipulation;

}

IE 11+ 可以用 touch-action: manipulation; 属性来阻止元素的双击缩放。

**3. CSS3 的方法** 虽然主要讲的是事件，但是有必要介绍一个 CSS3 的属性 —— pointer-events。

pointer-events: auto | none | visiblePainted | visibleFill | visibleStroke | visible | painted | fill | stroke | all | inherit;

pointer-events 属性有很多值，有用的主要是 auto 和 none，其他属性为 SVG 服务。

[查看浏览器支持情况](http://caniuse.com/#search=pointer-events) 可见移动端开发还是可以用的。

| **属性** | **含义** |
| --- | --- |
| auto | 默认值，鼠标或触屏事件不会穿透当前层 |
| none | 元素不再是target，监听的元素变成了下层的元素（如果子元素设置成 auto，点击子元素会继续监听事件） |

**4.延长消失事件**可以利用jquery的fadeout，设置事件大于300ms。

本文参考自：https://segmentfault.com/a/1190000003848737