[ie6 png 透明终极解决方案](http://www.cnblogs.com/yjzhu/archive/2012/11/09/2762059.html)

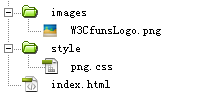
“**珍惜生命，远离 ie6**”，ie6 中的 bug 令很多前端开发人员实为头疼，因此不知道烧了多少脑细胞，在众多的 bug 中最令人抓狂的就是对 png 图片的不支持，导致设计师和重构师放弃了很多很炫的效果，为了让大家在这方面少烧点脑细胞，KwooShung 我就舍身一把，把当前最流行最实用的 ie6 支持 png 图片的方案整理到本贴内，以供大家学习参考！

**方案1 - 滤镜解决方案：**

**介绍：**

滤镜从IE4.0被微软正式引入，所以我们可以使用滤镜解决IE6的PNG透明问题，滤镜不仅可以实现目前CSS3的一些旋转效果而且还可以引入图片。注意：此方法在部分版本的IETest中无效，建议使用标准的IE6来进行测试！

**目录说明：**



**思路：**

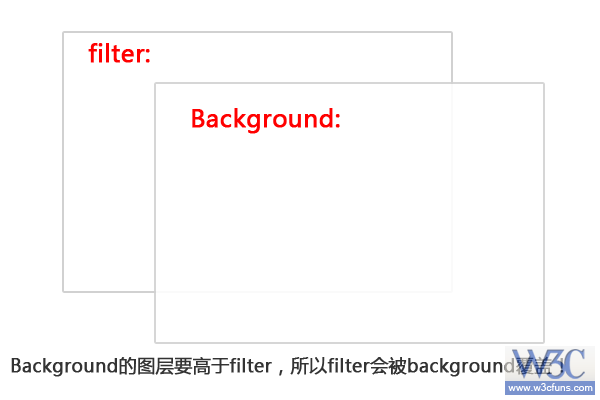
1、书写正常的CSS代码，通过background导入图片，这样所有的浏览器均使用了此PNG图片；

background:url(../images/W3CfunsLogo.png);

2、通过滤镜对引入图片，滤镜引入图片的时候是相对于HTML文件，而不是相对于CSS文件，语法如下：

filter:progid:DXImageTransform.Microsoft.AlphaImageLoader(src="images/W3CfunsLogo.png");

代码写到这里，我们放到IE6下测试后发现IE6还是没有透明，因为我们虽然设置了滤镜引入图片，但是background也同样加载了此图片，又因为background的图层比滤镜设置的高，所以才没有显示出来，如下图：



3、所以我们得出的结论就是当我们使用filter的时候，就要使background失效，因此我们可以使用 [CSSHack](http://www.cnblogs.com/yjzhu/archive/2012/11/06/2756395.html" \o "css hack) 来解决此问题（如果您不知道IE6的 [CSSHack](http://www.cnblogs.com/yjzhu/archive/2012/11/06/2756395.html" \o "css hack) 如何使用的话，请看[这里！](http://www.cnblogs.com/yjzhu/archive/2012/11/06/2756395.html" \o "css hack)），只需要将IE6的background:none;即可，那么可以得出的代码如下：

\_background:none; /\*此代码只有IE6识别\*/

又因为filter只在IE6下让其产生作用，IE6+版本的浏览器虽然也识别filter，但是png透明是没有灰底问题的，所以我们同样将filter也加上IE6 Hack即可。

4、最终我们可以得到如下代码：

[复制代码](javascript:void(0);)

#pics{

background:url(../images/W3CfunsLogo.png) no-repeat;

/\*以下为IE6设置PNG透明代码\*/

\_background:none;

\_filter:progid:DXImageTransform.Microsoft.AlphaImageLoader(src="images/W3CfunsLogo.png");

}

[复制代码](javascript:void(0);)

**提示：**如果需要使其支持链接的hover，那么需要在CSS中定义：cursor:pointer;使其呈现手型，否则将为默认的鼠标状态。

**优点：**

1. 绿色无插件；
2. 效率高，速度快；
3. 网速慢的时候，不会出现先灰底再透明的情况，支持远程图片；
4. 支持Hover等伪类，但是得使用两张图片，网速慢的情况下，会导致第二张图片暂时无法显示，因为还没有完全载入；

**缺点：**

1. 不支持平铺，虽然filter有sizingMethod="scale", 拉伸缩放模式，但是图片会变形，如果单纯的颜色或简单的渐变色还能横向平铺；
2. 不支持Img标签；
3. 不支持CSS Sprite；

**使用情况：**

1. 当没有img引入png时可考虑；
2. 当没有CSS Sprite需求时可考虑；
3. 当没有平铺需求时候可考虑；

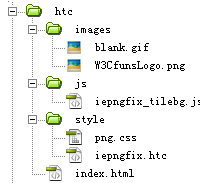
[滤镜解决方案 - DEMO入口](http://www.w3cfuns.com/demos/forum/2011-10/IE6_PNG/filter/)

**方案2 - HTC插件解决方案：**

**介绍：**

从IE 5.5版本开始，Internet Explorer（IE）开始支持Web 行为的概念。这些行为是由后缀名为.htc的脚本文件描述的，它们定义了一套方法和属性，程序员几乎可以把这些方法和属性应用到HTML页面上的任何元素上去。

**目录说明：**



**思路：**

1、首先下载压缩文件 [htc.zip](http://www.w3cfuns.com/forum.php?mod=attachment&aid=NTE2MXw4NTNmNjc3YXwxMzUyNDI0NDIzfDB8Mjk3)

2、复制并粘贴iepngfix.htc和blank.gif到您的网站文件夹中

3、在需要使用的PNG标签上定义如下，**相对于HTML文件的位置 （不相对于CSS文件！）**。例如，你可能看起来像这样：

<style type="text/css">

img,div{behavior:url(style/iepngfix.htc);}

</style>

4、如果您的网站使用的子文件夹，打开。HTC文件，大约在第16行更改blankImg变量，修改blank.gif路径像这样：**同样路径相对于HTML文件的位置 （不相对于CSS文件！）。**

IEPNGFix.blankImg = "images/blank.gif";

5、复制并粘贴iepngfix.htc和blank.gif到您的网站文件夹中。

6、由于此js只有使用IE6时才有用，所以为了让我们的页面更加高效的执行，我们可以将上方代码修改如下，只有IE6的时候才调用执行此JavaScript：

<!--[if IE 6]><script type="text/javascript" src="../js/iepngfix\_tilebg.js"></script><![endif]-->

**优点：**

1. 一次性配置好，只需要像平时一样引入png图片，也不需要考虑png相对于html路径的问题，当目录有所变化，只需要修改htc文件或css中htc文件路径即可。
2. 支持平铺属性。
3. 不支持Img标签；
4. 不支持Hover等伪类；

**缺点：**

1. 多引入了js、图片和htc，共三个文件；
2. 不支持CSS Sprite；
3. 当文件载入之前，会先暂时呈现灰底；

**使用情况：**

1. 当没有img引入png时可考虑；
2. 当没有CSS Sprite需求时可考虑；
3. PNG图片比较频繁修改时可考虑；

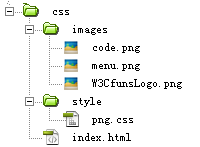
[htc解决方案 - DEMO入口](http://www.w3cfuns.com/demos/forum/2011-10/IE6_PNG/htc/)

**方案3 - 纯CSS解决方案：**

**介绍：**

虽说是纯CSS解决方案，但是也使用了JavaScript来运算，只不过是将脚本写到了CSS文件中，遗憾的是，此方案只支持img标签，对背景图片无效。

**目录说明：**



**思路：**

1、首先下载透明的图片文件 [blank.zip](http://www.w3cfuns.com/forum.php?mod=attachment&aid=NTAxNHw1NDNhMjJiOHwxMzUyNDI0NDIzfDB8Mjk3)

2、在需要设置透明的样式中加入下方代码，其中蓝色标注代码为刚才下载的透明图片，路径同样还是**相对于HTML文件的位置 （不相对于CSS文件！）**：

[复制代码](javascript:void(0);)

img{

\_azimuth:expression(this.pngSet?this.pngSet=true:(this.nodeName == "IMG" && this.src.toLowerCase().indexOf('.png')>-1?(this.runtimeStyle.backgroundImage = "none",this.runtimeStyle.filter = "progid:DXImageTransform.Microsoft.AlphaImageLoader(src='" + this.src + "', sizingMethod='image')",this.src = "images/blank.gif"):(this.origBg = this.origBg? this.origBg :this.currentStyle.backgroundImage.toString().replace('url("','').replace('")',''),this.runtimeStyle.filter = "progid:DXImageTransform.Microsoft.AlphaImageLoader(src='" + this.origBg + "', sizingMethod='crop')",this.runtimeStyle.backgroundImage = "none")),this.pngSet=true);

}

[复制代码](javascript:void(0);)

**优点：**

1. CSS代码看起来似乎很优雅，至少没有乱七八糟的文件了，基本没有外加的文件，效率还算不错。

**缺点：**

1. 多引入了一个本不应该存在的blank.gif图片文件；
2. 不支持背景图即Background；
3. 当文件载入之前，会先暂时呈现灰底；
4. 不支持Hover等伪类；

**使用情况：**

1. 大部分透明的png存在于img标签中时可考虑；
2. 如果有背景图的可以参考上面所说的支持背景图的两种方式；

[纯css解决方案 - DEMO入口](http://www.w3cfuns.com/demos/forum/2011-10/IE6_PNG/css/)

**方案4 - 原生JavaScript解决方案：**

**介绍：**

利用了方案1的滤镜原理来实现，但由于此javascript没有读取css文件中的样式，所以此方案同样只支持img标签，对背景图片无效。

**目录说明：**



**思路：**

1、首先下载透明此方案所用到的js文件 [iepngfix.zip](http://www.w3cfuns.com/forum.php?mod=attachment&aid=NTAxNnw0ODM4YzQ5ZXwxMzUyNDI0NDIzfDB8Mjk3)

2、由于此js只有使用IE6时才有用，所以为了让我们的页面更加高效的执行，我们可以将上方代码修改如下，只有IE6的时候才调用执行此JavaScript：

<!--[if IE 6]><script type="text/javascript" src="js/iepngfix.js"></script><![endif]-->

**优点：**

1. 代码看起来**似乎**很优雅，基本没有外加的文件，效率还算不错。

**缺点：**

1. 额外加入了js文件，增加http请求；
2. 不支持背景图即Background；
3. 当文件载入之前，会先暂时呈现灰底；
4. 不支持Hover等伪类；

**使用情况：**

1. 大部分透明的png存在于img标签中时可考虑；
2. 如果有背景图的可以参考上面所说的支持背景图的两种方式；

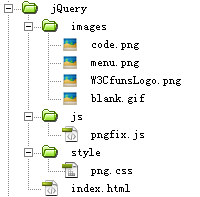
[原生JavaScript解决方案 - DEMO入口](http://www.w3cfuns.com/demos/forum/2011-10/IE6_PNG/javascript/)

**方案5 - jQuery解决方案：**

**介绍：**

jQuery为我们带来了很大的方便，jQuery没有让我们有太大的失望，img和png都同时得以支持，唯一美中不足的还是无法平铺，无法使用CSS Sprite。

**目录说明：**



**思路：**

1、首先下载此方案所用到的js文件和透明gif [jQueryPngFix.zip](http://www.w3cfuns.com/forum.php?mod=attachment&aid=NTAxN3wwN2I2MDBjMnwxMzUyNDI0NDIzfDB8Mjk3)

2、找到js文件中找到blankgif: 'images/blank.gif'，将路径修改为**相对于HTML文件的位置 （不相对于CSS或js文件！）**

3、由于此js只有使用IE6时才有用，所以为了让我们的页面更加高效的执行，我们可以将上方代码修改如下，只有IE6的时候才调用执行此JavaScript：

<!--[if IE 6]><script type="text/javascript" src="js/pngfix.js"></script><![endif]-->

**优点：**

1. CSS代码看起来很优雅，只需要引入js进行简单的配置一下就行了，效率还算不错；
2. 支持背景图，支持img；

**缺点：**

1. 额外加入了js文件和图片文件，增加http请求；
2. 加载了一个庞大的jQuery类库；
3. 多库共存的时候可能会出现问题；
4. 不支持平铺；
5. 不支持CSS Sprite；
6. 当文件载入之前，会先暂时呈现灰底；
7. 不支持Hover等伪类；

**使用情况：**

1. 当您的项目中使用jQuery的时可以考虑；

[jQuery解决方案 - DEMO入口](http://www.w3cfuns.com/demos/forum/2011-10/IE6_PNG/jQuery/)

**方案6 - PNG8格式的图片解决方案：**

**介绍：**

png8和gif都是8位的透明度，IE6与生俱来就支持png8的索引色透明度，但不支持png或8位以上的 alpha 透明度。而对于非动画的GIF建议你使用PNG8，因为体积会更小~

**思路：**

[一个最简单也最保险的方法让IE6支持PNG图片透明（小小的分享一下）](http://www.w3cfuns.com/thread-2435-1-1.html)

**优缺点：**

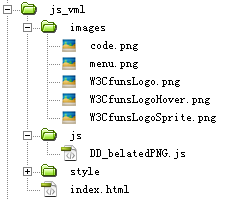
[点击此处前往](http://www.w3cfuns.com/forum.php?mod=redirect&goto=findpost&ptid=2435&pid=12084&fromuid=2)

**方案7 - DD\_belatedPNG解决方案：**

**介绍：**

我们都知道在目前所用的png图片透明解决方案基本都是使用滤镜、xpression解决的、透明gif替代。但是这些方法都有一个缺点,就是不支持CSS中backgrond-position与background-repeat。而这次的js插件使用了微软的VML语言进行绘制且不需要引入其他文件，一个小小的js就可以完美解决png图片bug就连img标签和hover伪类也可以很好的解决。

**目录说明：**



**思路：**

1、首先下载此方案所用到的文件 [DD\_belatedPNG.js](http://files.cnblogs.com/yjzhu/DD_belatedPNG.js)

2、引入刚下载的js文件，同样由于此js只有使用IE6时才有用，所以为了让我们的页面更加高效的执行，我们可以将上方代码修改如下，只有IE6的时候才调用执行此JavaScript：

<!--[if IE 6]><script type="text/javascript" src="js/DD\_belatedPNG.js"></script><![endif]-->

3、调用函数，设置参数如下：

DD\_belatedPNG.fix("#pngImg,#pics,#picsRepeat");

其中传入的参数为所使用png图片的标签的ID、类样式和标签名称，同样也可以按照下方这样来写：

DD\_belatedPNG.fix("#content img");

此方法则表示#content下的所有img标签透明

如果为链接和链接的hover设置透明，那么您按照下方这么来写，在部分版本里面可以不用加入:hover直接写选择器即可，但是为了保险，建议咱们还是加上:hover：

DD\_belatedPNG.fix("#links,#link:hover");

写到这里并且您使用过jQuery或者CSSQuery类库，那么您一定熟悉上面的这种选择方法，总之就是，在CSS中您是如何选择的元素，那么在这个js函数（方法）中传入什么，只不过多个选择的时候，使用逗号隔开即可。

**KwooShung用此方法时的小技巧：**如果页面中存在很多png，DD\_belatedPNG.fix();函数的参数岂不是很长？我们可以使用这种写法：

DD\_belatedPNG.fix(".pngFix,.pngFix:hover");

如果使用上述的写法，我们的html中只需要在相对应的标签上加入class="pngFix"就行了，如果有多个类样式，按照平时的多个类样式的写法即可class="abc cbc pngFix"，

使用此方法的时候，我们每次都要加载两个js文件或者写两个<script>标签才行，这样不太好，http请求会增多，那么我们可以打开DD\_belatedPNG.js文件，在尾部加入如下代码即可：

window.onload = function(){

DD\_belatedPNG.fix(".pngFix,.pngFix:hover");

}

这样我们只需要引入此JS，在需要透明的标签上加入class="pngFix"即可，简单 · 方便 · 快捷！

**优点：**

1. CSS代码无需任何修改，按照平时的思路来写即可；
2. 无需配置；
3. 没有多余的gif图片；
4. 支持img；
5. 支持平铺；
6. 支持CSS Sprite；
7. 支持Hover等伪类；

**缺点：**

1. 额外加入了js文件（6.39k）和http请求，可以忽略不计；
2. 当文件载入之前，会先暂时呈现灰底；
3. js文件过多的时候，可能会报错，导致js无法正常运行（这种情况极少出现，可以忽略不计）；

**使用情况：**

1. 当前6种方法均不能解决问题的时候可考虑；
2. 当png图片过多的时候可考虑，因为png图片太多，使用前面的几个方法，有的会导致CSS代码冗余过多，还不如引入此文件划算；

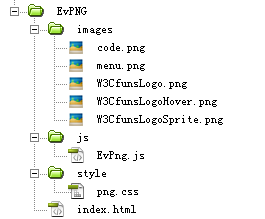
[DD\_belatedPNG解决方案 - DEMO入口](http://www.w3cfuns.com/demos/forum/2011-10/IE6_PNG/DD_belatedPNG/)

**方案8 - EvPng解决方案：**

**介绍：**

此方案与第七种方案差不多，使用方法也如出一辙，效果也非常不错。

**目录说明：**



**思路：**

1、首先下载此方案所用到的文件 [EvPng.zip](http://www.w3cfuns.com/forum.php?mod=attachment&aid=NTc1NnxjZGE0NzJiMnwxMzUyNDI0NDIzfDB8Mjk3)

2、参考第七种方案的使用方法。

**优点：**

1. CSS代码无需任何修改，按照平时的思路来写即可；
2. 无需配置；
3. 没有多余的gif图片；
4. 支持img；
5. 支持平铺；
6. 支持CSS Sprite；
7. 支持Hover等伪类；

**缺点：**

1. 额外加入了js文件（文件4.93k，比DD\_belatedPNG的6.39k还小）和http请求，可以忽略不计；
2. 当文件载入之前，会先暂时呈现灰底；
3. js文件过多的时候，可能会报错，导致js无法正常运行（这种情况极少出现，可以忽略不计）；
4. 使用CSS Sprite技术的hover效果在部分情况下top可能会有1像素的偏差。

**使用情况：**

1. 当前7种方法均不能解决问题的时候可考虑；
2. 当DD\_belatedPNG效果不理想的时候可以考虑；