

CSS content换行实现字符点点点loading效果

这篇文章发布于 2016年11月13日, 星期日, 20:58, 归类于 [CSS相关](#)。阅读 26375 次, 今日 12 次 [15 条评论](#)

by zhangxinxu from <http://www.zhangxinxu.com/wordpress/?p=5753>

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

一、之前我的字符loading实现

关于字符打点动画实现, 我之前使用过 `box-shadow` 模拟, 还有 `border+background` 模拟, 还有使用 `text-shadow` 实现, 以及今年自认为是最好的实现的利用CSS3 `ch` 单位的实现, 在“[等宽字体在web布局中应用以及CSS3 ch单位](#)”这篇文章有介绍, [demo在这里](#), 效果类似下面(偷懒, 非真实截图, 以前做的图):

提交订单中.

然后, 我今天才发现, 原来还真是一山还有一山高, 有个更巧妙的字符打点动画, 利用CSS content内容生成技术以及CSS3 animation实现的, 并且几乎没有任何的不足, 这里给大家展示下如何实现的。

CSS代码如下:

```
dot {
  display: inline-block;
  height: 1em; line-height: 1;
  text-align: left;
  vertical-align: -.25em;
  overflow: hidden;
}
dot::before {
  display: block;
  content: '...\A..\A.';
  white-space: pre-wrap; /* 也可以是white-space: pre */
  animation: dot 3s infinite step-start both;
}
@keyframes dot {
  33% { transform: translateY(-2em); }
  66% { transform: translateY(-1em); }
}
```

HTML代码如下:

订单提交中<dot>...\</dot>

然后成就即达成! 并且IE6-IE9完美自动向下兼容(静态3个点), IE10+就是动画。

你可以狠狠地点击这里: [content字符生成配合CSS3 animation点点点动画demo](#)

二、content换行技术是知道的, 关键是创意和思路

如果把上面使用的技术拆分下, 其实都是知道的。两个要点, 一是content字符内容生成的换行实现, 二

是CSS3 animation动画。

1. 关于 `content` 换行技术在我之前的这篇“[使用CSS\(Unicode字符\)让inline水平元素换行](#)”文章有详细介绍，其中 `'\A'` 其实指的是换行符中的LF字符，其unicode编码是 `000A`，在CSS `content` 属性中则直接写作 `'\A'`；换行符除了LF字符还有CR字符，其Unicode编码是 `000D`，在CSS `content` 属性中则直接写作 `'\D'`。CR字符和LF字符分别指回车（CR）和换行（LF）。

2. 关于CSS3 `animation` [介绍的文章](#)就更多了。

CSS这么语言的精彩之处就在于，要想学得好，不仅需要技术积累，细节扎实，而且还需要创意和思路，所以，我常说CSS要想学得好是需要点那么不可言喻的天赋的，我想这种天赋就是创造性思维和发散性思维，这和一板一眼的逻辑语言不一样，毕竟我们是为精彩纷呈的效果和各式各样的布局服务的。

回到这里，我有必要好好反思下，字符打点动画我这些年时不时就思考有没有更好的实现，虽然我知道 `content` 内容换行技术以及熟知CSS3 animation属性，但是，我却没有想到这种实现，说明自己的思路还是不够开阔，个中关联没有能够联系在一起，唉，对自己产生了一丝丝的怀疑.....

本文利用 `content` 实现字符打点loading效果我是在这个github项目看到的：[tawian/text-spinners](#)

当然，我自己还有一丝欣慰的，就是站在巨人的肩上，我细节要比原实现要好，好在没有使用任何hack技巧实现的所有浏览器的全兼容（不支持animation的浏览器静态三个点），原来的实现直接忽略了IE9-IE6浏览器，直接没有点。

原来的实现 `content` 是使用的 `` 元素，CSS部分使用的是 `::after` 伪元素，`display` 设置为 `inline-table`，第1行是 `1` 个点，代码如下：

```
<span class="loading"></span>

.loading::after {
  display: inline-table;
  content: "\A.\A.\A...";
  white-space: pre;
  animation: spin4 2s steps(4) infinite;
}
```

而我的实现的是使用自定义的 `<dot>` 元素，CSS部分使用的是 `::before` 伪元素，`display` 设置为 `block`，第1行是 `3` 个点，代码如下：

```
<dot>...</dot>

dot::before {
  display: block;
  content: '...\A.\A.';
  white-space: pre-wrap;
  animation: dot 3s infinite step-start both;
}
```

使用自定义 `<dot>` 元素的好处是IE7,IE8浏览器根本就不认识这个元素，相关CSS会忽略，直接显示3个点；使用 `::before` 伪元素，`display` 设置为 `block` 原因在于其他现代浏览器可以把原来的 `3` 个点推到看不见的最下面，显示的是 `content` 插入的第1行的 `3` 个点。于是，全浏览器都显示良好。

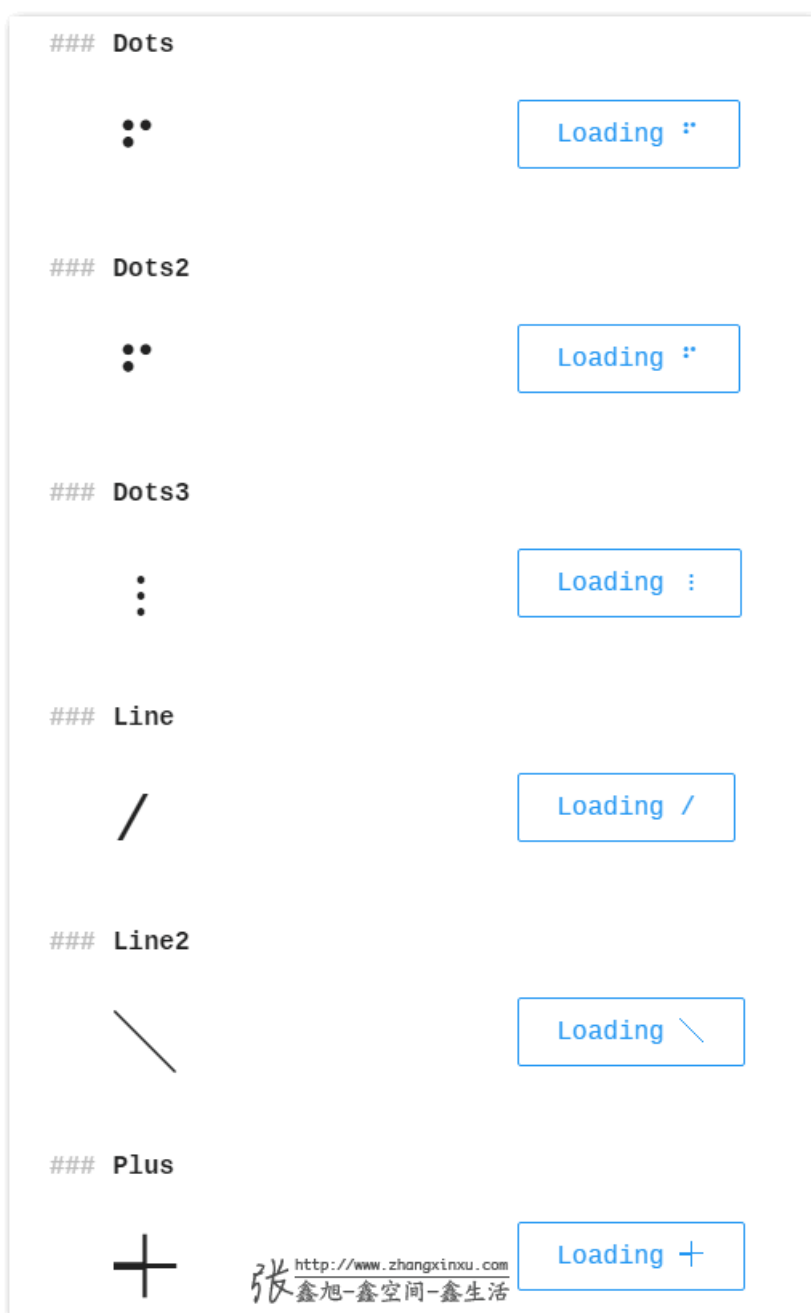
由此可见，创意固然重要，基础和细节也是不可忽视的，这样，才能实现完美的效果——现在这种实现，外面标签的宽度就是 `3` 个点的宽度，不要担心超出或剪裁，字符实现，和前面的文字字体，颜色保持

一致，完美！



三、content字符生成配合CSS3 animation各类字符loading效果实现

其实，`content` 字符生成配合CSS3 `animation` 可以实现的不仅仅是打点这一种字符动画效果，你使用其他一些有意思的字符就能时间更多有意思的loading效果，如下截图（图截自[这个演示页面](#)）：



例如，我们命令行中常见的loading动画实现只要 `content` 设置为下面这个值就可以：

```
.loading::after {
  display: inline-table;
  content: "\A-\A\\A|";
  white-space: pre;
  animation: spin4 2s steps(4) infinite;
}
```

这里就不一一介绍了，有兴趣的同学可以直接去这个项目围观。

四、结束语

学习和成长就是这么回事，身在舒适区，总感觉自己挺好的，觉得自己干活麻利，技术不错，实际上还差的远呢。就好比这里的打点动画的实现，以前使用 `text-shadow` 实现的时候，觉得挺厉害，佩服自己的创意和思路；然后今年7月份研究 `ch` 单位的时候，发现之前头疼的宽度处理有了解决方法，自认为很赞的 `text-shadow` 实现“啪”得打脸了，然后觉得 `ch` 实现基本上就是到头的最佳实现；然后才短短4个月，又再次打脸，使用 `content` 换行字符模拟显然要更上一层楼，虽然此刻自己又觉得这种实现应该是最佳实现了，但我心里清楚，什么时候又会出现更好的实现打脸。



所以，戒骄戒躁，学无止境，永不自满！

最后，我们不妨把自己的思维发散下，上面诸多字符loading虽然除了水平打点动画以外，其他的效果都比较粗糙，难以直接在项目中使用，但是，要知道，我们不是有 `icon fonts` 技术吗，可以把粗糙的字符映射成精致的图形。于是乎，我们要想实现一些更细腻的图形动画，就很简单啦，是不是可以搞个自定义的图形字体，然后利用这里的loading原理，很多很可爱细腻的动画效果是不是就出来了！比方说点赞的时候，放点礼花；点击删除图标，垃圾桶打开动画，对不对，实现都不是难事，都可以用起来！这种实现比以前 `background-image` 或者JS实现要更加简单可控（大小，颜色什么的），反正我是会找机会在项目用一用的，你们自己看着办啦！嘻嘻.....

感谢阅读，欢迎交流！

《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

- 小tip: 使用CSS(Unicode字符)让inline水平元素换行
- 再说CSS3 animation实现点点点loading动画
- 小tips: 纯CSS显示JS/HTML等源代码行数
- 关于文字内容溢出用点点点(...)省略号表示
- CSS unicode-range特定字符使用font-face自定义字体
- CSS content内容生成技术以及应用
- CSS3 box-shadow实现纸张的曲线投影效果
- CSS之before, after伪元素特性表现两则
- :after伪类+content内容生成经典应用举例
- 伪类+js实现CSS3 media queries跨界准确判断
- 纯CSS实现帅气的SVG路径描边动画效果

分享到:        0

标签: :before, after, animation, content, Unicode编码, 字符, 字符换行, 点点点

发表评论（目前15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. 威乙创客说道:
2017年04月12日 16:07

大神给力

[回复](#)



2. sol说道:
2017年02月20日 21:04

膜拜大神.... 这个真是思路开阔

[回复](#)



3. 小林说道:
2017年02月9日 13:46

大神结婚了没 考虑考虑我哈哈哈哈哈

[回复](#)



luogege说道:

2017年06月2日 23:27

一看就是对大神文章了解不多的人，哈哈

[回复](#)



4. angel说道:

2016年12月23日 10:46

content: ‘...\A..\A.’ \A是什么意思啊

[回复](#)



5. 阿磊说道:

2016年12月15日 15:25

后面的省略号，垂直居中不是更好看吗？（逃..

[回复](#)



阿磊说道:

2016年12月15日 15:26

然后感觉可以给dot设置个padding-left

[回复](#)



6. 苒说道:

2016年12月13日 09:56

涨知识了，感谢

[回复](#)

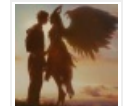


7. yangzj1992说道:

2016年11月23日 14:23

哈哈，赞。真的不要轻易说某方案是最佳实现啊，戒骄戒躁

[回复](#)



8. 爱踩键盘的猫说道:

2016年11月14日 15:51

鑫爷，抱歉打扰= =有个问题想请教你一下~关于JS正则表达式对象方法的问题。我写了这么一段代码:

```
var pattern = /ha/g;
var text = "ha li ha pi ha ha";
var result = new Array();
var match = pattern.exec(text);
while( pattern.test(text) ){
    result.push(match);
    match = pattern.exec(text);
}
```

console.log(result); //输出结果是: "ha,ha,,ha"

实在搞不懂= =讲道理应该输出的是“ha,ha,ha,ha”呀.....求解答T^T.....

[回复](#)



sduper说道:

2016年11月15日 10:24

在循环里面log下pattern.lastIndex就会明白为什么了，正则在全局(g)模式下，test，exec，没执行次都会改变lastIndex的值的

[回复](#)



爱踩键盘的猫说道：
2016年11月15日 14:53

这.....原来exec和test的pattern.lastIndex值是共享的呀.....这样就说得通了.....谢谢啦！

[回复](#)



9. 盖大楼说道：

2016年11月14日 14:55

学习了，确实通过一个自定义字体一行，快速的换行是先动画，就像小时候看的小人书。

[回复](#)



10. 李帅说道：

2016年11月13日 22:15

自定义元素的好处是IE7,IE8浏览器根本就认识这个元素？这句话是IE7、IE8本来就认识还是根本不认识？

[回复](#)



张 鑫旭说道：
2016年11月13日 22:50

感谢纠正~

[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

- 小tip: 使用CSS(Unicode字符)让inline水平元素换行
- 再说CSS3 animation实现点点点loading动画
- 小tips: 纯CSS显示JS/HTML等源代码行数
- 关于文字内容溢出用点点点(...)省略号表示
- CSS unicode-range特定字符使用font-face自定义字体
- CSS content内容生成技术以及应用
- CSS3 box-shadow实现纸张的曲线投影效果
- CSS之before, after伪元素特性表现两则
- :after伪类+content内容生成经典应用举例
- 伪类+js实现CSS3 media queries跨界准确判断
- 纯CSS实现帅气的SVG路径描边动画效果