在Web中使用什么[单位](http://www.w3.org/Style/Examples/007/units.en.html)来定义页面的字体大小，至今天为止都还在激烈的争论着，有人说[PX](http://inamidst.com/stuff/notes/csspx)做为单位好，有人说[EM](http://jontangerine.com/log/2007/09/the-incredible-em-and-elastic-layouts-with-css)优点多，还有人在说[百分比](http://css-discuss.incutio.com/wiki/Using_Percent)方便，以至于出现了[CSS Font-Size: em vs. px vs. pt vs. percent](http://kyleschaeffer.com/best-practices/css-font-size-em-vs-px-vs-pt-vs/)这样的PK大局。不幸的是，仍然有不同的利弊，使各种技术都不太理想，但又无法不去用。真是进也难，退也难呀。

最近在学习em的相关知识的时候，无意之间让我拾得一宝，就是使用**rem**来设置Web页面的字体大小。让我一下子就来劲了，一口气看完并测试了一回，还真是爽歪歪的呀。师傅说好东西不能吃独食，于我就在这里给大家吹吹这个从没见过的**REM**。

在详细介绍**rem**之前，我们先一起来回顾一下我们常用的两种记量单位，也是备受争论的两个：

1. PX为单位
2. EM为单位
3. **PX为单位**

在Web页面初期制作中，我们都是使用“px”来设置我们的文本，因为他比较稳定和精确。但是这种方法存在一个问题，当用户在浏览器中浏览我们制作的Web页面时，他改变了浏览器的字体大小，这时会使用我们的Web页面布局被打破。这样对于那些关心自己网站可用性的用户来说，就是一个大问题了。因此，这时就提出了使用“em”来定义Web页面的字体。

1. **em为单位**

前面也说了，使用是“px”为单位是比较方便，而又一致，但在浏览器中放大或缩放浏览页面时会存在一个问题，要解决这个问题，我们可以使用“em”单位。[Richard Rutter'](http://clagnut.com/about)在《[How to size text using ems](http://clagnut.com/blog/348/)》一文中有做过详细的介绍，追至早一点，[Richard Rutter](http://www.alistapart.com/authors/r/rrutter)也在《[How to Size Text in CSS](http://www.alistapart.com/articles/howtosizetextincss/)》中进行过深入的剖析。

这种技术需要一个参考点，一般都是以<body>的“font-size”为基准。比如说我们使用“1em”等于“10px”来改变默认值“1em=16px”，这样一来，我们设置字体大小相当于“14px”时，只需要将其值设置为“1.4em”。

body {

font-size: 62.5%;*/\*10 ÷ 16 × 100% = 62.5%\*/*

}

h1 {

font-size: 2.4em; */\*2.4em × 10 = 24px \*/*

}

p {

font-size: 1.4em; */\*1.4em × 10 = 14px \*/*

}

li {

font-size: 1.4em; */\*1.4 × ? = 14px ? \*/*

}

为什么“li”的“1.4em”是不是“14px”将是一个问号呢？如果你了解过“em”后，你会觉得这个问题是多问的。前面也简单的介绍过一回，在使用“em”作单位时，一定需要知道其父元素的设置，因为“em”就是一个相对值，而且是一个相对于父元素的值，其真正的计算公式是：

**1 ÷ 父元素的font-size × 需要转换的像素值 = em值**

这样的情况下“1.4em”可以是“14px”,也可以是“20px”，或者说是“24px”，总之是一个不确定值，那么解决这样的问题，要么你知道其父元素的值，要么呢在任何子元素中都使用“1em”。这样一来可能又不是我们所需要的方法。

这里我只是简单的介绍了一个这两个单位的使用，具体一点的大家可以参阅：

1. [Best Practices](http://kyleschaeffer.com/category/best-practices/)的站长[Kyle](http://www.w3cplus.com/css3/kyleschaeffer.com/about-me/)的《[CSS Font-Size: em vs. px vs. pt vs. percent](http://kyleschaeffer.com/best-practices/css-font-size-em-vs-px-vs-pt-vs/)》
2. [Converting px into percentage and em for relative CSS font sizes](http://www.hubbers.com/index.php/converting-px-into-percentage-and-em-for-relative-css-font-sizes/)
3. [Em Vs Percent Widths](http://css-discuss.incutio.com/wiki/Em_Vs_Percent_Widths)
4. [CSS: Units of Measurement](http://www.guistuff.com/css/css_units.html)
5. [Jennifer Kyrnin](http://webdesign.about.com/bio/Jennifer-Kyrnin-5105.htm)的[Using Points, Pixels, Ems, or Percentages for CSS Fonts](http://webdesign.about.com/cs/typemeasurements/a/aa042803a.htm)
6. **Rem为单位**

[CSS3](http://www.w3.org/TR/css3-values/)的出现，他同时引进了一些新的[单位](http://www.w3.org/TR/css3-values/#rem-unit)，包括我们今天所说的[**rem**](http://www.w3.org/TR/css3-values/#rem-unit)。在[W3C](http://www.w3.org/TR/css3-values/#rem-unit)官网上是这样描述[**rem**](http://www.w3.org/TR/css3-values/#rem-unit)的——**“font size of the root element”**。下面我们就一起来详细的了解[**rem**](http://www.w3.org/TR/css3-values/#rem-unit)。

前面说了“em”是相对于其父元素来设置字体大小的，这样就会存在一个问题，进行任何元素设置，都有可能需要知道他父元素的大小，在我们多次使用时，就会带来无法预知的错误风险。而[**rem**](http://www.w3.org/TR/css3-values/#rem-unit)是相对于根元素<html>，这样就意味着，我们只需要在根元素确定一个参考值，，在根元素中设置多大的字体，这完全可以根据您自己的需，大家也可以参考下图：

[](http://pxtoem.com/)

我们来看一个简单的代码实例：

html {font-size: 62.5%;*/\*10 ÷ 16 × 100% = 62.5%\*/*}

body {font-size: 1.4rem;*/\*1.4 × 10px = 14px \*/*}

h1 { font-size: 2.4rem;*/\*2.4 × 10px = 24px\*/*}

我在根元素<html>中定义了一个基本字体大小为62.5%（也就是10px。设置这个值主要方便计算，如果没有设置，将是以“16px”为基准 ）。从上面的计算结果，我们使用“rem”就像使用“px”一样的方便，而且同时解决了“px”和“em”两者不同之处。

1. **浏览器的兼容性**

[**rem**](http://www.w3.org/TR/css3-values/#rem-unit)是[CSS3](http://www.w3.org/TR/css3-values/)新引进来的一个度量单位，大家心里肯定会觉得心灰意冷呀，担心浏览器的支持情况。其实大家不用害怕，你可能会惊讶，支持的浏览器还是蛮多的，比如：[Mozilla Firefox 3.6+](http://firefox.com/)、[Apple Safari 5+](http://apple.com/safari/)、[Google Chrome](http://google.com/chrome)、[IE9+](http://windows.microsoft.com/en-US/internet-explorer/products/ie/home)和[Opera11+](http://www.opera.com/)。只是可怜的IE6-8无法，你们就把他们当透明了吧，我向来都是如此。

不过使用单位设置字体，可不能完全不考虑IE了，如果你想使用这个REM，但也想兼容IE下的效果，可你可考虑“px”和“rem”一起使用，用"px"来实现IE6-8下的效果，然后使用“Rem”来实现代浏览器的效果。就让IE6-8不能随文字的改变而改变吧，谁让这个Ie6-8这么老呢？哈。。。。大家不仿试试，还蛮有意思，说不定这个就是主流的度量单位了。