

# CSS Sprite VS SVG Sprite

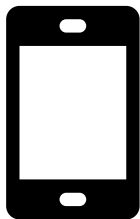
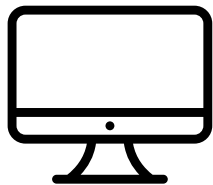
+Jack Pu



2014

我们不得不面对的问题？

不同的设备  
不同的分辨率



# SVG

Scalable Vector Graphics , SVG是基于可扩展标记语言 ( XML ) , 用于描述 二维矢量图形 的一种图形格式。

## SVG Sprite

- 方案一 类似 使用Adobe illustrator进行拼合，使用css background-position
- 方案二 使用symbol + use

## 使用AI生成SVG Sprite (DEMO1)

- 打开AI，新建一个30 x 60(px)的画布，设置好网格和参考线
- 用AI打开svg文件，然后复制路径到画布上调整大小
- 其他就和css-sprite没有差异了(注意SVG的大小)

## 使用PS+AI 生成SVG Sprite (DEMO2)

- 打开Photoshop 新建一个30 x 60 (px)的画布，我们计划30px \* 90px,设置好网格或者参考线
- 用AI打开SVG文件，然后Ctrl+C 复制路径，然后复制到photoshop文档中，选择图层形状，然后再进行调整
- 或者直接将是 图形形状的图层拖过来
- 选择 '文件' -> '导出' -> '路径到illustrator' -> 保存为 SVG

## 使用symbol + use 实现SVG Sprite(推荐)

- SVG `<symbol>`在svg中主要适用于定义可复用的符号，而这些定义在symbol元素的形状将不会被展示出来，而是通过use元素引用来显示
- 在SVG中`<use>`可以在任何地方复用svg文件中定义的形，包括`<g>`和 `<symbol>`以及`<defs>` 在使用 use 时，它必须要有一个id,这样 use 通过xlink:href的值找到该形状的引用.注意，一定要在前面加一个#，这样才能引用ID成功



## DEMO3

```
<svg style="display:none;" width="300" height="300">  
  <symbol id="wifi" viewBox="0 0 768 768">  
    <path d="M223.488 509.376l45.312 ..."></path>  
  </symbol>  
</svg>
```

在HTML中顶部引入SVG文件代码【注意隐藏】

# DEMO3

直接使用 use 引入

```
<svg class="svg-icon icon-demo">  
  <use xlink:href="#wifi"/>  
</svg>
```



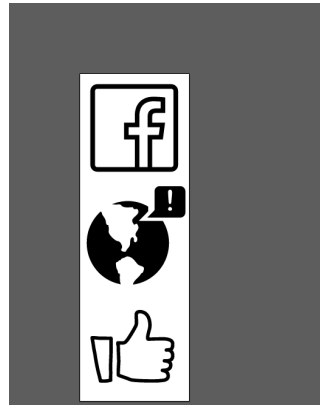
## 如何制作SVG Sprite

- 使用PS+AI
- 使用grunt svgstore 生成SVG Sprite
- 使用第三方工具诸如icomoon

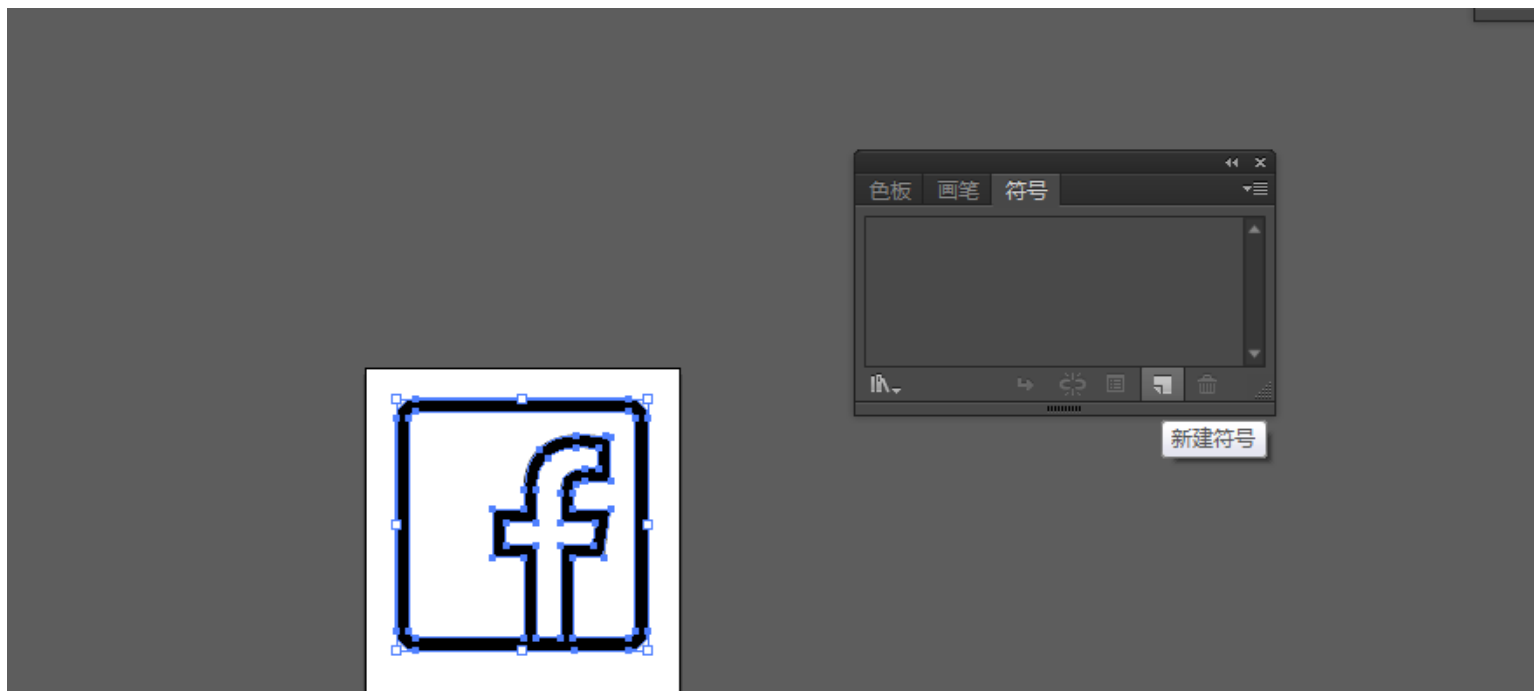
使用PS+AI生成SVG Sprite

前面步骤与生成背景的图类似，主要用于图形的拼合

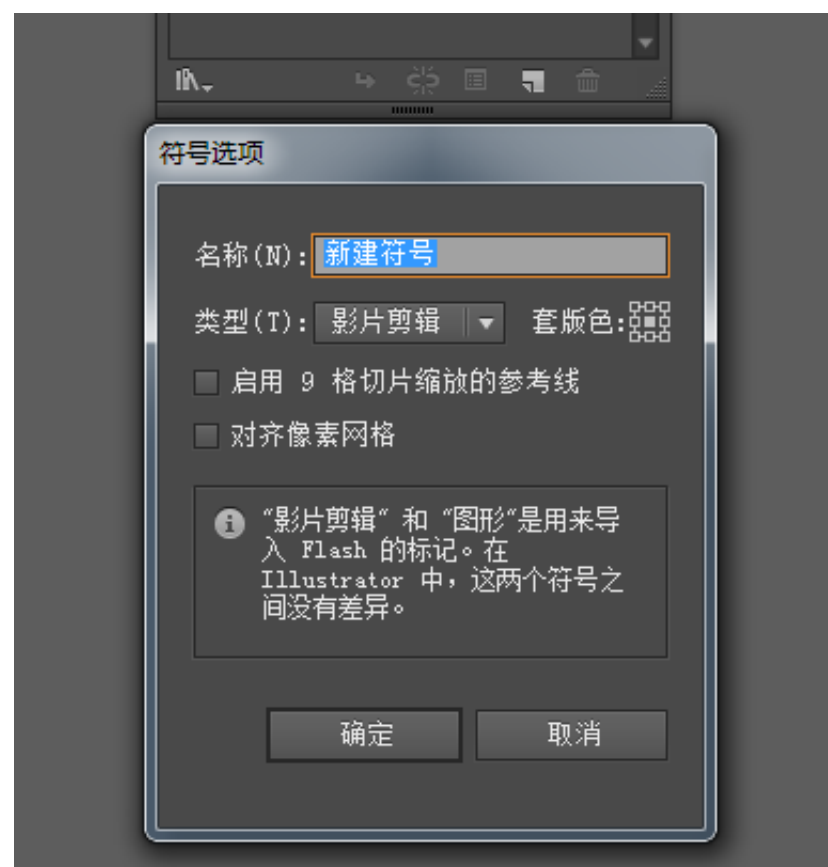
打开Symbols面板，在Window菜单栏中，或Shift+Ctrl+F11启用。



然后，选中单个元素,点击添加



给符号命名，该名称即为引用的ID





保存为SVG时，实际上生成的SVG代码并非我们要使用的，太过臃肿，可以到[这里](#) 进行处理下，方便使用

这里粘贴完整SVG代码：

```
9.945-8.777-9.955
c0.001,0,0.172-0.01,0.388-0.019c0.217-0.009,0.476-0.019,0.666-
0.019c0.051,0,0.097,0.001,0.14,0.002l0,0.001

c0.259,0.011,0.568,0.027,0.856,0.071l0.001,0c0.156,0.025,0.295,0.063,0.418,0.093c0.122,0.029,0.229,0.046,
0.288,0.045

c0.015,0,0.026-0.001,0.035-0.002C-5.837-9.8-5.691-9.744-5.592-
9.631c0.098,0.113,0.134,0.265,0.097,0.41
C-5.513-9.15-5.519-9.09-5.519-
9.004c0,0.085,0.007,0.197,0.007,0.338c0,0.024,0,0.049,0,0.074w0.001
c-0.002,0.078-0.003,0.157-0.003,0.235c0,0.37,0.02,0.753,0.02,1.152c0,0.142-0.003,0.285-
0.009,0.43

c-0.017,0.34-0.023,0.685-
0.023,1.033c0,1.435,0.115,2.929,0.116,4.384c0,0.973,0.006,1.573,0.006,2.549
c0,0.343,0.001,0.599,0.002,0.769c0,0.085,0.002,0.148,0.002,0.19c0,0.042,0,0.063,0,0.063C-
5.399,2.34-5.45,2.463-5.542,2.55
C-5.633,2.636-5.759,2.681-5.885,2.672z M-6.866-9.027l-6.866-9.027c-0.222-0.035-0.509-
0.051-0.759-0.062l0
c-0.022-0.002-0.057-0.002-0.1-0.002c-0.16,0-0.416,0.009-0.626,0.018c-0.013,0.001-
0.025,0.001-0.037,0.002
c-0.07,0.309-0.178,0.777-0.275,1.161l0,0C-9.072-6.308-9.382-4.98-9.735-2.646C-9.826-2.04-
9.904-1.668-9.977-1.313
```

转换萌萌哒

☒ 去掉填充色(可CSS控制图形颜色)

```
0.861-0.01756.979,2.49,6.621,2.49c-0.016,0-0.031,0-0.046,0c-0.016,0-0.03,0-0.045,0 c-0.373,0-0.603-0.013-
0.9-0.026C5.388,2.454,5.096,2.443,4.624,2.438c0.003,0.063,0.009,0.132,0.019,0.2
C4.654,2.731,4.67,2.821,4.684,2.896c0.04,0.781,3.443,4.97,4.101,5.12,4.667C5.208,5.003,5.255,5.411,5.256,5.8
5 c0,0.602-0.087,1.273-0.306,1.608c-0.11,0.317-0.255,0.626-
0.443,0.909C4.318,8.961,4.084,9.219,3.799,9.428
C3.582,9.583,3.31,9.715,3.004,9.819C2.698,9.821,2.363,9.993,2.024,9.993c-0.158,0-0.317-0.016-0.475-0.057
c-0.118-0.031-0.235-0.076-0.347-0.14c-0.167-0.096-0.32-0.24-0.425-
0.422C0.67,9.192,0.612,8.979,0.599,8.751 c-0.007-0.118-0.01-0.234-0.01-0.35c0.001-0.59,0.072-1.166,0.071-
1.691c0-0.19-0.009-0.374-0.032-0.548 c-0.014-0.11-0.056-0.238-0.122-0.38c-0.065-0.142-0.152-0.296-0.238-
0.46C0.19,5.176,0.034,4.695-0.163,4.551 C-0.46,4.037-0.853,3.38-1.227,2.806c-0.187-0.287-0.37-0.554-
0.531-0.789c-0.081-0.108-0.156-0.203-0.222-0.28 C-2.045,1.68-2.103,1.622-2.14,1.588l0,0c-0.063-0.055-
0.113-0.093-0.154-0.119C-2.358,1.431-2.399,1.413-2.47,1.397 C-2.54,1.382-2.642,1.374-2.793,1.374c-0.25,0-
0.623,0.024-1.166,0.041c-0.25,0.007-0.458-0.188-0.465-0.437 c-0.001-0.035,0.002-0.07,0.009-0.103C-
```

使用示意：

这里显示效果

这里显示代码




```
<svg><use xlink:href="#earth"></use></svg>
<svg><use xlink:href="#facebook"></use></svg>
<svg><use xlink:href="#like"></use></svg>
```

## 使用 svgstore 生成SVG Sprite

svgstore 是 grunt 的一个插件，用于自动获取文件中的SVG文件并自动合并。

1.新建一个项目文件夹如：demo4

2.在项目文件夹新建一个文件夹如:src，将需要合并的SVG 文件放进去

帮助(H)				
▼	刻录	新建文件夹		
名称	修改日期	类型	大小	
 facebook.svg	2015/2/2 11:54	Adobe Illustrato...	1 KB	
 instagram.svg	2015/2/2 12:02	Adobe Illustrato...	2 KB	
 twitter.svg	2015/2/2 11:54	Adobe Illustrato...	2 KB	

### 3.新建package.json

```
{  
  "name": "svg-store",  
  "version": "0.1.0",  
  "private": true,  
  "devDependencies": {  
    "grunt": "^0.4.5",  
    "grunt-svgstore": "~0.5.0"  
  }  
}
```

#### 4.然后新建Gruntfile.js,里面配置选项大致如下：

```
module.exports = function(grunt) {  
  // 配置  
  grunt.initConfig({  
    svgstore: {  
      options: {  
        prefix : 'icon-', // This will prefix each ID  
        svg: { // will add and override the the default xmlns="http://www.w3.org/20  
          viewBox : '0 0 100 100',  
          xmlns: 'http://www.w3.org/2000/svg'  
        },  
        includedemo:true,  
      },  
      default : {  
        files: {  
          'dest/dest-svg.svg': ['src/*.svg'],  
        },  
      }  
    }  
  });  
  // 载入grunt-svgstore  
  grunt.loadNpmTasks('grunt-svgstore');  
  // 注册任务  
  grunt.registerTask('default', ['svgstore']);  
};
```

5.输入npm install

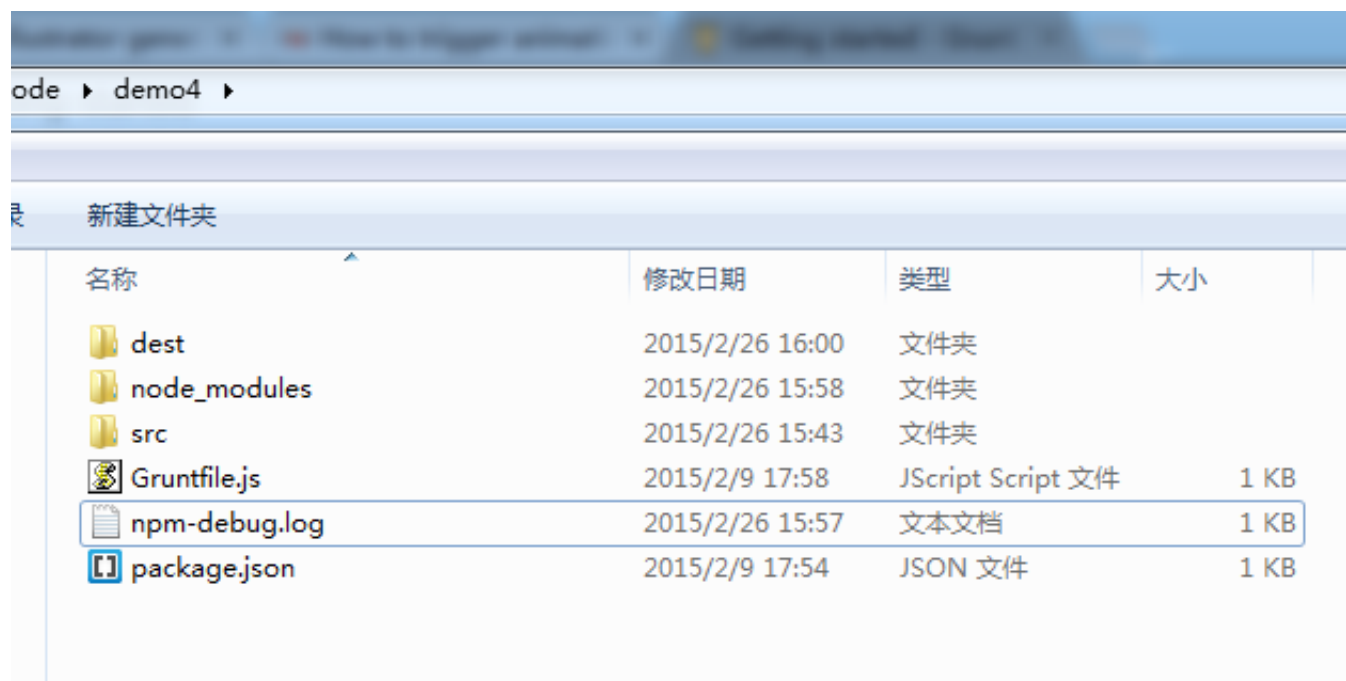
## 6.输入 grunt

```
E:\node\demo4>grunt
Running "svgstore:default" <svgstore> task
File dest/dest-svg.svg created.
Demo file E:\node\demo4\dest\dest-svg-demo.html created.

Done, without errors.
E:\node\demo4>
```



生成的文件夹中含有dest文件夹，里面包含了合并的svg文件以及可以使用的HTML文件

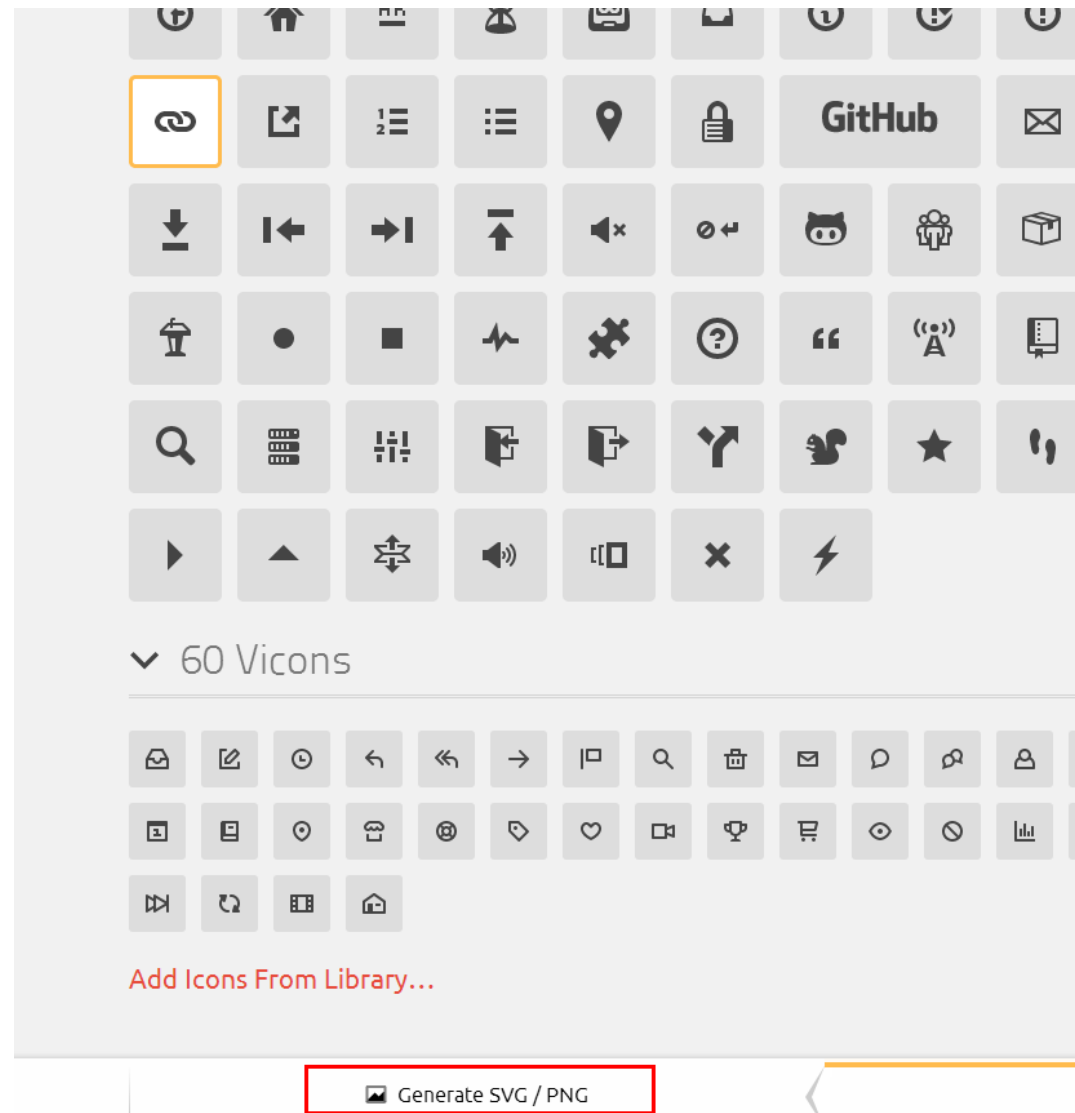


The screenshot shows a Windows File Explorer window with the address bar displaying 'ode > demo4 >'. The main area shows a list of files and folders. A toolbar at the top includes a '新建文件夹' (New Folder) button. The file list has columns for '名称' (Name), '修改日期' (Modified Date), '类型' (Type), and '大小' (Size). The files listed are 'dest' (folder), 'node\_modules' (folder), 'src' (folder), 'Gruntfile.js' (JScript Script file, 1 KB), 'npm-debug.log' (text document, 1 KB), and 'package.json' (JSON file, 1 KB). The 'npm-debug.log' file is currently selected.








名称	修改日期	类型	大小
dest	2015/2/26 16:00	文件夹	
node_modules	2015/2/26 15:58	文件夹	
src	2015/2/26 15:43	文件夹	
Gruntfile.js	2015/2/9 17:58	JScript Script 文件	1 KB
npm-debug.log	2015/2/26 15:57	文本文档	1 KB
package.json	2015/2/9 17:54	JSON 文件	1 KB

使用 icomoon 制作SVG Sprite

icommon不仅可以生成icon fonts还可以生成SVG Sprite



文件下载完成解压可以得到的文件夹里面会有demo.html 可以直接打开源码参考使用

 demo-files	2015/2/10 17:45
 PNG	2015/2/10 17:45
 SVG	2015/2/10 17:45
 demo.html	2015/2/10 17:45
 Read Me.txt	2015/2/10 17:45
 style.css	2015/2/10 17:45
 svgdefs.svg	2015/2/10 17:45

通过CSS去设置填充(fill)或者描边的颜色(stroke)

```
.icon-blue{  
    fill:#1ba1e2;  
}  
...  
<svg class="icon-blue ..">...</svg>
```



# 兼容性(IE8+)

- JS

```
if (!Modernizr.svg) {  
    ...  
    $("img[src$='.svg']") .attr("src", fallback);  
}
```

- CSS

```
.my-element {  
    background-image: url(image.svg);  
}  
.no-svg .my-element {  
    background-image: url(image.png);  
}
```

- inline svg

```
<svg>  
<foreignObject>  
    <span class="icon-not-svg icon-facebook"></span>  
</foreignObject>  
</svg>
```

# 使用SVG实现icon动画

## 扩展阅读

[SVG教程](#)

[SVG Sprite](#)

[Animated SVG Icons](#)

[How SVG Fragment Identifiers Work](#)

[How SVG Shape Morphing Works](#)

[Inline SVG vs Icon Fonts](#)

[icomoon](#)

[thenounproject](#)

[grunt-svgstore](#)

[A Compendium of SVG Information](#)



# Thanks!





**Powered By nodePPT v0.9.8-5**