

您可能不知道的CSS元素隐藏“失效”以其妙用

这篇文章发布于 2012年02月14日, 星期二, 16:35, 归类于 [CSS相关](#)。阅读 128868 次, 今日 15 次 [23 条评论](#)

by zhangxinxu from <http://www.zhangxinxu.com>

本文地址: <http://www.zhangxinxu.com/wordpress/?p=2191>

一、CSS元素隐藏

在CSS中, 让元素隐藏(指屏幕范围内肉眼不可见)的方法很多, 有的占据空间, 有的不占据空间; 有的可以响应点击, 有的不能响应点击。后宫选秀——一个一个看。

```
{ display: none; /* 不占据空间, 无法点击 */ }
```

```
{ visibility: hidden; /* 占据空间, 无法点击 */ }
```

```
{ position: absolute; clip: rect(1px 1px 1px 1px); /* 不占据空间, 无法点击 */ }
```

```
{ position: absolute; top: -999em; /* 不占据空间, 无法点击 */ }
```

```
{ position: relative; top: -999em; /* 占据空间, 无法点击 */ }
```

```
{ position: absolute; visibility: hidden; /* 不占据空间, 无法点击 */ }
```

```
{ height: 0; overflow: hidden; /* 不占据空间, 无法点击 */ }
```

```
{ opacity: 0; filter: Alpha(opacity=0); /* 占据空间, 可以点击 */ }
```

```
{ position: absolute; opacity: 0; filter: Alpha(opacity=0); /* 不占据空间, 可以点击 */ }
```

```
{
    zoom: 0.001;
    -moz-transform: scale(0);
    -webkit-transform: scale(0);
    -o-transform: scale(0);
    transform: scale(0);
    /* IE6/IE7/IE9不占据空间, IE8/FireFox/Chrome/Opera占据空间。都无法点击 */
}
```

```
{
    position: absolute;
    zoom: 0.001;
    -moz-transform: scale(0);
    -webkit-transform: scale(0);
    -o-transform: scale(0);
    transform: scale(0);
    /* 不占据空间, 无法点击 */
}
```

欢迎补充!

以上隐藏方法如果针对最小辈的子元素（膝下无子女），例如 `` 图片元素，则上面的注释说明都是OK的(若有误，请指正)。

我以前就提过，为了便于理解，我们会使用情感化思维，类比现实，在内心形成只属于自己的具象化的想法。这是很OK的，但是，要知道，事物是具有两面的，具象的东西容易让我们陷入固定思维模式，尤其在现实世界中非特殊情境的验证下，从而让我们不经意间产生一些错误的认识。

上面的话有点像老油条——难嚼。要让老少皆懂，很简单，举几个简单的示例就可以了。而这里的示例就是上头几个隐藏的例子：由于具象化的固定思维，您可能并不知道下面要提到的几种CSS元素隐藏“失效”的情况！

二、`display:none`和`visibility:hidden`

我记得校园招聘那会儿，技术面的时候有问我 `display:none` 和 `visibility:hidden` 的区别在哪儿？

“这个问题简单？”我心里头暗自得意，按耐住自己得意又紧张的小心脏，自信满满地说，“这两个声明都可以让元素隐藏，不同之处在于 `display:none` 隐藏后的元素不占据任何空间，而 `visibility:hidden` 隐藏的元素空间依旧存在。”

想想，自己实在是浮躁了点。这种程度的答案说明你在HTML/CSS方面的造诣只是了解这个层面，并不能让面试官眼前一亮。放眼整个中国，写页面的人何其多，像这种程度回答的人一定就像河里的沙子一样，一抓一大把。随使用脚趾头想想也知道，像度娘，企鹅公司花费那么多人力和财力来校园是肯定来挑金子的，不起眼的沙子就让它随波逐流，流到大海暖化石去吧。

目前，我所知道的不同有三点（欢迎补充）：

1. 空间占据
2. 回流与渲染
3. 株连性

第一点，想必都知道；第二点我之前有[多篇译文](#)提过，`display:none` 隐藏产生reflow和repaint(回流与重绘)，而 `visibility:hidden` 没有这个影响前端性能的问题；第三点估计是不少同行不知道的，就是“株连性”方面的差异。

所谓“株连性”，就是如果祖先元素遭遇某祸害，则其子子孙孙无一例外也要遭殃。我顿时想起了《地球反击战》或是《木乃伊之蝎子王》，一旦把母体搞破了，小辈啊、下属啊什么的都瞬间烟消云散。`display:none` 就是“株连性”明显的声明：一旦父节点元素应用了 `display:none`，父节点及其子孙节点元素全部不可见，而且无论其子孙元素如何不屈地挣扎都无济于事。

在实际的web应用中，我们要经常实现一些显示隐藏的功能，由于 `display:none` 本身特性以及jQuery潜在的驱动，使得我们对 `display:none` 这种隐藏特性相当熟知。因此，久而久之会形成比较牢固的情感化认识，并无法避免地将这种认识迁移到其他类似表现属性(eg. `visibility`)的认识上，再加上一些常规经验……

举例来说吧，通常情况下，我们给一个父元素应用 `visibility:hidden`，则其子孙后代也都会全部不可见。于是，我们就会有类似的认识迁移：应用了 `visibility:hidden` 声明下的子孙元素如何不屈地挣扎都摆脱不了不可见被抹杀的命运。而实际上却存在隐藏“失效”的情况。

何时隐藏“失效”？很简单，如果子孙元素应用了 `visibility:visible`，那么这个子孙元素又会刘谦般地显现出来。

您可以狠狠地点击这里：[CSS visibility属性的株连性测试demo](#)



`visibility` 就是这样一个funny的属性。

对比总结：

`display:none` 是个相当惨无人道的声明，子孙后代全部搞死（株连性），而且连块安葬的地方都不留（不留空间），导致全体民众哗然（渲染与回流）。

`visibility:hidden` 则具有人道主义关怀，虽然不得已搞死子孙，但是子孙可以通过一定手段避免（伪株连性），而且死后全尸，墓地俱全（占据空间），国内民众比较淡然（无渲染与回流）。

附上

本节末了，您可以狠狠地点击这里：[display:none株连性测试demo](#)

可以看到demo页面中张美女都跟靓仔过情人节去了。

三、visibility隐藏“失效”妙用

visibility隐藏“失效”的应用其实在老早的时候就有了，等下，让我找找~~

啊哈，找到了。

去年的去年5月份[CSS3模拟window7炫酷界面效果demo](#)，原文点击[这里](#)。



复选框勾选与否对应的部分显示与隐藏。例如，偶们要让父标签的圆角啊、投影啊、背景啊什么的都隐

掉，只留里面的文本框。该怎么实现？如果是要 `display:none`，则里面的文本框框也会被一并抹杀掉的，如果是要脚本控制style，哇哦哦，这么多样式，想想脑袋就大了。毋庸置疑，这里使用 `visibility` 控制是最高效也是最巧妙的方法。

四、height:0和overflow:hidden的组合

`overflow:hidden` 用中文理解就是“溢出隐藏”，也就是盒子以外的内容都咔嚓掉不可见的。加上 `height:0`，只要是一般的非inline水平元素，则元素内部所有子孙都应该是不可见的。这是我们具象化的认识，而且N多的经验验证都是如此，但是实际上是这样子吗？

先看一个反例demo，您可以狠狠地点击[这里](#)：[0高度溢出隐藏“失效”demo](#)



`height:0` 和 `overflow:hidden` 组合隐藏“失效”的条件如下：祖先元素没有 `position:relative/absolute/fixed` 声明，同时内部子元素应用了 `position:absolute/fixed` 声明。

为什么会这样？

先看下CSS2.1规范中对overflow的一点解释：

This property specifies whether content of a block container element is clipped when it overflows the element's box. It affects the clipping of all of the element's content except any descendant elements (and their respective content and descendants) whose containing block is the viewport or an ancestor of the element.

什么意思呢？我琢磨了又琢磨，唉，只叹理解不够深入，也有些理不清。我想大致意思应该这样（若有不对应该指正）：`overflow` 可以剪裁超出块状元素之外的元素。除非超出元素的包含块是整个视区或是该overflow元素的祖先元素。然后绝对定位元素的包含块应该就是含有 `position:relative/absolute/fixed` 的祖先元素。

因此：

```
body
  height: 0; overflow: hidden;
  position: absolute; /* 不会被隐藏 */
```

```
position: relative;
  height: 0; overflow: hidden;
  position: absolute; /* 不会被隐藏 */
```

```
height: 0; overflow: hidden; position: relative;
  position: absolute; /* 会被隐藏 */
```

```
height: 0; overflow: hidden;
  position: relative;
  position: absolute; /* 会被隐藏 */
```

记住这个：`position: absolute` 元素溢出 `overflow: hidden` 元素的时候，如果其第一个含有 `position` 属性(`static` 除外)的祖先元素（一直到 `body`）是 `overflow: hidden` 元素祖先元素的时候，则不隐藏；否则，隐藏。

或者记住这个：火星人在地球触发了法律，如果火星人的老爸在这法律之外，则这个火星人啥事没有；否则，坐牢！

五、height:0和overflow:hidden“失效”妙用

前不久在[“absolute元素在text-align属性下的对齐显示”](#)一文中介绍了一种简单使用 `text-align` 属性实现“返回顶部”按钮定位的方法。

您可以狠狠地点击这里：[text-align:right声明下的返回顶部定位demo](#)

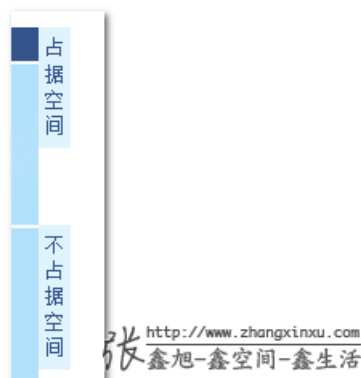
后来有人反映，此方法好是好，但是有一个问题，因为标签内打了个 ` ` 空格，产生了空白高度啊！

如何fix这里的空格产生的高度呢？使用 `line-height: 0;` ？或者使用 `font-size: 0;` ？如果不是因为 `line-height` 和 `font-size` 具有继承性，这确实是个好方法（font-size还有Chrome默认文字无法小于12像素问题，考古版本IE下0像素文字呈现小点的问题）。

经过我多方测试，以在OOCSS开发模式下，这里使用 `height:0` 和 `overflow:hidden` 是最好的方法。

```
h0 ovh tr  
&nbsp; abs
```

您可以狠狠地点击这里：[text-align下返回顶部不占据空间demo](#)



demo中，上面一个条条 ` ` 产生高度，因此头部背景色，而下面这个条条就没有。因为定位元素是绝对定位的，因此，不用担心会被 `overflow` 不小心干掉！

六、情人节结语

据说今天 @淘宝超仁 花了10W在淘宝上买了个广告位向 @倩小莲 表白。



这位哥们真是大手笔啊，我辈实在望尘莫及，只能用毛主席换些德芙了。想了想，挤不出什么话了，就这样吧。

末了，再附上一个 `opacity` 透明度覆盖性测试demo，您可以狠狠地点击这里：[opacity透明度覆盖性测试demo](#)

结果显示，透明度属性会重叠，具有覆盖性。

《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](#)。



« [HTML5 placeholder实际应用经验分享及拓展](#)

[热门: 响应图片\(Responsive Images\)技术简介](#) »

猜你喜欢

- 实验: absolute/display隐藏与回流等性能
- 对overflow与zoom"清除浮动"的一些认识
- CSS 相对/绝对(relative/absolute)定位系列 (三)
- 纯CSS实现侧边栏/分栏高度自动相等
- CSS深入理解流体特性和BFC特性下多栏自适应布局
- 小tip: transition与visibility
- 小tip: 微博新版查看大图前后浏览的另外一种实现
- 深入理解CSS中的层叠上下文和层叠顺序
- CSS clip:rect矩形剪裁功能及一些应用介绍
- CSS hover效果的逆向思维实现
- border-collapse与table td边框opacity透明度诡异解析

分享到: [+](#) [QQ](#) [微信](#) [微博](#) [贴吧](#) [豆瓣](#) [知乎](#) [爱发电](#) [0](#)

标签: absolute, clip, display:none, opacity, overflow:hidden, relative, visibility

发表评论 (目前23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.

老张说道:

2017年09月27日 09:44

我觉得祸害就在这里，实际应用中肯定是知道的，但是表述肯定不是株连性这个叫法。为什么要取很多看起来逼格的而实际无用的名字呢，display:none,不占据空间的意思是没有空间占存了，其子子孙孙也就没有空间去显示了。visibility:hidden本身就占据空间，只是不显示，其子子孙孙还有蹦跶的空间。就像一个人的祖宗本身就是穷屌丝，即使活着，你还让他子孙继承万顷良田？一个富翁即使死了，他的子子孙孙照样可以有万顷良田。

回复




2.

seayao说道:

2017年09月13日 15:16

张大仙儿:
什么时候用display:none,
什么时候用visibility:hidden;position: absolute;呢?
我很少用visibility:hidden，或者说什么情况下用visibility:hidden更好呢?

回复



3.

周顺峰说道:


2017年05月28日 08:42

zoom: 0.001;
-moz-transform: scale(0);
-webkit-transform: scale(0);
-o-transform: scale(0);
transform: scale(0);

//这个元素隐藏的方法解释 经测试有误
IE 环境目前我的系统没办法测 只测了谷歌、火狐、欧朋
//结果显示如下:
/* FireFox/ 占据空间。都无法点击 */
/* chrome Safari Opera 不占据空间 无法点击 */

不知道我测试的结果是否正确？求大神解答~

回复



4.

real说道:

2017年03月3日 09:43

clip-path(0,0,0,0,0,0);.....


回复

real说道:

2017年03月3日 09:47

clip-path: polygon(0,0,0,0,0,0);

回复



5.

陈大文说道:

2016年10月5日 15:31


张老板
{ position: absolute; visibility: hidden; }
这东西我给它个类名叫.hidden就出BUG，给它起名叫.none就正常是什么意思?

回复

张鑫旭说道:

2016年10月8日 10:48

我没遇到过，可否提供demo?



回复

6. the1826说道:

2016年07月7日 12:44

如果想隐藏一个视频元素的顶部20px高的部分，如何控制呢？一直没找到可以控制的CSS，OVERFLOW好像无法控制，TOP可以偏移，但无法隐藏掉。

回复



7. 乐说道:

2016年05月6日 11:23

请问如何设置display: none兼容ie8，上面的设置都放入父级元素，没有效果

回复



8. 李帅说道:

2016年04月29日 16:34

猫眼电影里的表格布局，利用的是ul，有个问题是在content里加了一行“”其样式是：position:static;visibility:hidden;这是为什么呢

回复



李帅说道:

2016年04月29日 16:34

<http://pf.maoyan.com/>

回复



9. 刘亮说道:

2015年11月8日 01:21

记住这个：position: absolute元素溢出overflow: hidden元素的时候，如果其第一个含有position属性(static除外)的祖先元素（一直到body）是overflow: hidden元素祖先元素的时候，则不隐藏；否则，隐藏。是不是可以这样理解，overflow:hidden元素如果是static的话，则position: absolute的子元素则不隐藏，否则隐藏

回复



10. 四季度说道:

2015年08月24日 15:01

Thans 风趣幽默，易于理解！

回复



11. 何必如我说道:

2015年04月28日 18:26

帮大忙了，怒赞！！！！

回复



12. appla说道:

2014年10月23日 09:02

非常感谢

回复



13. **ScriptJava**说道：
2013年09月10日 23:37



很好的文章，不错，看了之后很受启发。

顺便说一下我对overflow:hidden的理解。该属性只会隐藏以拥有该属性的元素为containing block的子级元素。结合楼主的举得四个例子解释一下就明白了。

对于例1：拥有position:absolute属性的containing block是body元素，而不是他的直接父级元素。所以overflow:hidden 失效。

对于例2：拥有position:absolute属性的containing block是拥有position:relative属性的元素，同样，不是他的直接父级元素， overflow:hidden还是失效。

对于例3：对应的containing block是他的直接父元素，所以overflow:hidden生效

对于例4：对应的containing block是他的直接父级元素，而该父级元素的containing block 是拥有overflow:hidden的元素，所以overflow:hidden有效

说白了，overflow: hidden只能隐藏自己的儿孙，对于一些拥有position:absolute属性的元素来说，有可能已经脱离了父元素的控制（虽然html结构上没有脱离），因此也就不收overflow:hidden的影响了。

[回复](#)

14. **Keinez**说道：
2012年02月28日 09:24



今天试图调上传框样式，对input file用了position: absolute; opacity: 0; filter:Alpha(opacity=0)，结果chrome下crash掉了。搞来搞去发现貌似就是这串代码有问题.....无奈只能放弃改变样式==

[回复](#)

15. **ag**说道：
2012年02月19日 20:01



原来是这样~

[回复](#)

16. **d_b**说道：
2012年02月16日 15:15



隐藏文本还可以text-indent:-9999px和line-height:height*3;overflow:hidden;

[回复](#)

17. **akasuna**说道：
2012年02月14日 17:12



文章末尾，“觉得这里的文章不错，希望他一直走下去？”处，标签嵌套错误

[回复](#)

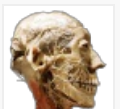
张 鑫旭说道：
2012年02月14日 19:22

@akasuna 是HTML编辑器中换行了

[回复](#)



18. **wordgold**说道：
2012年02月14日 17:11



<http://bingo.taobao.com/valentine/>

那个广告谁都可以

[回复](#)

张鑫旭说道:

2012年02月14日 19:24

@wordgold 原来是这样，唉，信了微博上了传言~~

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 (193)
- » 未来必热: SVG Sprite技术介绍 (120)
- » 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 (80)
- » 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

- 实验: absolute/display隐藏与回流等性能
- 对overflow与zoom"清除浮动"的一些认识
- CSS 相对/绝对(relative/absolute)定位系列 (三)
- 纯CSS实现侧边栏/分栏高度自动相等
- CSS深入理解流体特性和BFC特性下多栏自适应布局
- 小tip: transition与visibility
- 小tip: 微博新版查看大图前后浏览的另外一种实现
- 深入理解CSS中的层叠上下文和层叠顺序
- CSS clip:rect矩形剪裁功能及一些应用介绍
- CSS hover效果的逆向思维实现
- border-collapse与table td边框opacity透明度诡异解析

