## 一．SVG简介与实例

1. SVG简介

（1）SVG 是使用 XML 来描述二维图形和绘图程序的语言（ Version 1.1）

（2）什么是SVG？

SVG指的是可伸缩矢量图形（scalable vector graphics）

SVG使用 XML 格式定义图形

SVG图像在放大或改变尺寸的情况下，图形质量不会损失

SVG图像与JPG，GIF图像比起来，尺寸更小，可压缩性更强

SVG 于 2003 年 1 月 14 日成为 W3C 推荐标准。

（3）支持SVG的浏览器：

Internet Explorer9+，火狐，谷歌Chrome，Opera和Safari都支持SVG。

IE8和早期版本都需要一个插件 - 如Adobe SVG浏览器，这是免费提供的。

Internet Explorer 9、Firefox、Opera、Chrome 以及 Safari 支持内联 SVG。

1. SVG实例

|  |
| --- |
| ❶<?xml version="1.0" standalone="no"?>  ❷<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"  ❸"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">  <svg xmlns="http://www.w3.org/2000/svg" version="1.1">  <circle cx="50" cy="50" r="40" stroke="black" stroke-width="2" fill="red" />  </svg>  version 表示xml版本  standalone="no" 表示svg文档会引用一个外部文件，在这里是dtd文件  第二行，第三行引用了这个外部的SVG，DTD文件。该DTD位于"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd"  该DTD位于W3C，含有所有允许的SVG 元素 |

SVG 代码以 <svg> 元素开始，包括开启标签 <svg> 和关闭标签 </svg> 。

version 属性可定义所使用的 SVG 版本

xmlns 属性可定义 SVG 命名空间。（命名空间是唯一的URL），表示<svg>标记及其子标记属于名称空间为“http://www.w3.org/2000/svg”的XML方言

命名空间的出现就是为了区分相同名字的元素他们有不同的含义。

通过命名空间就能区分究竟是你的还是我的元素

必需引入命名空间，不然浏览器将此元素当做普通html标签插入文档流

所有的开启标签必须有关闭标签！关闭标签的作用是关闭 SVG 元素和文档本身。

* SVG的元素和属性必须按标准格式书写，因为XML是区分大小写的（这一点和html不同）
* SVG里的属性值必须用引号引起来，就算是数值也必须这样做。

属性version必不可少的，供其它类型的验证方式确定SVG版本

## 二．SVG 常用属性

最常用的属性有三个：fill，stroke，stroke-width

* 1. fill:设置图形内部填充颜色
  2. fill-rule属性的取值可以是nonzero | evenodd | inherit
  3. stroke:设置边缘线条颜色
  4. stroke-width:设置边缘线条的厚度
  5. opacity：设置图形的不透明度
  6. fill-opacity:填充色的不透明度
  7. stroke-opacity：控制描边的不透明度
  8. stroke-linecap：控制边框终点的形状，属性值：butt直线结束线段，square ，round终点是圆角
  9. stroke-linejion：用来控制两条线段之间用什么方式连接，属性值：miter(默认)，round(圆角连接)，bevel（连接处形成斜接）
  10. stroke-dasharray：使描边为虚线，属性值：一组用逗号分割的数列
  11. 伪类元素在svg中不能正常使用，如before，after
  12. 除了sing一对象的属性，还可以用CSS来样式化填充和描边，但是注意不是所有的属性都能用CSS来设置
  13. 可用CSS设置的有：fill，stroke等；不可用CSS设置的有：渐变，滤镜，图案等，
  14. 利用<style>可以设置一段样式段落。就像在html里这样的<style>一般放在<head>里，在svg里<style>则放在<defs>标签里。

<defs>表示定义，这里面可以定义一些不会在SVG图形中出现、但是可以被其他元素使用的元素。

<svg width:" 200" 
<defs 
#MyRect { 
stroke : 
height= " 200" xmlns="http://www.w3.org/2000/svg' 
< style ! [CDATA[ 
black; 
fill: red; 
</defs 
(rect x: " 10" height=' 
< / svg> 
'180' 
'10" 
width:" 
180" 
"MyRect " / > 

## 三．SVG元素

* 1. Internet Explorer和Safari不支持SVG滤镜！
  2. 形状元素：
     1. 矩形（包括可选的圆角），使用<rect>元素创建：x，y（矩形的起始位置），width，height（矩形宽高）
     2. 圆形，使用<circle>元素创建：cx，cy：圆点的x，y坐标。省略cx和cy，圆的中心会被设置为(0, 0)，r属性定义圆的半径
     3. 椭圆形，使用<ellipse>元素创建：cx：椭圆中心的x坐标;cy:椭圆中心的y坐标;rx:定义水平半径;ry:定义垂直半径
     4. 直线，使用<line>元素创建:x1,y1(起点位置)，x2,y2(结束位置)
     5. （1）折线，使用<polyline>元素创建：points

（2）多边形，使用<polygon>元素创建：points

属性points 定义多边形每个角的x和y坐标，多个角坐标用空格区分，一个角的x和y坐标用，隔开

f. path元素的形状是通过属性d定义的，属性d的值是一个“命令+参数”的序列，一个点集数列以及其它关于如何绘制路径的信息；

M：Move TO ，M命令经常出现在路径的开始处，用来指明从何处开始画。只会移动画笔，但不会在两点之间画线

真正画线的有三个命令，最常用的是 Line to命令，另外两个是H，绘制平行线。V，绘制垂直线。

L： Line to，两个参数x轴和y轴坐标

Z：Z命令会从当前点画一条直线到路径的起点（不区分大小写）

以上所有命令均允许小写字母。大写表示绝对定位，小写表示相对定位。

path是最常用的命令，理论上，这些基础图形元素都能用<path>元素来构建的相同的图形。并且所有可用于<path>元素的属性都可以应用在这些基础图形上。

z 
V 
c 
s 
Q 
A 
horizontal 
vertical 
Bézjer 
smooth 
quadratic dézier curveto 
smooth quadratic aézier 
elliptical arc($E3J7) 
none 
(xl yl x2y2xy)+ 
(x2 Y2 x 
(xl yl xy)+ 
(rx ry x-axis-rotation large-arc-flag sweep-flag x y)+ 

* 1. 文本元素
     1. 文本：text：x,y 文本位置
     2. 路径上的文字：

<defs>

<path id="path1" d="M75,20 a1,1 0 0,0 100,0" />

</defs>

<text x="10" y="100" style="fill:red;">

<textPath xlink:href="#path1">I love SVG I love SVG</textPath>

</text>

c.与文本相关的其他元素：

tspan:标记大块文本的子部分，它必须是一个text元素或别的tspan元素的子元素。一个典型的用法是把句子中的一个词变成粗体红色。

<text>

<tspan font-weight="bold" fill="red">This is bold and red</tspan>

</text>

tspan有几个自定义属性：

x:容器新的坐标，覆盖当前文本坐标；dx：用一个水平偏移开始绘制文本

rotate：字符旋转；

textPath：该元素利用它的xlink:href属性取得一个任意路径，把字符对齐到路径，于是字体会环绕路径、顺着路径走：

<path 20,20 C 40,40 80,40 lee,2€" 
t ext > 
textPath 
>This text follows a curve. </textPath 
text 

* 1. 渐变元素

渐变有两种类型:线性渐变和径向渐变

必须给渐变内容一个id属性，这样别的元素才可以引用它；为了让渐变能够重复使用，渐变内容需要定义在<defs>标签内，而不是定义在形状上边

* 1. 线性渐变：

线性渐变沿着直线改变颜色，要插入一个线性渐变，你需要在SVG文件的defs元素内部，创建一个<linearGradient> 节点。

x1，x2，y1，y2属性定义渐变开始和结束位置,

每种颜色通过一个<stop>标签来规定

offset属性用来定义渐变的开始和结束位置。

填充属性fill把标签元素（例如rect）链接到此渐变

当y1和y2相等，而x1和x2不同时，可创建水平渐变

当x1和x2相等，而y1和y2不同时，可创建垂直渐变

当x1和x2不同，且y1和y2不同时，可创建角形渐变

* 1. 径向渐变：

径向渐变与线性渐变相似，只是它是从一个点开始发散绘制渐变。创建径向渐变需要在文档的defs中添加一个<radialGradient>元素

cx,cy和r属性定义的最外层圆和Fx和Fy定义的最内层圆

每种颜色通过一个<stop>标签来规定

offset属性用来定义渐变的开始和结束位置。

填充属性fill把标签元素（例如rect）链接到此渐变

* 1. SVG坐标元素
     1. viewBox：viewBox属性允许指定一个给定的一组图形伸展以适应特定的容器元素。

viewBox = <min-x> <min-y> <width> <height>

<min-x> 和 <min-y> 值决定viewBox的左上角，width和height决定视窗的宽高。这里要注意视窗的宽高不一定和父<svg>元素的宽高一样。<width>和<height>值为负数是不合法的。值为0的话会禁止元素的渲染。

* 1. defs元素
     1. 类别：容器元素，结构元素
     2. SVG允许我们定义以后需要重复使用的图形元素；建议把所有需要再次使用的引用元素定义在defs元素中，增加SVG内容的易读性和可访问性；
     3. 允许的内容物：
        1. 动画元素：<animate>, <animateColor>, <animateMotion>, <animateTransform>, <set>
        2. 形状元素：<circle>, <ellipse>, <line>, <path>, <polygon>, <polyline>, <rect>
        3. 渐变元素：<linearGradient>, <radialGradient>
        4. 描述性元素：<desc>, <metadata>, <title>
        5. 结构化元素：<defs>, <g>, <svg>, <symbol>, <use>。等

## 四．SVG在HTML中应用

SVG可以通过以下标签嵌入HTML文档：<embed>,<object>,<iframe>

SVG的代码可以直接嵌入到HTML文件中，或者可以直接链接到SVG文件

1. 使用<embed>标签
   1. embed标签被所有主流浏览器支持，并允许使用脚本；
   2. 在页面中嵌入SVG时使用<embed>标签，是 Adobe SVG Viewer 推荐的方法，

然而，要创建合法的XHTML，就不能使用<embed>。任何HTML规范中都没有<embed>标签

1. 使用<object>标签
   1. <object> 标签是 HTML 4 的标准标签，被所有较新的浏览器支持。
   2. 缺点是不允许使用脚本。
2. 使用<iframe>标签
   1. <iframe>标签可在大多数浏览器中使用，并允许使用脚本
   2. 缺点：不推荐在HTML4和XHTML中使用（但在HTML5允许）
3. 直接在HTML中嵌入SVG代码：
   1. Firefox、Internet Explorer9、谷歌Chrome和Safari中，可以直接在HTML嵌入SVG代码。

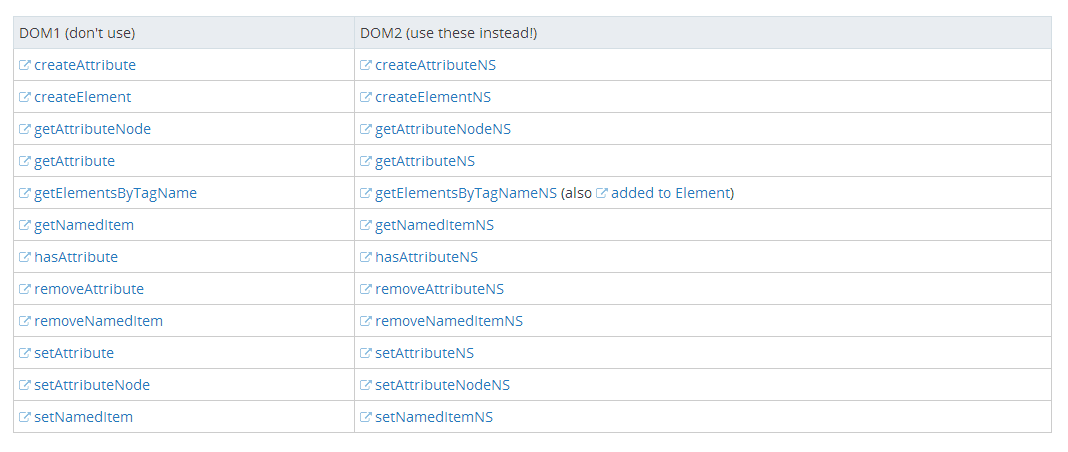
<svg xmlns="http://www.w3.org/2000/svg" version="1.1">

<circle cx="100" cy="50" r="40" stroke="black" stroke-width="2" fill="red" />

</svg>

## 五．JS动态创建SVG节点

在svg中应该使用的dom方法：



有效的命名空间的URI:

HTML - 使用 <http://www.w3.org/1999/xhtml>

SVG - 使用 <http://www.w3.org/2000/svg>

创建svg节点时，要使用createElementNS函数并传入节点名称的命名空间。否则创建出来的节点默认为html dom而不是svg dom。用document.createElement创建出来的节点是属于html dom，而svg的节点是svg dom。所以需要用createElementNS函数并传入节点名称的命名空间。

JS创建svg元素节点过程：

（1）使用document对象创建一个svg元素rect，为元素引入命名空间（必需引入命名空间，不然浏览器将此元素当做普通html标签插入文档流）

var rectObj = document.createElementNS("http://www.w3.org/2000/svg","rect");

(2) 使用标签元素对象的setAttribute函数为元素设置属性

rectObj.setAttribute("width",200)；

rectObj.setAttribute("height",100);

rectObj.setAttribute("fill","pink");

(3)有两种方式将svg子元素插入到svg父元素中：

* 1. 利用document.getElementById获取id，再结合appendChild（）方法将子元素插入到父元素中
     1. let svg = document.getElementById("svgID");
     2. svg.appendChild(rectObj);
  2. 利用jQuery提供的append()方法将svg子元素插入到svg父元素中(推荐使用)
     1. $("#svgID").append(rectObj);

比较推荐第二种方法，可以直接用jq方法直接在svg父元素上添加多个节点，书写比较简洁

## 六．SVG动画

1.animate：动画元素放在形状元素内部，用来定义一个元素的某个属性如何踩着时间点改变，在指定持续时间里，属性从开始值变为结束值

专有属性：

attributeName（要被改变的属性名），

attributeType：该属性指定目标属性和它相对应的值处于哪个命名空间里，

可取值：

CSS：指定attributeName的值是一个CSS属性名，

XML：指定attributeName的值是一个XML属性名，在目标元素的默认XML命名空间里。

auto:先搜索CSS属性列表找不到，再为这个元素搜索默认XML命名空间。

from：取值和attributeName属性有关，若attributeName=“width”，from = 100，则表示动画从100 宽度开始

to：动画修改的属性的最终值

by:动画的相对变化值。

value：用分号分隔的一个或多个值，可以看出是动画的多个关键值点。

---------------------------------------------------------------------------------------------------------------------------------from; to ， 坷 0 values 虽 然 属 于 一 族 但 是 相 互 之 间 还 是 有 制 约 关 系 的 。 有 以 下 一 此 规 则 
a ． 如 果 动 画 的 起 始 值 与 元 素 的 默 认 值 是 一 样 的 ， from 参 可 以 省 略 。 
b. 〔 不 考 虑 lues ） to ， by 两 个 参 数 至 少 要 有 一 个 出 现 。 否 则 动 画 效 果 没 有 。 to 表 示 绝 对 值 ， by 表 示 相 对 值 。 莖 位 移 距 离 ， 如 果 from 是 1 ， to 
值 为 1 则 表 示 移 动 到 1 汶 首 立 置 ， 但 是 ， 如 果 by 值 是 1 ， 则 表 示 移 动 到 9 + 1 ： 2 汶 个 亡 置 。 
c ． 如 果 to ： by 同 时 出 现 ， 则 by 打 酱 涠 ， 只 识 别 to 
d ． 如 果 to ： by 0 艹 s 都 没 设 置 自 然 没 动 画 效 果 。 如 果 任 意 （ 包 括 fr 、 om 
一 个 属 性 的 值 不 台 法 ， 规 范 上 说 是 没 有 动 画 效 果 。 但 是 ， 据 我 测 试 ， FireFoxÄlJ 览 
器 晦 实 如 此 ， 但 是 Chrome 特 意 做 了 写 容 处 理 。 例 如 ， 丕 来 是 数 值 的 属 性 ， 写 了 个 诸 如 a 汶 吓 台 法 的 值 ， 其 会 当 作 9 来 处 理 ， 动 画 效 果 依 然 存 在 
e. values 可 以 是 一 个 值 並 多 值 。 根 据 我 在 Chrome 浏 览 器 下 的 测 试 ， 是 一 个 值 的 时 候 是 没 有 动 画 效 果 。 多 值 时 候 有 动 画 效 果 。 当 values 值 设 置 并 能 识 别 时 候 ， 
from; to ， by 的 值 都 会 被 忽 略 。 那 values 属 性 是 干 什 么 的 呢 ？ 另 刂 看 名 字 挺 大 众 的 ， 其 还 是 有 些 功 力 的 。 我 们 实 现 动 画 ， 不 可 能 就 是 单 纯 的 从 a 位 置 到 b 位 置 
有 时 候 ， 要 去 c 位 置 过 哣 下 。 此 时 ， 实 际 上 有 3 个 动 画 关 踺 。 而 from ， to / by 只 能 竺 驭 两 个 ， 此 时 就 是 1 艹 s 大 显 身 手 的 时 候 了 ！ 

---------------------------------------------------------------------------------------------------------------------------------

dur：动画持续时间

repeatCount：动画发生次数，属性值为infinite，表示循环一直播放

fill:freeze | remove. 其中remove是默认值，表示动画结束直接回到开始的地方。freeze“冻结”表示动画结束后像是被冻住了，元素保持了动画结束之后的状态。

animate是可以叠加的，可以同时写几个animate

1. animateMotion：引导元素延指定运动路径移动

<animateMotion> 元素的mpath子元素使<animateMotion> 元素能够引用一个外部的<path>元素作为运动路径的定义。

<mpath xlink:href="#path1"/> path指的是指定路径的id

1. animateTransform：

animateTransform元素变动了目标元素上的一个变形属性，从而允许动画控制转换、缩放、旋转或斜切。