

## 小tip: CSS3 animation渐进实现点点点等待提示效果

这篇文章发布于 2013年06月9日, 星期日, 17:20, 归类于 [CSS相关](#)。阅读 88109 次, 今日 12 次 [14 条评论](#)

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

本文地址: <http://www.zhangxinxu.com/wordpress/?p=3411>

### 一、同步/异步操作的等待提示

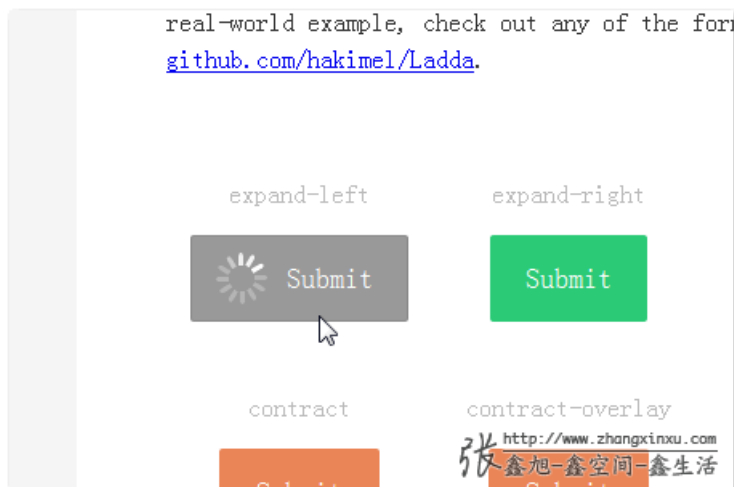
一个名叫“阿贾克斯”的屌丝男在名为“客户端”的学校上学,为了追女神,下狠心,借了高利贷给女神买Mac. 结果,期限已到,要债的找上门,没办法,只能向身在“服务器端”的父母要钱。但是,要钱是要一定的过程和时间的。他必须想办法向要债的表明,钱正在过来的过程中,否则,可能要债的会认为你在忽悠,直接使大招——”Ctrl+W”,瞬间灭了你,眼不见心不烦。

如何告知钱正在过来的过程中呢?

有人可能如下处理,弄个转转转的小图片告诉要债的,看,我在原地不停地转,一直转到钱过来,否则我一刻都停不下来~~



类似下面效果 (点击[这里](#)亲自体验) :



转转转的告知方法优点在于动作大,效果显著。不足在于动作大,实现成本比较高。考虑到IE6~IE8的兼容,如是否撑开按钮高度,是否影响域后面文字的对齐 (vertical-align的解析不规范是最麻烦的) 等,或者再套标签背景图片式定位等,这种提示一般只能在某些特定的重要的场合使用使用,要想一个模子走

到底，此方法严重不推荐。

要是我，直接把手机拿出来，给要债的看父母的短信——“钱汇来中...”



虽然没有惊天地泣鬼神，但是从功能作用上来讲完全够了。

若按钮为提示对象，我的做法是：在按钮默认文字后面加上“中...”或使用宽泛描述“处理中...”（实现整合在组件中）（按钮宽度自适应为前提），例如下面这个：



然而，文字描述虽然具意，但毕竟是静态的，一般情况下，我们的请求时间都是很短的，数秒已经是极限，因此，这种提示完全够用，多半恰到好处。然而，总会遇到堵车的情况——父母没有网银，需要去银行汇款，结果公交车遇到极端主义者汽油烧车，逃命要紧，哪里赶得上汇钱给儿子呢！

但是讨债的那边不知道啊，“钱汇来中...”后等了半个小时，没有其他信息，说不定就等不及，又一次使大招——“Ctrl+W”，瞬间灭了你，眼不见心不烦。

但是，如果信息是动态的，每10分钟自动发过来条类似短信，要债的估计会耐心等的。考虑到现实，父母老人家顾着逃命，还要时不时发个信息过来，有点难为他们老人家了.....

在Gmail中，一个请求如果时间很长，它会在中途有如下提示：“仍在加载中...”。从“加载中...”到“仍在加载中...”的提示文字变化，个人觉得会大大减少减小撕票的可能性。这种nice体验的交互我在手机项目中是经常使用的（考虑到经常会信号不好）。

PC上也是可以类似处理，例如，请求（假设超时设置30秒）发送后可以设定一个 5 秒的定时器，如果请求有返回，清除定时器；如果没有返回，按钮文字修改为“仍在处理中...”；再过 5 秒还没有返回，提示可以是“网络较慢，耐心等待...”等.....然后继续 5 秒~~😁

我们搞这些名堂的目的都是要把“请求中...”提示从纯静态向动态方向靠拢。昨天在折腾某某东西的时候（具体什么已经不记得了），不知为何，脑子突然冒出，诶，是不是可以渐进使用CSS3 `animation`，让提示最后的三个点 “...” 以动画形式呈现，纯字符形式的动画，面向视网膜屏幕，同时无请求，对于 25% Chrome浏览器用户以及其他众多移动设备浏览器用户，岂不是细节体验上的一次提升。

这里的点点点动画效果指的是下图gif动态所演示的效果：



## 二、CSS3 animation与点点点等待动画效果

上gif实现的CSS代码如下：

```
.ani_dot {
    font-family: simsun;
}
:root .ani_dot { /* 这里使用Hack是因为IE6~IE8浏览器下，vertical-align 解析不规范，值为 bottom 或其他会改变按钮的实际高度*/
    display: inline-block;
    width: 1.5em;
    vertical-align: bottom;
    overflow: hidden;
}
@-webkit-keyframes dot {
    0% { width: 0; margin-right: 1.5em; }
    33% { width: .5em; margin-right: 1em; }
```

```

66% { width: 1em; margin-right: .5em; }
100% { width: 1.5em; margin-right: 0;}
}

.ani_dot {
  -webkit-animation: dot 3s infinite step-start;
}

@keyframes dot {
  0% { width: 0; margin-right: 1.5em; }
  33% { width: .5em; margin-right: 1em; }
  66% { width: 1em; margin-right: .5em; }
  100% { width: 1.5em; margin-right: 0;}
}

.ani_dot {
  animation: dot 3s infinite step-start;
}

```

而HTML很简单了：

```
提交订单中>...</span>
```


## demo

您可以狠狠地点击这里：[animation渐进实现点点点动画效果demo](#)

点击demo中下图所示的按钮，效果即出现：



一些说明：

1. IE10+以及其他浏览器，点点点动画消失(better)，IE6-IE9是普通的点点点文字(ok as usual)~~
2. 英文结束字符，也就是键盘逗号和斜杠旁边的字符“.”，在simsun字体下(貌似就是宋体)，所占据的宽度在FireFox/Chrome浏览器下，是当前字体大小的1/2，也就是 `0.5em-1px`，在IE浏览器下几乎等于1/2，也就是 `0.5em`，我的系统是win7. 不知Mac中是否也有如此表现。  

3. 目前，支持 `animation` 的IE10浏览器以及FireFox21已经无需私有前缀了。目前我使用的Chrome25尚未脱离~ 这也使得渐进使用CSS3的一些更高级特性在代码量成本大又减少了！
4. 我起初想使用 `text-indent` 控制，后发现每个字符的间距无法100%精确控制，IE10与FireFox/Chrome似乎有 `1px` 差异，于是，改成的保险的 `width` + `margin` 控制，如果你希望宽度动态可变，直接 `width` 即可。
5. 一般的 `animation` 动画都是连续的，我们这里需要点点点效果是要一帧一帧的， `step-start` 就是用来实现该功能的，后面会有简单介绍。
6. `animation` 缩写的时候，各个属性的顺序其实大可不必在意，貌似浏览器会自己寻找匹配的关键词。到是两个时间——动画周期时间和动画延迟时间——需要前者在前，后者在后。
7. `:root` 为IE9+以及其他现代浏览器Hack, IE6-7甚至IE8下， `inline-block` 水平元素的vertical-align:bottom解析与 `inline` 水平是有差异的，会导致高度撑开，因此， `display: inline-block` 要hack处理，至于其他，实际都无所谓。

## 三、animation的一些参数

### 1. animation-name

就是 `@keyframes` 后面跟着的动画名称，本demo本文中名为 `dot`，意思为“点”。

### 2. animation-timing-function

定义动画播放的方式，参数设置类似 `transition-timing-function`。

常见缓动类型有：先快后慢(OOXX-out)，还是后快先慢(OOXX-in)，还是中间快两头慢(OOXX-in-out)——(OOXX = ease)。这个邪恶记法以前介绍过，不好意思展开解释🙄。

而这里使用的是 `step-start`，顾名思义，“一步一步开始”，表现在动画中就是一帧一帧播放、一顿一顿画面；有时候就是需要这样的效果哦！

### 3. animation-iteration-count

顾名思义：“动画迭代次数”。默认就 1 次，当然，我们可以设置 2 次，3 次，4 次，... 这辈子撸的数目 次。如果撸的次数实在太壮观，建议使用 `infinite` 关键字，表示无限、无穷无尽，撸到天荒地老，铁杵成针！本demo就是这么干的~~😁

### 4. animation-duration

指一个动画周期持续时间。单位秒 `s` 或毫秒 `ms`。

### 5. animation-delay

指动画延时执行时间。单位秒 `s` 或毫秒 `ms`。

### 6. animation-direction

指动画时间轴上帧前进的方向。默认为 `normal`，表示一路向前，往前播放。另外可选值 `alternate`，表示动画往前播放完了之后，然后再倒带，倒带到头了再往后播放，来来回回~~

本demo如果应用 `alternate` 值，那效果就会是：一个点→两个点→三个点→三个点→两个点→一个点.....；而不是目前展示的一个点→两个点→三个点→一个点→两个点→三个点.....

### 7. animation-fill-mode

顾名思义，“动画填充模式”，啥子意思呢？我们装修时候，都见过铺地面砖或地板的，地砖与地砖时间只有缝隙的，我们需要填充，如何个填充法，我们就称之为“填充模式”。



一个动画周期就好比一块地面砖，动画与动画周期之间就会存在类似的缝隙，`animation-fill-mode` 就是确定动画遭遇缝隙时如何“填充”的。一图胜千言：



由图可见（网上的解释都TM简单的敷衍）：

- **none**（默认值），表示动画应用之时、动画延时执行之前之前、以及动画结束之后，元素呈现的都是默认状态。
- **forwards**，前进，表示动画结束后，元素就是当前动画结束时候的状态。对应 `keyframe` 中的 `"to"` 或 `"100%"` 帧。如果应用 `alternate` 值，同时无限或偶数次动画，此时最终 `keyframe` 是 `"from"` 或 `"0%"` 关键帧。
- **backwards**，返回，表示动画开始之前，元素处于 `keyframe` 是 `"from"` 或 `"0%"` 关键帧的状态。由于大多数动画的 `animation-delay` 为0，由于没有指定 `forwards` 状态(如：`both` 值)，因此我们视觉上看到的表现是：动画结束后，动画回到了起始关键帧状态；实际是动画开始之前就如此，而不是结束，万万不可被此假象误导。

要想看清现实，可以设置 `animation-delay` 为非 0 值，我们就可以看到动画开始之前，元素就是 `"0%"` 起始关键帧状态，而不是元素默认状态。直观且准确反映了 `backwards` 的准确释义。

实际应用中，`animation-delay` 设置了非 0 值，同时不是 `step-start` 动画形式，此参数慎用，除非元素默认状态就是起始帧状态，否则动画犹如抽风了一样~🙄

- **both**，`forwards` 和 `backwards` 双P，这是个略考智商的属性，既当爹又当妈的，这可怎么搞

！好搞的，如果要求同一时间既爹又妈，你不是人妖，搞不来。但是白天当爹，晚上当妈，我想相对容易多。这里也是如此，`both` 是与的关系，中文意思是“同时”，表示：动画开始之前是 `"from"` 或 `"0%"` 关键帧；动画完成之后是 `"to"` 或 `"100%"` 关键帧状态。

## 8. animation-play-state

动画运行状态，属性为：`running`（默认）以及 `paused`。这个什么时候有用的，使用 `animation` 实现视频播放效果的时候。视频播放？没错，诸位out了吧，并不是HTML5 `video` 才可以玩转视频哦~~不多讲，啦啦啦啦啦~~🐼

## 四、最后说点啥吧

目前webkit浏览器还需要私有前缀，要不然，我立即放在实际项目中使用了。目前的CSS的代码量，虽然不多，但还是超过自己的忍受程度（代码洁癖啊），而且就功能来讲就是点点点效果，虽然全局公用，但功能点狭隘。

不过，在折腾这篇文章的时候，我有了另外的idea，对于这些现代浏览器渐进增强的CSS，我可以放在一个独立的CSS文件中，然后在页面完成后，或一段时间后，或触发某些条件后，Ajax请求这些CSS字符（如本地木有），干什么呢？使用 `localStorage` 存储在本地，永久存储（当然，版本控制机制还是要有的）。这样，哥就不用担心这些渐进的CSS撑大的基本CSS大小，也一定程度上避免了N请求。由于是渐进增强的东西，因此，即使本地读取速度比从内存差几个档次，也没关系（至少比从服务器请求快吧）。哟，还不错哦~



诶，自己又想到，直接使用 `media query` 查询调用CSS是不是也不错，把渐进CSS放在其中，`media query` 只是个浏览器过滤器，好像也不错。恩，都去试试，都去研究研究~~

本文是偶然想法之后一番折腾的展示、多有卖弄之嫌，然成熟之后，定会实际项目中使用。也希望这里东扯西扯的文字能够对您工作、学习有所启发。也欢迎分享你认为不成熟的想法，什么东西都是从无到有的，互相探讨，在CSS3以及HTML5风起云涌的浪尖开辟全新的道路——哪怕是羊肠小道，时代的弄潮人不一定是技术最牛的，而是敢想敢做的。

明几个就端午了，很多亲们今天就回家了，估计本文发布后，就像掉入窰井盖的花季少女，沉了，个把月后才以“死尸状”浮现出来。哈哈，..... 唉，我自己在这里自言自语个什么，好无聊哦~~🐼

大家端午节快乐！🎉

## 五、重要更新于2013-06-13

@玉面小飞鱼在评论中提到关于进度的想法，很好的想法，而且自己立马就有了idea. 使用CSS `animation` 模拟进度条效果。

我们先看效果吧：

您可以狠狠地点击这里：[CSS animation模拟进度条等待效果demo](#)

demo中三个按钮，第一个按钮模拟超时；第二个为请求小图片（快速loading效果）；第三个为请求大图片（缓慢请求loading效果，明显进度条效果）。



截图看不出效果的精髓的，建议IE10+, FireFox, Chrome26+下亲自体验效果。

CSS `animation` 相关代码如下：

```
.ani_prog::after {
  padding-right: 0%;
  background-color: #34538b;
  background-clip: content-box;
  box-sizing: border-box;
  animation: progress 30s;
}
@keyframes progress {
  0% { padding-right: 100%; }
  1% { padding-right: 50%; }
  3% { padding-right: 30%; }
  10% { padding-right: 20%; }
  20% { padding-right: 8%; }
  30% { padding-right: 5%; }
  40% { padding-right: 4%; }
  60% { padding-right: 2%; }
  80% { padding-right: 1%; }
}
```

从代码可以看出：

1. 进度条为伪元素创建，`::before`，`::after` 正好被支持 `animation` 的浏览器支持，因此，无需 Hack 打补丁，很好很好！
2. 进度条动画最长时间为 `30s`；
3. 进度条开始进度相当快，以应付实际等待时间大多不超过1秒的情况；到后来就越来越慢，但是，仍然有进度，用户会知道仍在加载中，不会中途抛弃。换句话说，此进度条只是一个迷惑“耍债”的假的进度条，但是，效果绝对杠杠的。个人觉得Gmail中的进度条效果，也有这种假象之嫌。

CSS中还有一个重要一段：

```
/* 等待结束直接到底显示 */
.ani_prog_finish::after {
  animation: none;
}
```

页面的等待操作得到成功反馈的时候，建议加上这个类名，作用很简单，让进度条的过程完整（去除animation动画，直接显示最终效果）。否则，如果请求速度较快，用户会感觉进度条只进行了一半，就成功了，有些不自然。

demo中请求小图，虽然请求过程很短，但是依然可以明显看到进度条100%显示之后才隐藏的。

CSS驱动的进度处理，JS仅控制起始与结束状态，成本不高，个人觉得还是很实用的，您可以试试，至少我会在手机、甚至PC上尝试的。

《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

- CSS3实现鸡蛋饼状图loading等待转转转
- CSS3 animation属性中的steps功能符深入介绍
- 文本框邮箱地址自动提示jQuery插件
- HTML5 progress元素的样式控制、兼容与实例
- 伪元素表单控件默认样式重置与自定义大全
- 内容loading加载后高度变化CSS3 transition体验优化
- URL锚点HTML定位技术机制、应用与问题
- 小tips: 纯CSS实现打字动画效果
- 翻译 - CSS Sprites:实用技术还是生厌之物?
- 新版无图片版zxxbox jQuery弹出框插件
- 一些SVG向下兼容优雅降级技术

分享到:        { 0 }

标签: animation, animation-fill-mode, progress, 加载, 动画效果, 提示, 渐进增强

### 发表评论（目前14条评论）

|                                     |                  |
|-------------------------------------|------------------|
| <input type="text"/>                | 名称 (必须)          |
| <input type="text"/>                | 邮件地址(不会被公开) (必须) |
| <input type="text"/>                | 网站               |
| <div></div>                         |                  |
| <input type="button" value="提交评论"/> |                  |

1. OSC余年说道:  
2016年08月6日 14:26

ie9不支持

[回复](#)
2. 简陋的艺术家说道:  
2014年11月19日 10:51

老师, 说实话那个:root没太明白。能否再详细指点迷津一下下? 感激~

[回复](#)

新人说道：  
2015年04月20日 21:15  
那是css3新增加的选择器  
[回复](#)



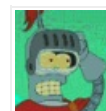
3. **rg**说道：  
2014年05月20日 10:08  
点点点的效果，低版本的手机效果不怎么支持额  
[回复](#)



4. **Alvin**说道：  
2013年08月8日 11:35  
经过我的测试，animation很影响性能。  
[回复](#)



5. **阿拉希**说道：  
2013年08月2日 13:23  
兼容性tips：CSS3里的时间函数steps(n)/step-start/step-stop，在android 2.3以下的内置浏览器里不被支持……  
[回复](#)



6. **海农**说道：  
2013年07月4日 14:15  
鑫哥写的东西真的很好用。  
[回复](#)



7. **玉面小飞鱼**说道：  
2013年06月16日 10:23  
更新后的进度条很有爱，考虑在项目中用一下~  
[回复](#)



8. **一路**说道：  
2013年06月13日 15:43  
鑫哥就是有这样的文笔，一个点能搞出这么多名堂，不错哦，+1  
[回复](#)



9. **blue**说道：  
2013年06月12日 16:36  
前端博客感觉就这还经常更新～都干嘛去了  
[回复](#)



10. **Sean**说道：  
2013年06月11日 16:49  
写的真好!!  
“从“加载中…”到“仍在加载中…”的提示文字变化，个人觉得会大大较少撕票的可能性。”  
“大大较少撕票”？是否为“减少”  
[回复](#)





11. 玉面小飞鱼说道:  
2013年06月10日 09:41



前排占个座儿！

点点点的效果跟转圈圈相比，优势是资源消耗少，但是没有解决点点点时间久了被撕票的问题呐。

有木有可能添加处理进度，比如显示数字0%，10%，....100%，这样会有一种强烈的带入感，不过应该要用js实现，成本又会增加了。

开篇的阿贾克斯骚年的故事很带感！

[回复](#)

12. 靖鸣君说道:  
2013年06月9日 21:23



要是能写个插件处理因兼容性而存在的-webkit,-moz,-o,-ms-(甚至以后的-blink-)前缀，那肯定还是会方便很多的~

[回复](#)

张 鑫旭说道:  
2013年06月13日 10:25



@靖鸣君 随时光流逝，此需会日渐黯淡，吾等应静待而非刻意为之。

[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 scrollToView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

- [CSS3实现鸡蛋饼状图loading等待转转转](#)
- [CSS3 animation属性中的steps功能符深入介绍](#)
- [文本框邮箱地址自动提示jQuery插件](#)
- [HTML5 progress元素的样式控制、兼容与实例](#)
- [伪元素表单控件默认样式重置与自定义大全](#)
- [内容loading加载后高度变化CSS3 transition体验优化](#)
- [URL锚点HTML定位技术机制、应用与问题](#)
- [小tips: 纯CSS实现打字动画效果](#)
- [翻译 - CSS Sprites:实用技术还是生厌之物?](#)
- [新版无图片版xxxbox jQuery弹出框插件](#)
- [一些SVG向下兼容优雅降级技术](#)