

SVG 参考手册

SVG 元素

元素	说明	属性
<a>	创建一个SVG元素周围链接	xlink:show xlink:actuate xlink:href target
<altGlyph>	允许对象性文字进行控制，来呈现特殊的字符数据	x y dx dy rotate glyphRef format xlink:href
<altGlyphDef>	定义一系列象性符号的替换	id
<altGlyphItem>	定义一系列候选的象性符号的替换	id
<animate>	随时间动态改变属性	attributeName="目标属性名称" from="起始值" to="结束值" dur="持续时间" repeatCount="动画时间将发生"
<animateColor>	定义随着时间的推移颜色转换	by="相对偏移值" from="起始值" to="结束值"
<animateMotion>	使元素沿着动作路径移动	calcMode="动画的插补模式。可以是'discrete', 'linear', 'paced', 'spline'" path="运动路径" keyPoints="沿运动路径的对象目前时间应移动多远"

		<p>rotate="应用旋转变换"</p> <p>xlink:href="一个URI引用<path>元素，它定义运动路径"</p>
<animateTransform>	动画上一个目标元素变换属性，从而使动画控制平移，缩放，旋转或倾斜	<p>by="相对偏移值"</p> <p>from="起始值"</p> <p>to="结束值"</p> <p>type="类型的转换其值是随时间变化。可以是 'translate', 'scale', 'rotate', 'skewX', 'skewY'"</p>
<circle>	定义一个圆	<p>cx="圆的x轴坐标"</p> <p>cy="圆的y轴坐标"</p> <p>r="圆的半径". 必需.</p> <p>+ 显现属性：颜色，FillStroke，图形</p>
<clipPath>	用于隐藏位于剪切路径以外的对象部分。定义绘制什么和什么不绘制的模具被称为剪切路径	<p>clip-path="引用剪贴路径和引用剪贴路径交叉"</p> <p>clipPathUnits="userSpaceOnUse'或'objectBoundingBox"。第二个值children一个对象的边框，会使用掩码的一小部分单位（默认："userSpaceOnUse"）"</p>
<color-profile>	指定颜色配置文件的说明（使用CSS样式文件时）	<p>local="本地存储颜色配置文件唯一ID"</p> <p>name=""</p> <p>rendering-intent="auto perceptual relative-colorimetric saturation absolute-colorimetric"</p> <p>xlink:href="ICC配置文件资源URI"</p>
<cursor>	定义一个独立于平台的自定义光标	<p>x="x轴左上角光标（默认为0）"</p> <p>y="y轴的左上角光标（默认为0）"</p> <p>xlink:href="使用光标图像URI"</p>
<defs>	引用的元素容器	
<desc>	对 SVG 中的元素的纯文本描述 - 并不作为图形的一部分来显示。用户代理会将其显示为工具提示	
<ellipse>	定义一个椭圆	<p>cx="椭圆x轴坐标"</p> <p>cy="椭圆y轴坐标"</p> <p>rx="沿x轴椭圆形的半径". 必需.</p> <p>ry="沿y轴长椭圆形的半径". 必需.</p>

		+ 显现属性：颜色，FillStroke，图形
<feBlend>	使用不同的混合模式把两个对象合成在一起	mode="图像混合模式： normal multiply screen darken lighten" in="标识为给定的滤镜原始输入：SourceGraphic SourceAlpha BackgroundImage BackgroundAlpha FillPaint StrokePaint <filter-primitive-reference>" in2="第二输入图像的混合操作"
feColorMatrix	SVG滤镜。适用矩阵转换	
feComponentTransfer	SVG 滤镜。执行数据的 component-wise 重映射	
feComposite	SVG 滤镜	
feConvolveMatrix	SVG 滤镜	
feDiffuseLighting	SVG 滤镜	
feDisplacementMap	SVG 滤镜	
feDistantLight	SVG滤镜。定义一个光源	
feFlood	SVG滤镜	
feFuncA	SVG 滤镜。feComponentTransfer 的子元素	
feFuncB	SVG 滤镜。feComponentTransfer 的子元素	
feFuncG	SVG 滤镜。feComponentTransfer 的子元素	
feFuncR	SVG 滤镜。feComponentTransfer 的子元素	
feGaussianBlur	SVG滤镜。执行高斯模糊图像	
feImage	SVG滤镜。	
feMerge	SVG滤镜。建立在彼此顶部图像层	

feMergeNode	SVG 滤镜。feMerge的子元素	
feMorphology	SVG 滤镜。对源图形执行"fattening" 或者 "thinning"	
feOffset	SVG滤镜。相对其当前位置移动图像	
fePointLight	SVG滤镜	
feSpecularLighting	SVG滤镜	
feSpotLight	SVG滤镜	
feTile	SVG滤镜	
feTurbulence	SVG滤镜	
filter	滤镜效果的容器	
font	定义字体	
font-face	描述一种字体的特点	
font-face-format		
font-face-name		
font-face-src		
font-face-uri		
foreignObject		
<g>	用于把相关元素进行组合的容器元素	id="该组的名称" fill="该组填充颜色" opacity="该组不透明度" + 显现属性: All
glyph	为给定的象形符号定义图形	
glyphRef	定义要使用的可能的象形符号	

hkern		
<image>	定义图像	<p>x="图像的左上角的x轴坐标"</p> <p>y="图像的左上角的y轴坐标"</p> <p>width="图像的宽度". 必须.</p> <p>height="图像的高度". 必须.</p> <p>xlink:href="图像的路径". 必须.</p> <p>+ 显现属性:</p> <p>Color, Graphics, Images, Viewports</p>
<line>	定义一条线	<p>x1="直线起始点x坐标"</p> <p>y1="直线起始点y坐标"</p> <p>x2="直线终点x坐标"</p> <p>y2="直线终点y坐标"</p> <p>+ 显现属性:</p> <p>Color, FillStroke, Graphics, Markers</p>
<linearGradient>	定义线性渐变。通过使用矢量线性渐变填充对象，并可以定义为水平，垂直或角渐变。	<p>id="id 属性可为渐变定义一个唯一的名称。引用必须"</p> <p>gradientUnits=""userSpaceOnUse' or 'objectBoundingBox'.使用视图框或对象，以确定相对位置矢量点。（默认为'objectBoundingBox'）"</p> <p>gradientTransform="适用于渐变的转变"</p> <p>x1="渐变向量x启动点（默认0%）"</p> <p>y1="渐变向量y启动点（默认0%）"</p> <p>x2="渐变向量x的终点。（默认100%）"</p> <p>y2="渐变向量y的终点。（默认0%）"</p> <p>spreadMethod=""pad' or 'reflect' or 'repeat'"</p> <p>xlink:href="reference to another gradient whose attribute values are used as defaults and stops included. Recursive"</p>
<marker>	标记可以放在直线，折线，多边形和路径的顶点。这些元素可以使用marker属性的"marker-start"，"marker-mid"和"marker-end"，继承默认情况下或可设置为"none"或定义的标记的URI。您必须先定义标记，然后才可以通过其	<p>markerUnits="strokeWidth'或'userSpaceOnUse"。如果是strokeWidth"那么使用一个单位等于一个笔划宽度。否则，标记尺度不会使用同一视图单位作为引用元素（默认为'strokeWidth'）"</p> <p>refx="标记顶点连接的位置（默认为0）"</p> <p>refy="标记顶点连接的位置（默认为0）"</p> <p>orient=""auto'始终显示标记的角度。"auto"将计算某个角度使</p>

	URI引用。任何一种形状，可以把标记放在里面。他们绘制元素时把它们附加到顶部	<p>得X轴一个顶点的正切值（默认为0）</p> <p>markerWidth="标记的宽度（默认3）"</p> <p>markerHeight="标记的高度（默认3）"</p> <p>viewBox="各点"看到"这个SVG绘图区域。由空格或逗号分隔的4个值。（min x, min y, width, height）"</p> <p>+ presentation attributes:</p> <p>All</p>
<mask>	度屏蔽是一种不透明度值的组合和裁剪。像裁剪，您可以使用图形，文字或路径定义掩码的部分。一个掩码的默认状态是完全透明的，也就是裁剪平面的对面的。在掩码的图形设置掩码的不透明部分	<p>maskUnits=""userSpaceOnUse' or 'objectBoundingBox'.设定裁剪面是否是相对完整的视窗或对象（默认：'objectBoundingBox'）"</p> <p>maskContentUnits="第二个掩码相对对象的图形位置使用百分比'userSpaceOnUse'或'objectBoundingBox'（默认：'userSpaceOnUse'）"</p> <p>x="裁剪面掩码（默认值：-10%）"</p> <p>y="裁剪面掩码（默认值：-10%）"</p> <p>width="裁剪面掩码（默认是：120%）"</p> <p>height="裁剪面掩码（默认是：120%）"</p>
metadata	指定元数据	
missing-glyph		
mpath		
<path>	定义一个路径	<p>d="定义路径指令"</p> <p>pathLength="如果存在，路径将进行缩放，以便计算各点相当于此值的路径长度"</p> <p>transform="转换列表"</p> <p>+ 显现属性:</p> <p>Color, FillStroke, Graphics, Markers</p>
<pattern>	定义坐标，你想要的视图显示和视图的大小。然后添加到您的模式形状。该模式命中时重复视图框的边缘（可视范围）	<p>id="用于引用这个模式的唯一ID。"必需的。</p> <p>patternUnits=""userSpaceOnUse'或'objectBoundingBox".第二个值X, Y, width, height 一个会使用模式对象的边框的小部分，单位（%）。"</p> <p>patternContentUnits=""userSpaceOnUse'或'objectBoundingBox'"</p>

		<p>patternTransform="允许整个表达式进行转换"</p> <p>x="模式的偏移量，来自左上角（默认为0）"</p> <p>y="模式的偏移量，来自左上角（默认为0）"</p> <p>width="模式平铺的宽度（默认为100%）"</p> <p>height="模式平铺的高度（默认为100%）"</p> <p>viewBox="各点"看到"这个SVG绘图区域。由空格或逗号分隔的4个值。（min x, min y, width, height）"</p> <p>xlink:href="另一种模式，其属性值是默认值以及任何子类可以继承。递归"</p>
<polygon>	定义一个包含至少三边图形	<p>points="多边形的点。点的总数必须是偶数"。必需的。</p> <p>fill-rule="FillStroke演示属性的部分"</p> <p>+ 显现属性:</p> <p>Color, FillStroke, Graphics, Markers</p>
<polyline>	定义只有直线组成的任意形状	<p>points=折线上的"点"。必需的。</p> <p>+ 显现属性:</p> <p>Color, FillStroke, Graphics, Markers</p>
<radialGradient>	定义放射性渐变。放射性渐变创建一个圆圈	<p>gradientUnits=""userSpaceOnUse' or 'objectBoundingBox'.</p> <p>使用视图框或对象以确定相对位置的矢量点。（默认为'objectBoundingBox'）"</p> <p>gradientTransform="适用于渐变的变换"</p> <p>cx="渐变的中心点（数字或% - 50%是默认）"</p> <p>cy="渐变的中心点。（默认50%）"</p> <p>r="渐变的半径。（默认50%）"</p> <p>fx="渐变的焦点。（默认0%）"</p> <p>fy="渐变的焦点。（默认0%）"</p> <p>spreadMethod=""pad' or 'reflect' or 'repeat'"</p> <p>xlink:href="引用到另一个渐变，其属性值作为默认值。递归"</p>
<rect>	定义一个矩形	<p>x="矩形的左上角的x轴"</p> <p>y="矩形的左上角的y轴"</p> <p>rx="x轴的半径（round元素）"</p> <p>ry="y轴的半径（round元素）"</p> <p>width="矩形的宽度"。必需的。</p>

		<p>height="矩形的高度"。必需的。</p> <p>+ 显现属性: Color, FillStroke, Graphics</p>
script	脚本容器。（例如ECMAScript）	
set	设置一个属性值指定时间	
<stop>	渐变停止	<p>offset="偏移停止（0到1/0%到100%）". 参考</p> <p>stop-color="这个stop的颜色"</p> <p>stop-opacity="这个Stop的不透明度（0到1）"</p>
style	可使样式表直接嵌入SVG内容内部	
<svg>	创建一个SVG文档片段	<p>x="左上角嵌入（默认为0）"</p> <p>y="左上角嵌入（默认为0）"</p> <p>width="SVG片段的宽度（默认为100%）"</p> <p>height="SVG片段的高度（默认为100%）"</p> <p>viewBox="点"seen"这个SVG绘图区域。由空格或逗号分隔的4个值。（min x, min y, width, height）"</p> <p>preserveAspectRatio=""none'或任何'xVALYVAL'的9种组合,VAL是"min", "mid"或"max".（默认情况下none）"</p> <p>zoomAndPan=""magnify' or 'disable'.Magnify选项允许用户平移和缩放您的文件（默认Magnify）"</p> <p>xml="最外层<svg>元素都需要安装SVG和它的命名空间：</p> <p>xmlns="http://www.w3.org/2000/svg"</p> <p>xmlns:xlink="http://www.w3.org/1999/xlink"</p> <p>xml:space="preserve""</p> <p>+ 显现属性: All</p>
switch		
symbol		
<text>	定义一个文本	<p>x="列表的X -轴的位置。在文本中在第n个字符的位置在第n个x轴。如果后面存在额外的字符，耗尽他们最后一个字符之后放置的位置。0是默认"</p> <p>y="列表的Y轴位置。（参考x）0是默认"</p>

		<p>dx="在字符的长度列表中移动相对最后绘制标志符号的绝对位置。（参考x）"</p> <p>dy="在字符的长度列表中移动相对最后绘制标志符号的绝对位置。（参考x）"</p> <p>rotate="一个旋转的列表。第n个旋转是第n个字符。附加字符没有给出最后的旋转值"</p> <p>textLength="SVG查看器将尝试显示文本之间的间距/或字形调整的文本目标长度。（默认：正常文本的长度）"</p> <p>lengthAdjust="告诉查看器，如果指定长度就尝试进行调整用以呈现文本。这两个值是'spacing'和'spacingAndGlyphs'"</p> <p>+ 显现属性:</p> <p>Color, FillStroke, Graphics, FontSpecification, TextContentElements</p>
textPath		
title	对 SVG 中的元素的纯文本描述 - 并不作为图形的一部分来显示。用户代理会将其显示为工具提示	
<tref>	引用任何SVG文档中的<text>元素和重用	相同的<TEXT>元素
<tspan>	元素等同于<text>，但可以在内部嵌套文本标记以及内部本身	Identical to the <text> element + in addition: xlink:href="引用一个<TEXT>元素"
<use>	使用URI引用一个<G>,<svg>或其他具有一个唯一的ID属性和重复的图形元素。复制的是原始的元素，因此文件中的原始存在只是一个参考。原始影响到所有副本的任何改变。	x="克隆元素的左上角的x轴" y="克隆元素的左上角的y轴" width="克隆元素的宽度" height="克隆元素的高度" xlink:href="URI引用克隆元素" + 显现属性: All
view		
vkern		

← SVG 实例

✎ 点我分享笔记