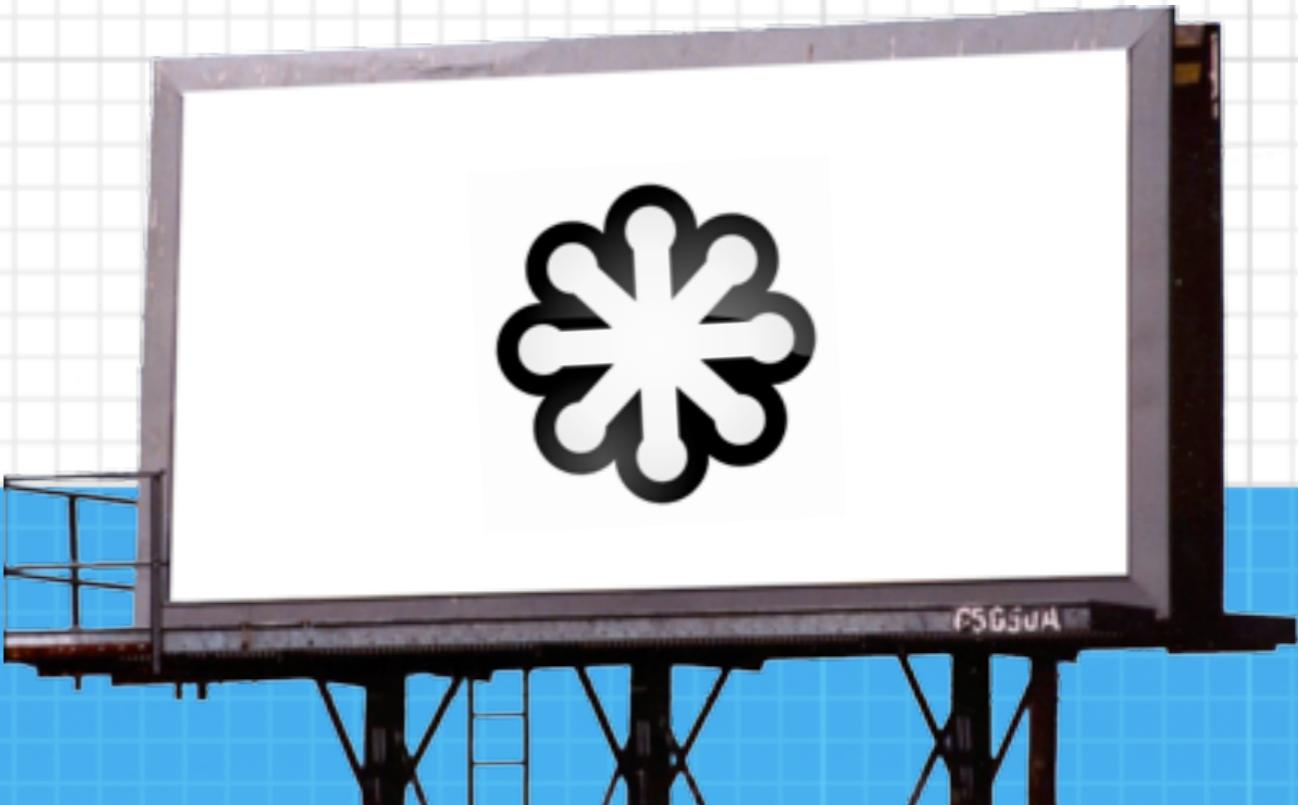




Whirlwind Tour of Scalable Vector Graphics

by Marc Grabanski



Agenda

- 1. What is SVG? Why?**
- 2. Overview SVG Elements**
- 3. Embed SVG**
- 4. SVG Features**
- 5. DEMOs!**
- 6. Raphael**
- 7. Other Cool Tools**

Who?

Marc Grabanski



jQuery UI Datepicker

MarcGrabanski.com

LOTS of UI Dev

Currently...

UI/UX Development Consultant



Publisher of:

**FRONTEND
MASTERS**

<http://FrontendMasters.com>

1. What is SVG?

SVG



HTML...
for graphics!

One SVG For All

Scaling bitmaps (jpeg/gif/png)
have to add more pixel data

Scaling SVG..use that same file

SVG
Zoomed



PNG
Zoomed



Modify



Static



why Use SVG?

SVG Advantages

- Scales and Retina-ready
- HTML-like elements - DOM Structure
- CSS3 and JavaScript
- Printer friendly
- Accessibility Features and SEO
- Browser Coverage
- Gzip-friendly
- More! Filters, Paths, Viewbox, etc.

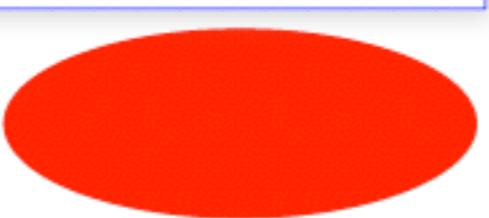
2. SVG Elements

Elements Walkthrough

`<text>`

Hello, out there

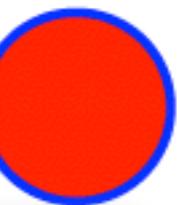
`<ellipse>`



`<rect>`

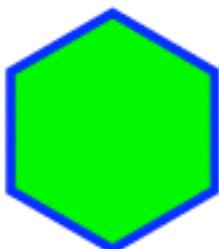


`<circle>`



`<line>`

`<polygon>`



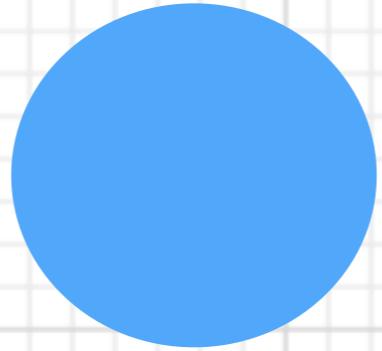
`<path>`

Hello, out there

```
<text x="250" y="150">
```

Hello, out there

```
</text>
```

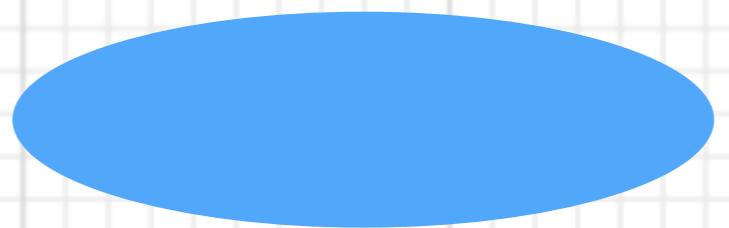


<circle cx=50 cy=50 r=200 fill="#56a9f6" />

center x, center y and radius

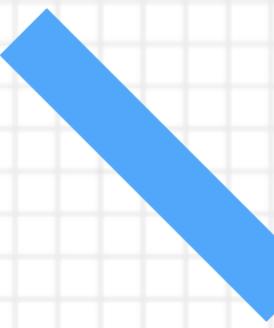


```
<rect x=50 y=50 width=200 height=100  
fill="#56a9f6" />
```



```
<ellipse rx="200" ry="100" fill="#56a9f6" />
```

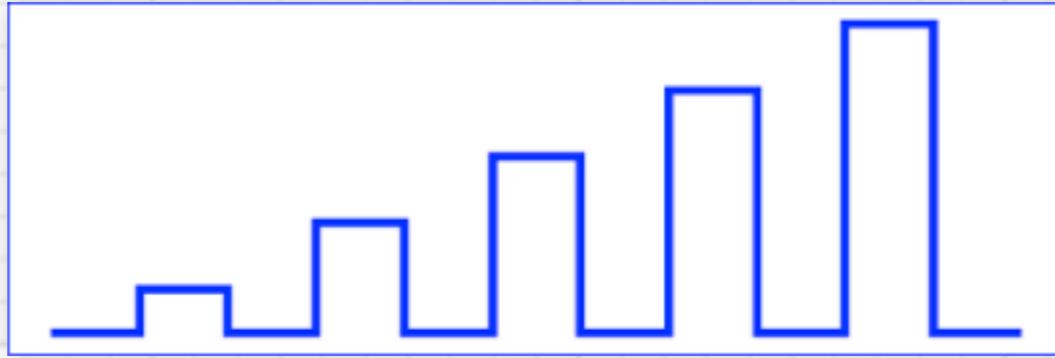
radius x, radius y



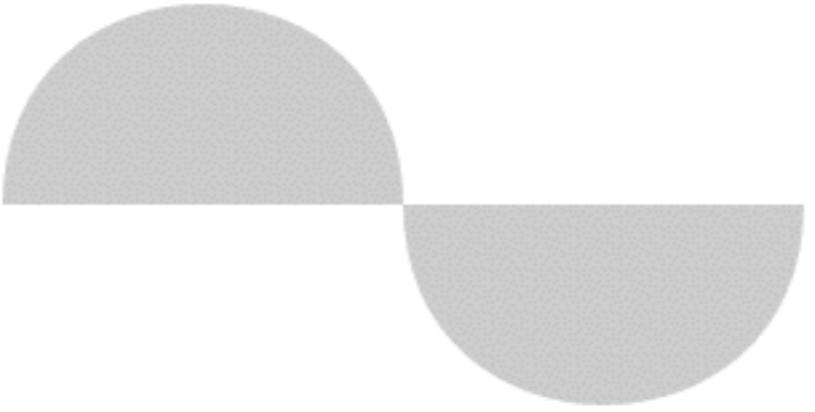
```
<line xl="0" yl="0" x2="100" y2="100"  
stroke="#56a9f6"  
stroke-width="25" />
```



```
<polygon fill="#56a9f6"  
points="350,75 379,161 469,161 397,215  
423,301 350,250 277,301 303,215  
231,161 321,161" />
```



```
<polyline fill="none"  
stroke="blue" stroke-width="10"  
points="50,375 150,375 150,325 250,325 250,375  
      350,375 350,250 450,250 450,375  
      550,375 550,175 650,175 650,375  
      750,375 750,100 850,100 850,375  
      950,375 950,25 1050,25 1050,375 1150,375' />
```



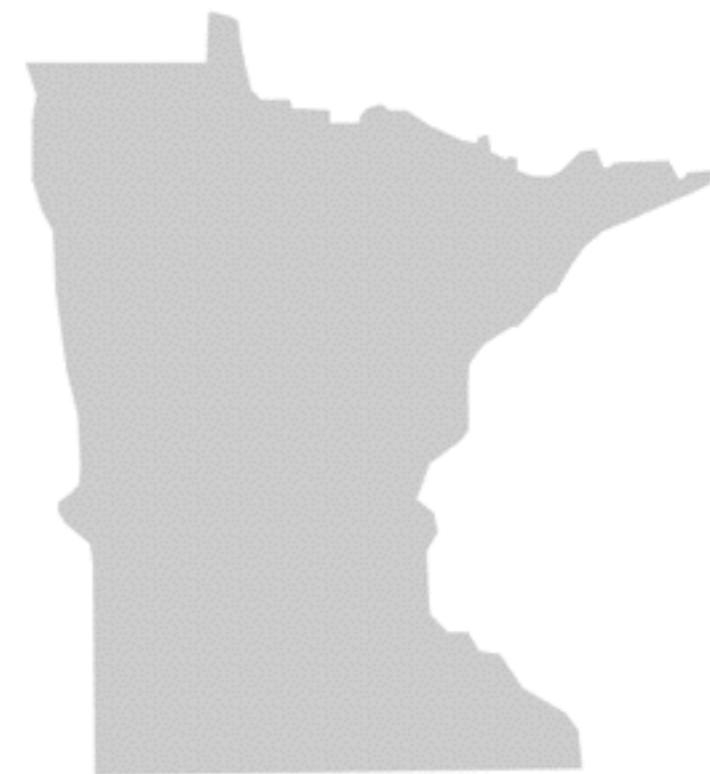
```
<path d="M100,200 C100,100 250,100 250,200  
S400,300 400,200" />
```

Command Driven:

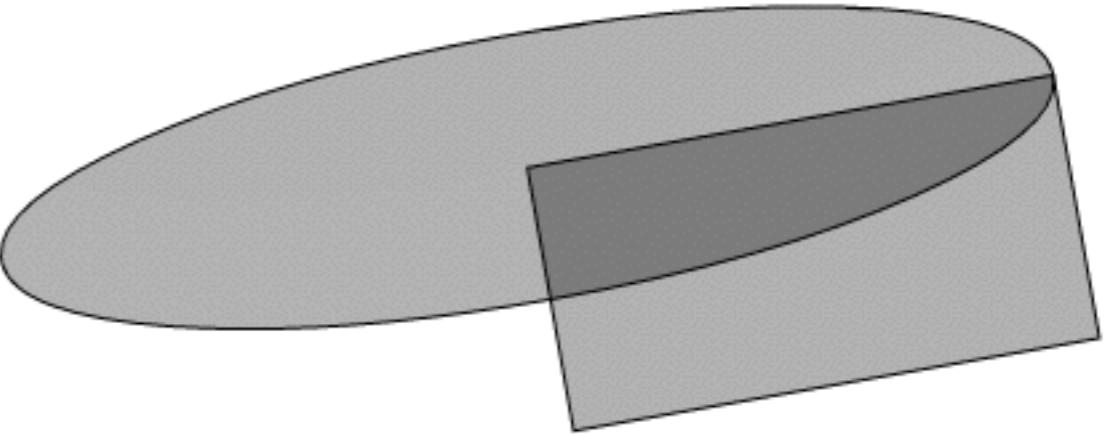
**Moveto, Line, Curveto, Bézier Curves,
Quadratic Bézier Curves, Elliptical Arc...**

**Path is Incredibly
Powerful**

<path d="M 475.23781,128.82439 L 474.78056,120.36535 L 472.95158,113.04943
L 471.1226,99.560705 L 470.66535,89.729927 L 468.83637,86.300584 L
467.23601,81.270889 L 467.23601,70.982869 L 467.92187,67.096282 L
466.10094,61.644615 L 496.23336,61.679886 L 496.55668,53.435202 L
497.20332,53.273541 L 499.46657,53.758523 L 501.40649,54.566825 L
502.21479,60.063281 L 503.66974,66.206379 L 505.28634,67.822984 L
510.13616,67.822984 L 510.45948,69.277928 L 516.76424,69.601249 L
516.76424,71.702835 L 521.61405,71.702835 L 521.93737,70.409551 L
523.06899,69.277928 L 525.33224,68.631286 L 526.62552,69.601249 L
529.53541,69.601249 L 533.41526,72.187816 L 538.75006,74.612723 L
541.17497,75.097705 L 541.65995,74.127742 L 543.11489,73.64276 L
543.59987,76.552649 L 546.18644,77.845933 L 546.67142,77.360951 L
547.96471,77.522612 L 547.96471,79.624198 L 550.55127,80.594161 L
553.62282,80.594161 L 555.23943,79.785858 L 558.47264,76.552649 L
561.0592,76.067668 L 561.86751,77.845933 L 562.35249,79.139216 L
563.32245,79.139216 L 564.29241,78.330914 L 573.18374,78.007593 L
574.962,81.079142 L 575.60865,81.079142 L 576.32226,79.994863 L
580.76217,79.624198 L 580.15007,81.903657 L 576.21135,83.740782 L
566.96557,87.80191 L 562.19083,89.808807 L 559.11928,92.395375 L
556.69437,95.951905 L 554.43113,99.831756 L 552.65286,100.64006 L
548.12637,105.65153 L 546.83308,105.81319 L 542.5053,108.57031 L
540.04242,111.77542 L 539.8138,114.96681 L 539.90816,123.01016 L
538.53212,124.69891 L 533.45058,128.45888 L 531.2205,134.44129 L
534.09225,136.675 L 534.77214,139.90198 L 532.9169,143.14091 L
533.08769,146.88893 L 533.45655,153.61933 L 536.4848,156.62132 L
539.8138,156.62132 L 541.70491,159.75392 L 545.08408,160.25719 L
548.94324,165.92866 L 556.03053,170.04541 L 558.17368,172.92053 L
558.84483,179.36004 L 477.63333,180.50483 L 477.29541,144.82798 L
476.83817,141.85589 L 472.72296,138.42655 L 471.57984,136.59757 L
471.57984,134.9972 L 473.63744,133.39685 L 475.00918,132.02511 L
475.23781,128.82439 z" fill="#ccc" />



Grouping



```
<g fill="rgba(0,0,0,0.3)"  
transform="rotate(-10)" stroke-width="5"  
stroke="black">  
  
  <rect x=200 y=200 width=200 height=100 />  
  
  <ellipse cx=200 cy=200 rx=200 ry=50 />  
  
</g>
```

3. Embed SVG

Embed Methods

- Object Tag
- Inline (HTML5)
- As Image
- CSS Background
- Image Fallback

SVG Object

```
<object type="image/svg+xml"  
data="test.svg">  
    
</object>
```

For JS scripting you need to grab object:
`element.contentDocument`

Inline SVG

```
<!doctype html>
```

```
<html><body>
```

```
<svg width=200 height=200>
```

```
  <polygon fill="#ccc" points="350,75 379,161  
    469,161 397,215 423,301 350,250 277,301 303,215  
    231,161 321,161" />
```

```
</svg>
```

```
</body></html>
```



Inline SVG Support

Inline SVG in HTML5 - Candidate Recommendation												Global user stats*:		
	IE	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Blackberry Browser	Opera Mobile	Chrome for Android	Firefox for Android	IE Mobile	
27 versions back			4.0											
26 versions back			5.0											
25 versions back		2.0	6.0											
24 versions back		3.0	7.0											
23 versions back		3.5	8.0											
22 versions back		3.6	9.0											
21 versions back		4.0	10.0											
20 versions back		5.0	11.0											
19 versions back		6.0	12.0											
18 versions back		7.0	13.0											
17 versions back		8.0	14.0											
16 versions back		9.0	15.0											
15 versions back		10.0	16.0											
14 versions back		11.0	17.0											
13 versions back		12.0	18.0		9.0									
12 versions back		13.0	19.0		9.5-9.6									
11 versions back		14.0	20.0		10.0-10.1									
10 versions back		15.0	21.0		10.5									
9 versions back		16.0	22.0		10.6									
8 versions back		17.0	23.0		11.0									
7 versions back		18.0	24.0	3.1	11.1									
6 versions back	5.5	19.0	25.0	3.2	11.5			2.1		10.0				
5 versions back	6.0	20.0	26.0	4.0	11.6	3.2		2.2		11.0				
4 versions back	7.0	21.0	27.0	5.0	12.0	4.0-4.1		2.3		11.1				
3 versions back	8.0	22.0	28.0	5.1	12.1	4.2-4.3		3.0		11.5				
2 versions back	9.0	23.0	29.0	6.0	15.0	5.0-5.1		4.0		12.0				
Previous version	10.0	24.0	30.0	6.1	16.0	6.0-6.1		4.1	7.0	12.1				
Current	11.0	25.0	31.0	7.0	17.0	7.0	5.0-7.0	4.2-4.3	10.0	16.0	31.0	25.0	10.0	

<http://caniuse.com/svg-html5>

Image SRC

```
<!doctype html>  
<html><body>  
  
</body></html>
```



JS inside SVG is disabled.

SVG as IMG SRC

# SVG in HTML img element - Candidate Recommendation										*Usage stats:		Global			
Method of displaying SVG images in HTML using 										Support:	82.92%				
										Partial support:	0.01%				
										Total:	82.93%				
Show all versions	IE	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Blackberry Browser	IE Mobile					
											2.1				
						3.2				2.2					
						4.0-4.1				2.3					
	8.0			5.1		4.2-4.3				3.0					
	9.0		29.0	6.0		5.0-5.1				4.0					
	10.0	24.0	30.0	6.1		6.0-6.1				4.1	7.0				
Current	11.0	25.0	31.0	7.0	17.0	7.0	5.0-7.0	4.2-4.3	10.0	10.0					

<http://caniuse.com/#feat=svg-img>

CSS Background

```
<div id="foo"></div>

<style>
#foo { background: url(goat.svg); }

</style>
```

SVG in CSS Background

# SVG in CSS backgrounds - Candidate Recommendation										*Usage stats:		Global			
Method of using SVG images as CSS backgrounds										Support:	76.97%				
										Partial support:	5.95%				
										Total:	82.92%				
Show all versions	IE	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Blackberry Browser	IE Mobile					
											2.1				
						3.2				2.2					
						4.0-4.1				2.3					
8.0				5.1		4.2-4.3				3.0					
9.0			29.0	6.0		5.0-5.1				4.0					
10.0	24.0	30.0	6.1			6.0-6.1				4.1	7.0				
Current	11.0	25.0	31.0	7.0	17.0	7.0	5.0-7.0	4.2-4.3	10.0	10.0					

<http://caniuse.com/#feat=svg-css>



SVGeneration

<http://www.svgeneration.com/>

Image Fallback

```
<svg>  
  <image  
    xlink:href="svg.svg" src="svg.png"  
    width="100%" height="100%" />  
</svg>
```

Image Fall Scripts

SVGMagic

The jQuery SVG fallback that does it all

SVGeezy

Before IE9

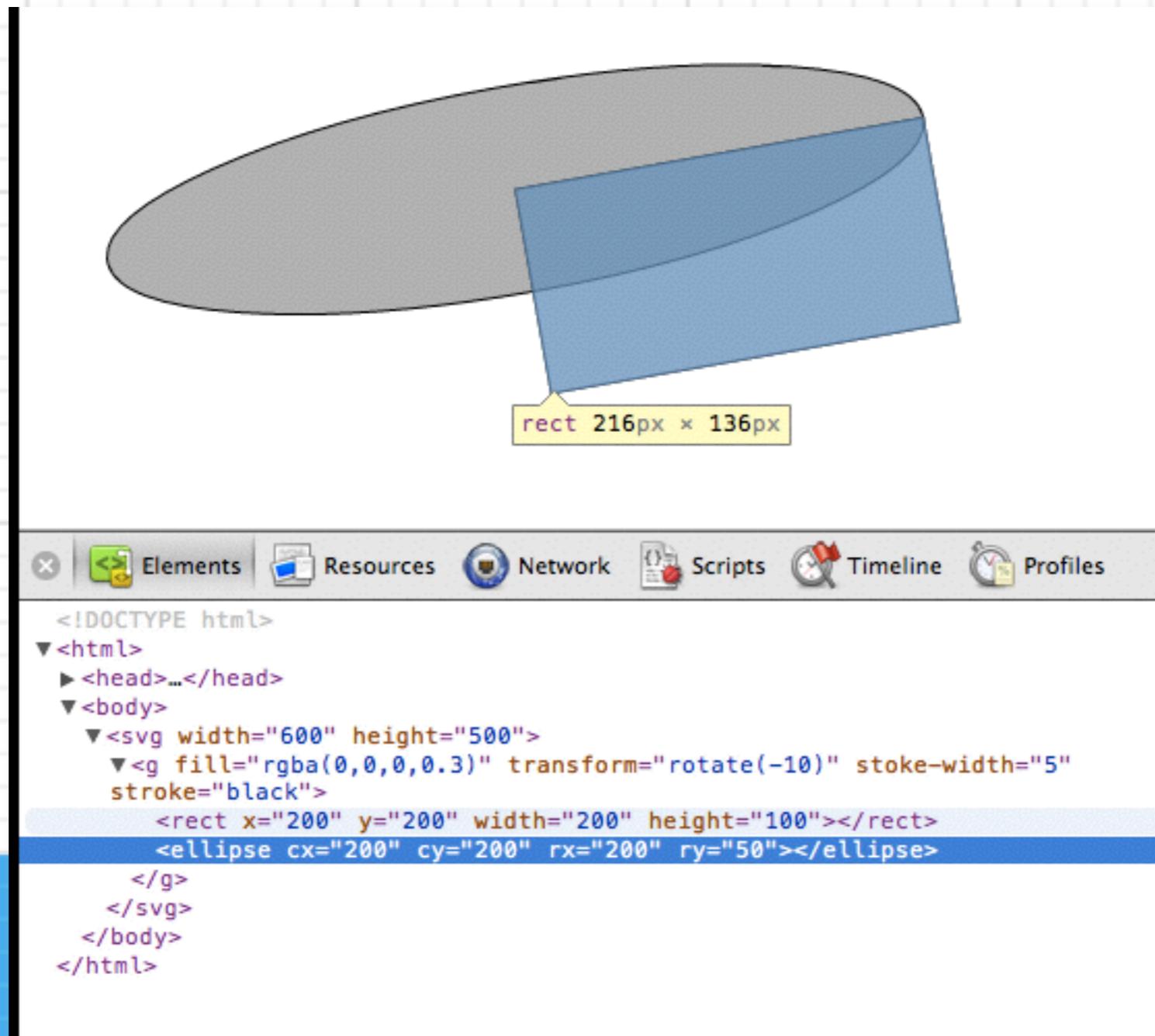


Raphael JS

4. SVG Features

DOM Structure

Inspectable Elements





document.getElementById('star').onclick = ...

```
<polygon id="star" fill="#ccc"  
points="350,75 379,161 469,161 397,215  
423,301 350,250 277,301 303,215  
231,161 321,161" />
```

SVG Links



```
<a xlink:href="http://minnesota.com">  
  <path d="..state of MN.." fill="#ccc" /></a>  
  
<a xlink:href="http://wisconsin.com">  
  <path d="..state of WI.." fill="#ccc" /></a>
```

CSS Animations



<http://www.anthonycalzadilla.com/i-twitty-the-fool/>

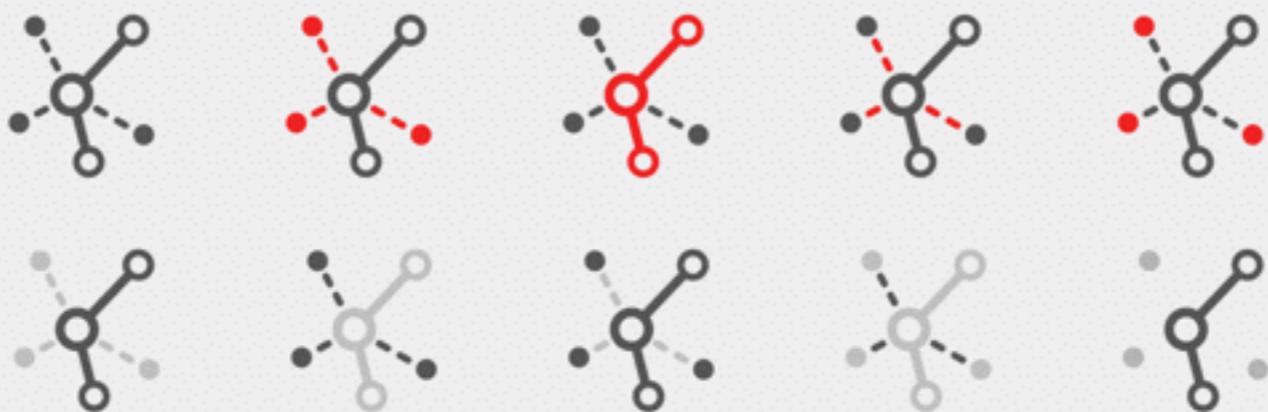
Great for Icons



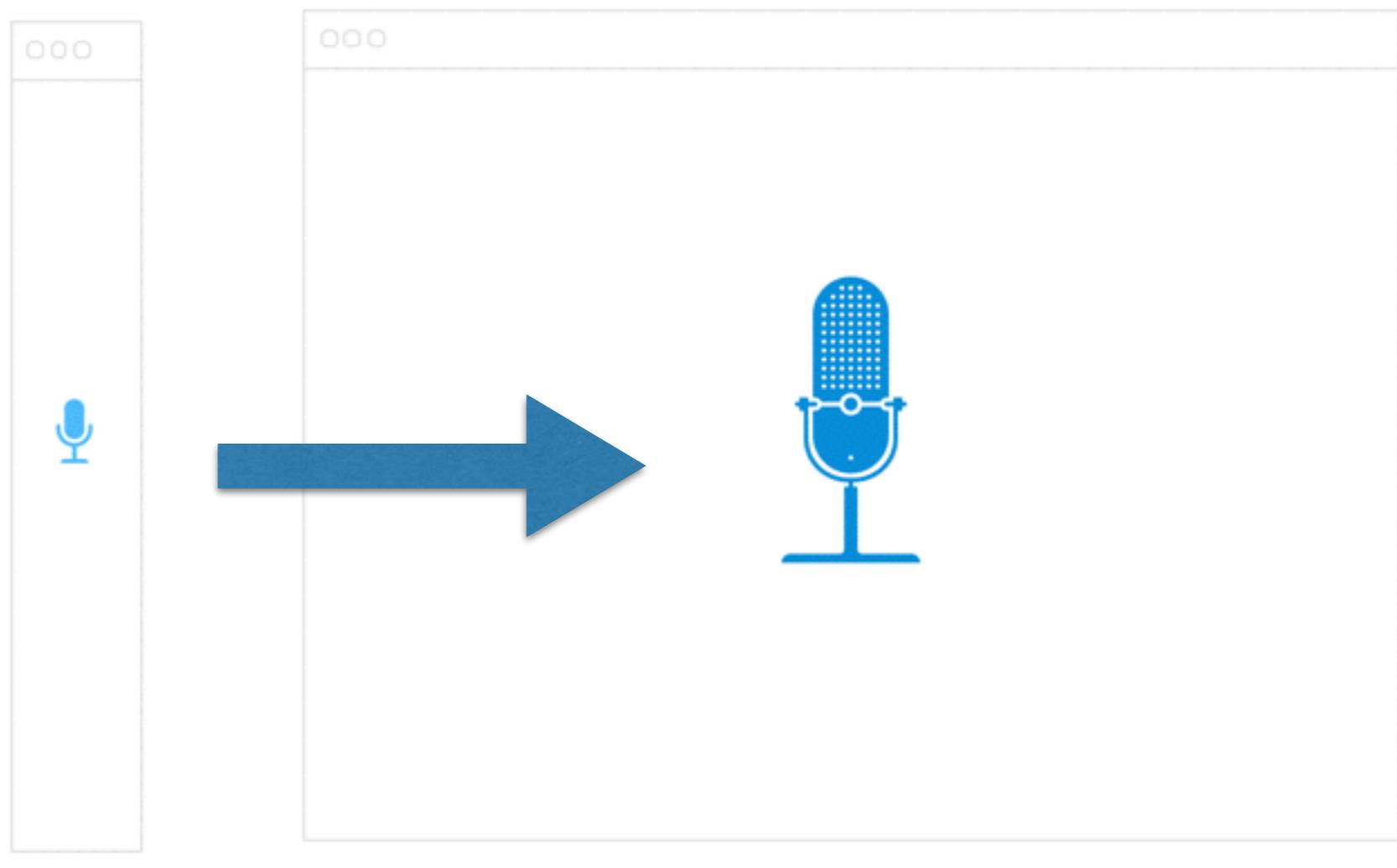
CSS + Icons

SVG Possibilities

1 icon, 10 different styles



Media Queries



Vector	Icon Font	SVG
CSS Control	yup (but anti-aliasing...)	yup
Positioning	some	better
Weird Failures	die XXX murder	good
Semantics	yup	less
Accessibility	none 	better <svg><desc>
Ease of use	be very careful	better, could use more info
Browser Support	ok	better
	IE 6+	IE 9+



SVG Filters



<http://svg-wow.org/filterEffects/chiseled.svg>

Hands-On Filters



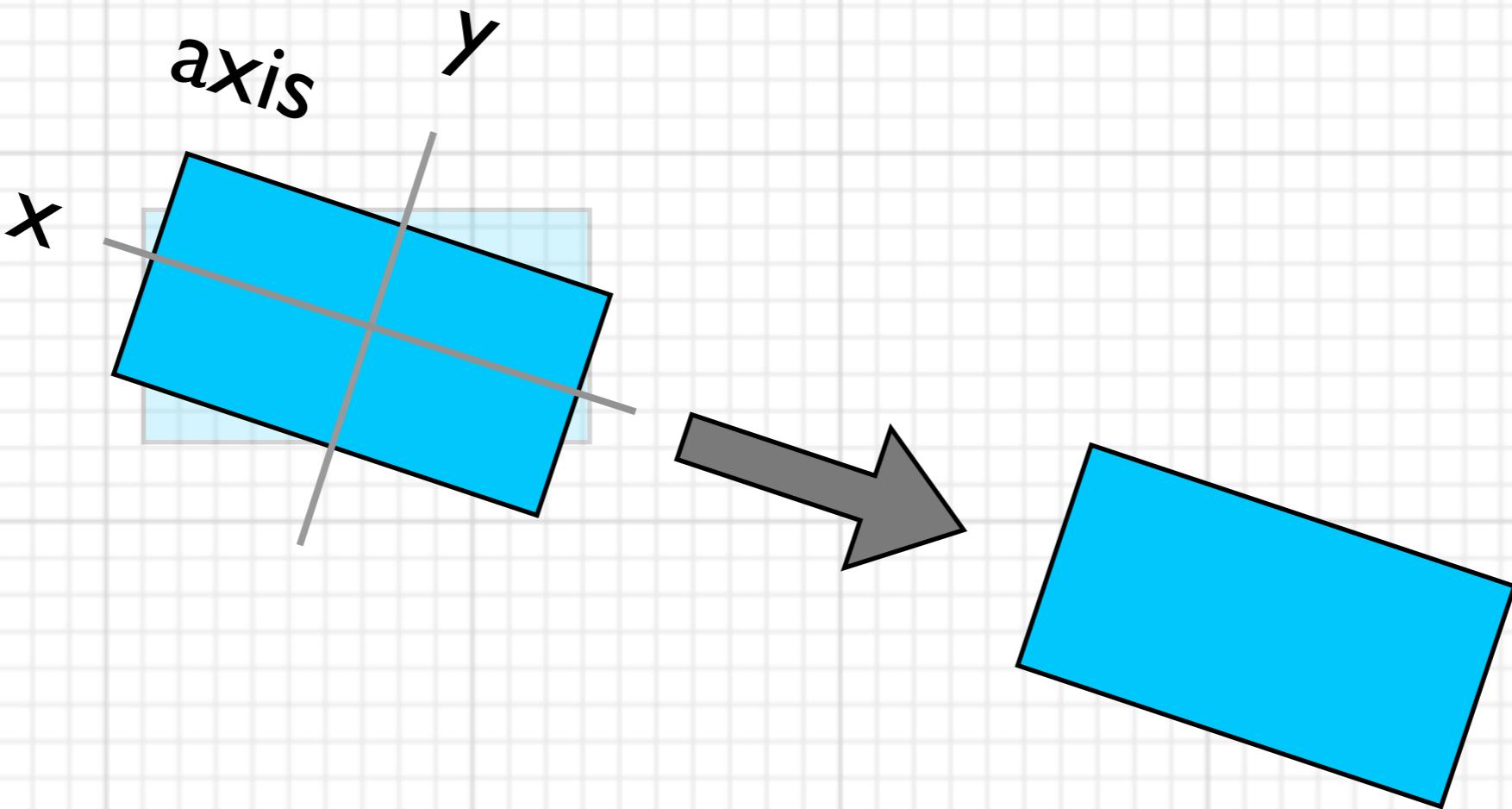
http://ie.microsoft.com/testdrive/graphics/hands-on-css3/hands-on_svg-filter-effects.htm

SVG Filter Support

# SVG filters - Recommendation										*Usage stats:		Global	
Method of using photoshop-like effects on SVG objects including blurring and color manipulation.										Support:		72.87%	
										Partial support:		0.08%	
										Total:		72.95%	
Show all versions	IE	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Blackberry Browser	IE Mobile			
											2.1		
						3.2				2.2			
						4.0-4.1				2.3			
8.0				5.1		4.2-4.3				3.0			
9.0			29.0	6.0		5.0-5.1				4.0			
10.0	24.0		30.0	6.1		6.0-6.1			4.1	7.0			
Current	11.0	25.0	31.0	7.0	17.0	7.0	5.0-7.0	4.2-4.3	10.0	10.0			

<http://caniuse.com/#feat=svg-filters>

Transforms



Demo: Stacking Transforms



The screenshot shows a CodePen interface with the following details:

- HTML:**

```
<form id="transforms">
  <p style="text-align:left">Try stacking transforms by clicking "apply transform" and then add another.</p>
  <select id="transforms-select">
    <option data-transform="translate">translate (x [y])</option>
    <option data-transform="scale">scale (x [y])</option>
    <option data-transform="rotate">rotate (a [x y])</option>
    <option data-transform="skewX">skewX (a)</option>
```
- CSS:**

```
svg {
  position: absolute;
  top:0;
  bottom:0;
  left:0;
  right:0;
  z-index: -1;
}
```
- JS:**

```
var $fields = $('.translate-field').hide(),
  $results = $("#results"),
  $applyTransforms = $("#apply-transforms"),
  selectedTransform,
  transforms = [],
  result = '';
$fields.first().show();
selectedTransform = $fields.first().data('transform');
```

The preview area displays a large orange 3D-style letter 'H' with a white center. To its right is a control panel with the word "HTML" in large bold letters. Below it is a text input field containing "Try stacking transforms by clicking "apply transform" and then add another." To the right of the input is a dropdown menu labeled "translate (x [y])". Below the input are two sliders labeled "x:" and "y:", each with a circular slider handle. A button labeled "apply transform" is positioned below the sliders. To the right of the sliders is a "result:" label.

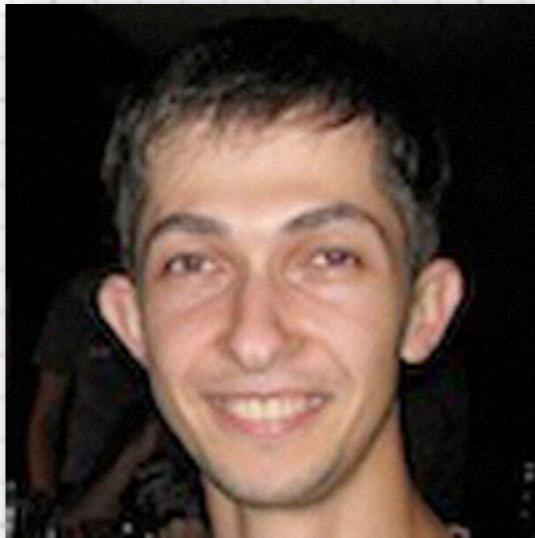
<http://codepen.io/1Marc/pen/DCvFm>

5. DEMOs!

6. RaphaelJS

Source
Your Sources

I learned from



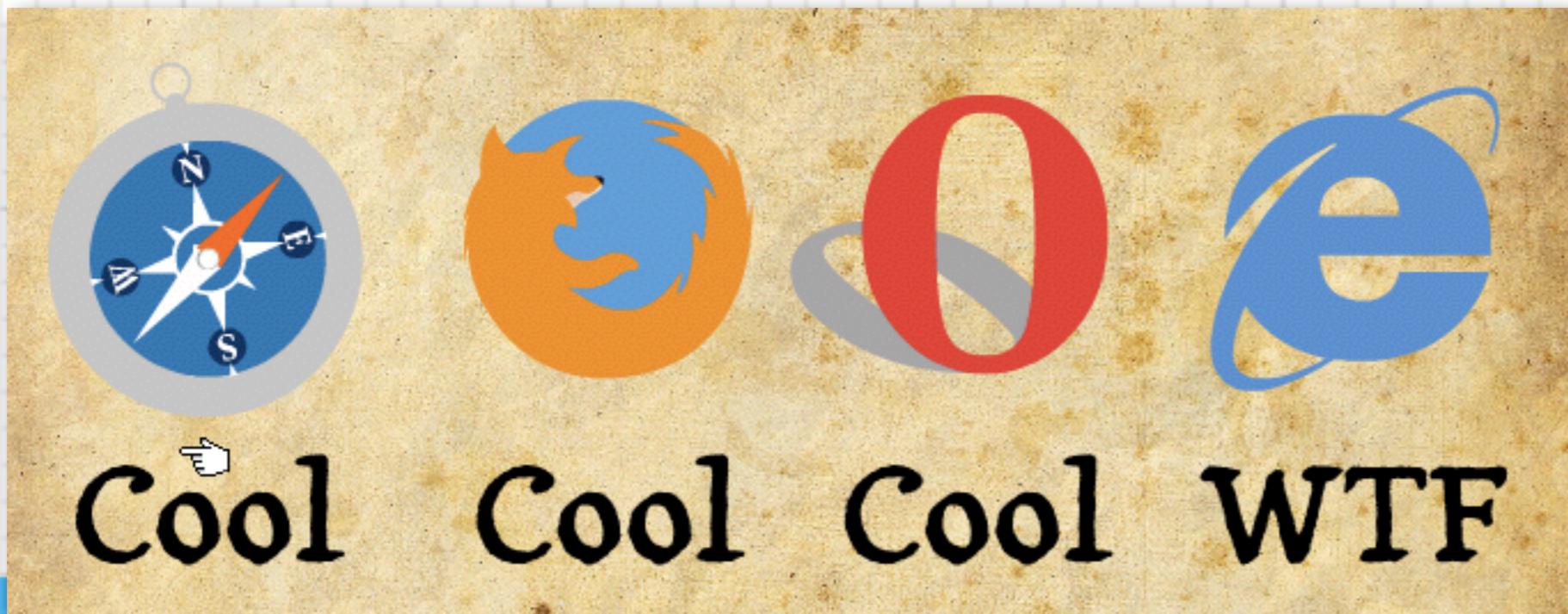
about using





Raphael JS

**SVG (Most Browsers) +
VML (MS IE6-8)**



Graphic Credit: Dmitry



Raphael JS

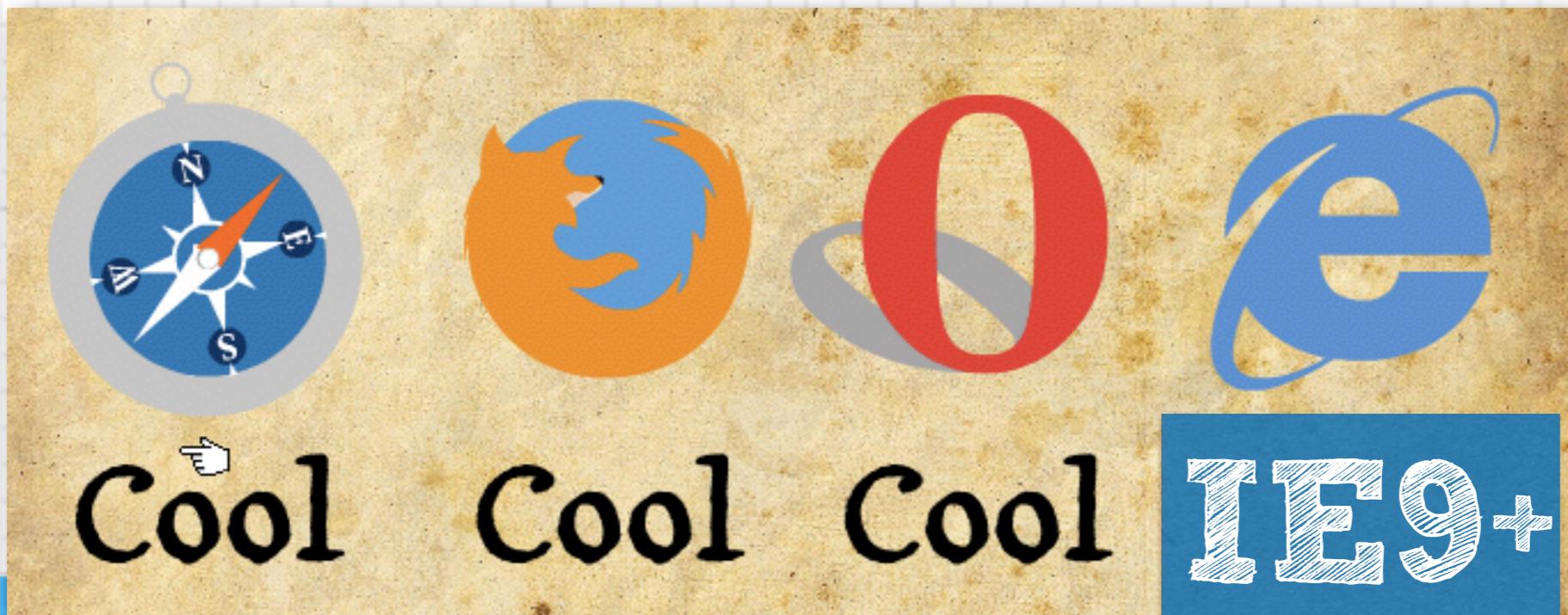
**“The jQuery of
Vector Graphics”**

- Common API for VML/SVG
- Utils make vector sane.



Dropping IE8

SVG Works Everywhere



Don't Support IE8? Use Snap.svg

Snap.svg

The JavaScript SVG library for the modern web.





Raphael.js

- **Primitives (Rectangle, Circle, etc)**
- **Attributes (Stroke, Dimensions, etc)**
- **Events (Click and Touch)**
- **Animation and Easing**
- **Sets (Grouping)**
- **Transforms (Rotate, Scale, etc)**

Transforms in Raphael



`el.transform("T 10,10");`

- move x, y

`el.transform("S 10,10,0,0);`

- scale 10,10 around 0,0

`el.transform("R 45,0,0);`

- rotate 45 around 0,0 (optional)

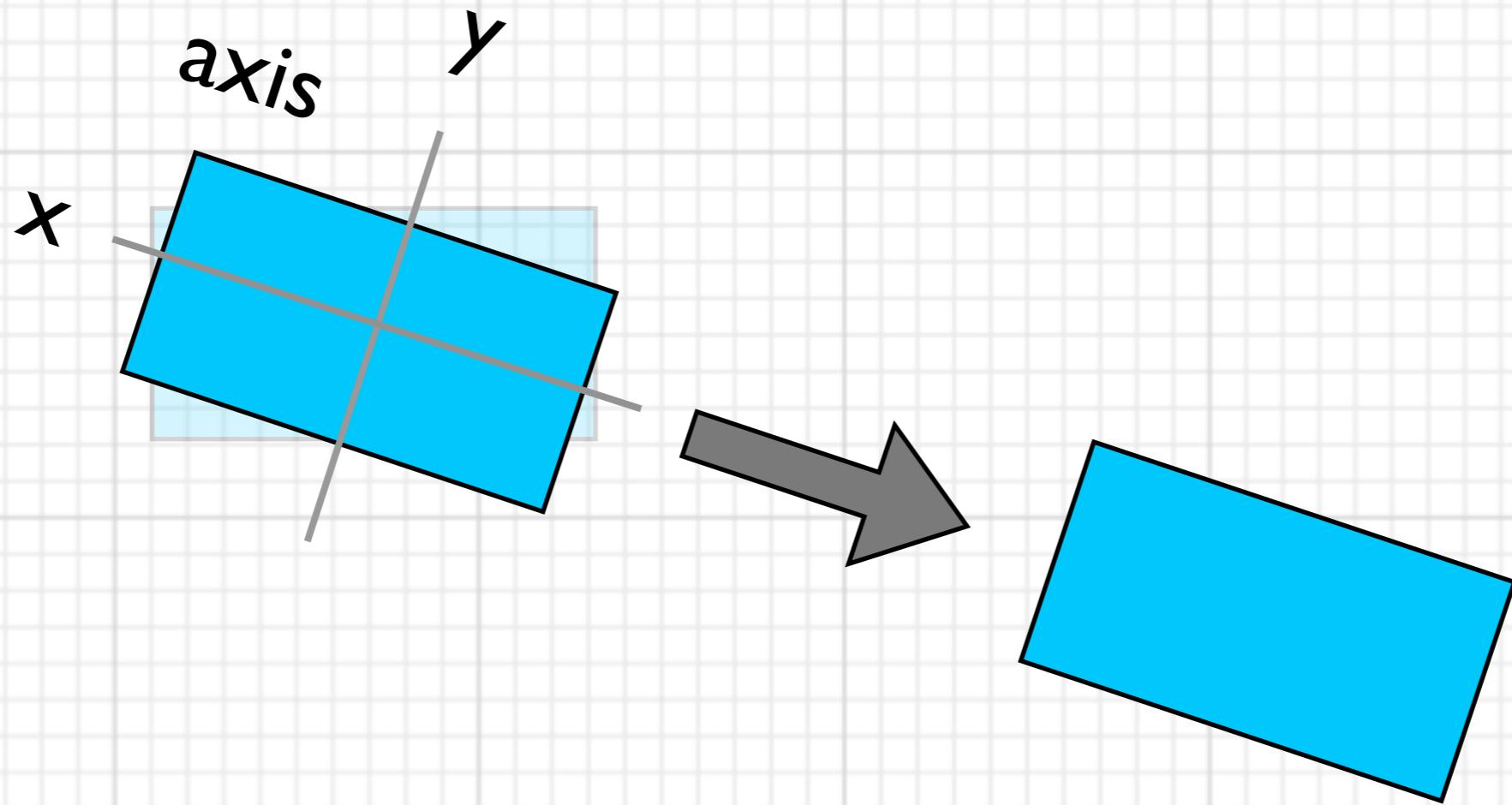


Transforms in Raphael

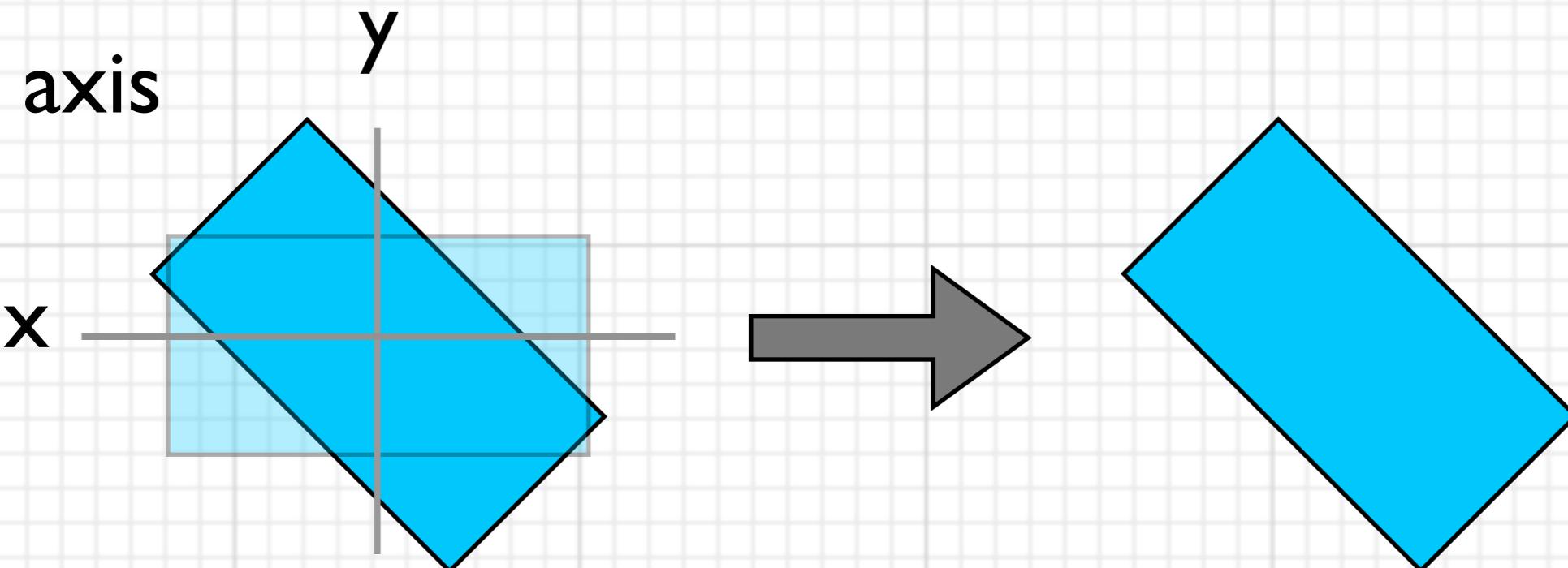
**Capital T, S and R absolute
transforms**

**Lower case t, s and r are relative
to previous transforms.**

Before Raphael



Absolute “T” transform



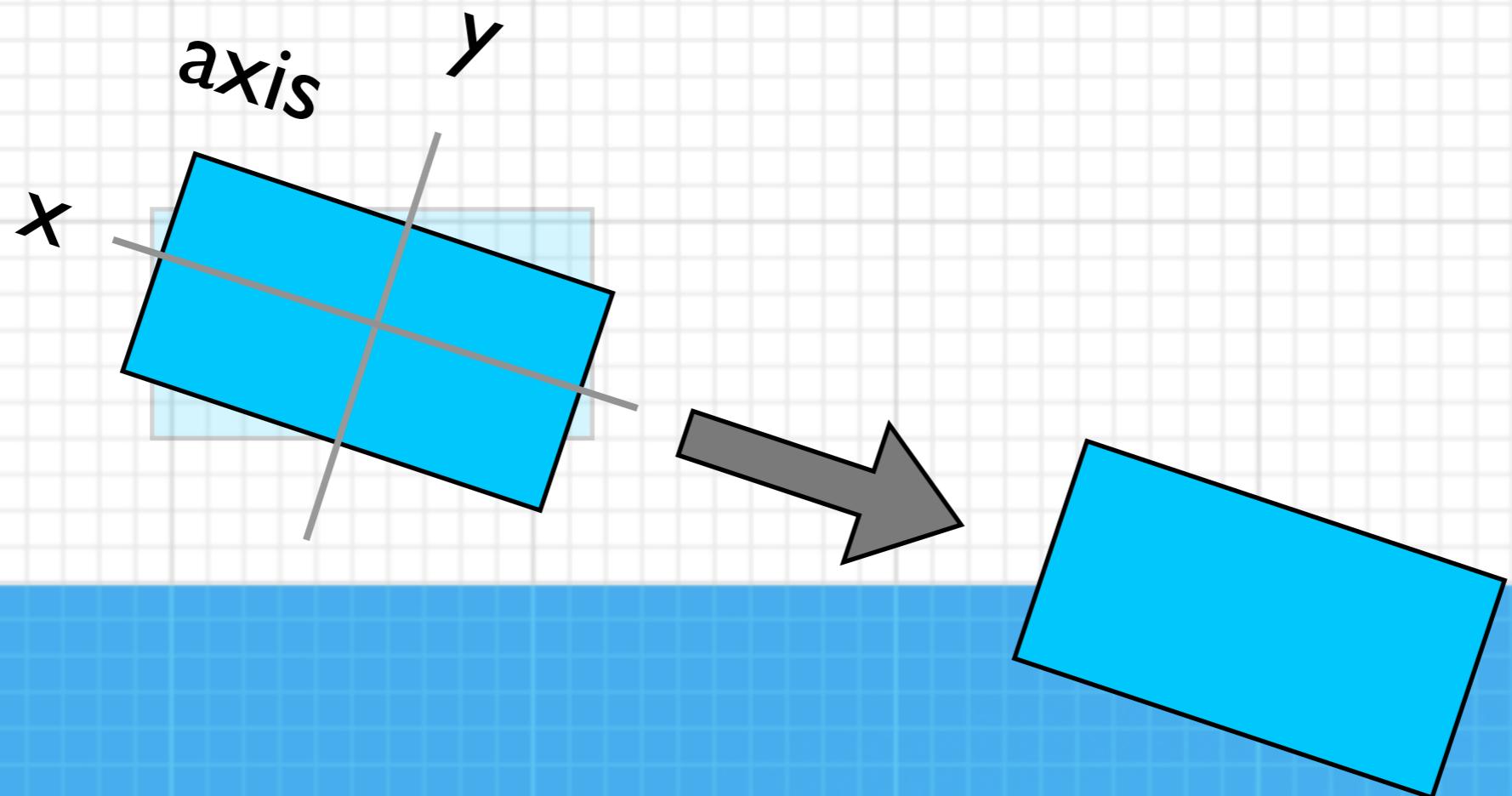
`transform('R45,100,50 T100,0')`

Relative to original axis

Relative “t” transform

`transform('r 20,100,50 t 100,0')`

Relative to current axis



Demo: Absolute/Relative Transforms in Raphael

CODEPEN

Save Fork Info Share Editor Details Professor Collab

HTML

```
<form id="transforms">
  <p style="text-align:left">Try stacking transforms by clicking "apply transform" and then add another.</p>
  <select id="transforms-select">
    <option data-transform="T">Absolute Translate T (x, y)
  </option>
    <option data-transform="S">Absolute Scale S (sx, sy, [ex], [cy])
  </option>
    <option data-transform="R">Absolute Rotate R (deg, [cx], [cy])
  </option>
</select>
  Try stacking transforms by clicking "apply transform" and then add another.
```

CSS

```
#paper {
  position: absolute;
  top:0;
  bottom:0;
  left:0;
  right:0;
  z-index: -1;
}

svg {
```

JS

```
// converted SVG to RaphaelJS with raphael.js
// https://github.com/DmitryBaranovskiy/raphael
var html5svg = [
  {"type": "path", "fill": "#E44026", "stroke": "none", "path": "M107.644,470.877 74.633,100.62 437.367,100.62 404.321,470.819 255.778,512 \t\t\ztz"}, {"type": "path", "fill": "#F16529", "stroke": "none", "path": "M256,480.523 376.03,447.246 404.27,130.894 256,130.894 \t\t\ztz"}, {"type": "path", "fill": "#EBEBEB", "stroke": "none", "path": "M256,130.894 376.03,447.246 404.27,130.894 256,130.894 \t\t\ztz"}]
```

A Pen by Marc Grabanski

Shortcuts

<http://codepen.io/1Marc/pen/rsmbF>



Transforms in Raphael

`el.transform('...t 10, 10');`

add transform to end

`el.transform('r45...');`

add transform before



Transforms in Raphael

`el.transform();`

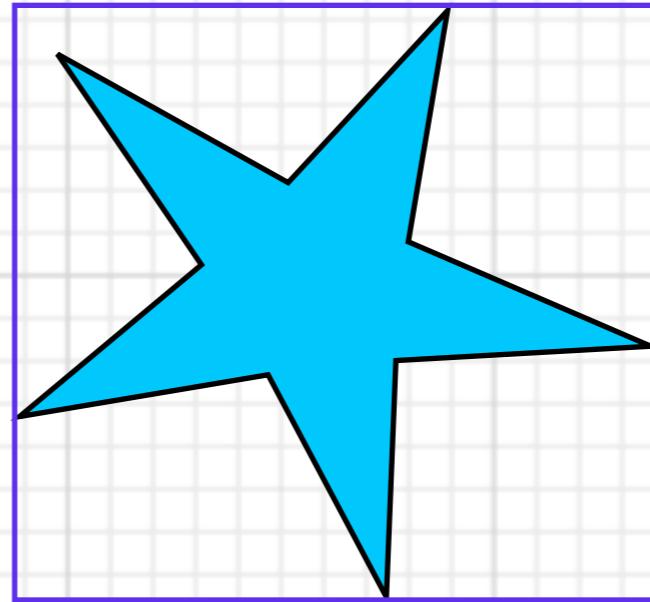
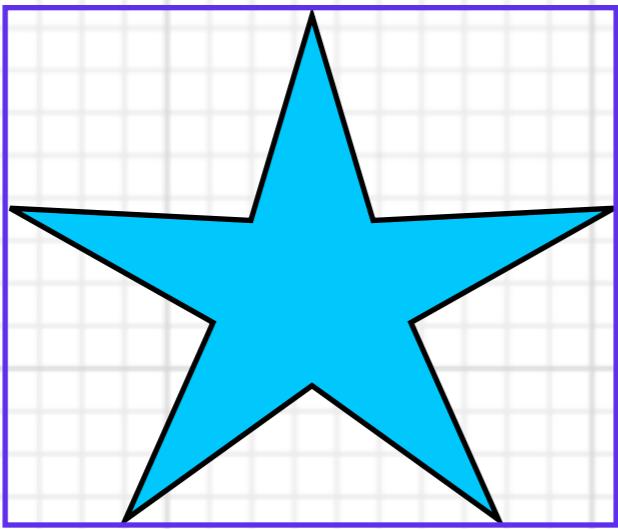
get the current transform object

`el.matrix.split();`

get results of all applied transforms

AWESOME method!

Bounding Box



`el.getBBox();`

gets x, y, width and height of
bounding box

7. Other Cool Tools



**Save on http requests by packing
multiple SVGs into one**

<http://jsvg.swissamigos.com/>

Fabric.js

Render SVG into Canvas

Convert Bitmap to SVG

vectormagic *Easily Convert Bitmap Images To Clean Vector Art*

Home **Online** Hand-Tracing Desktop Pricing Support

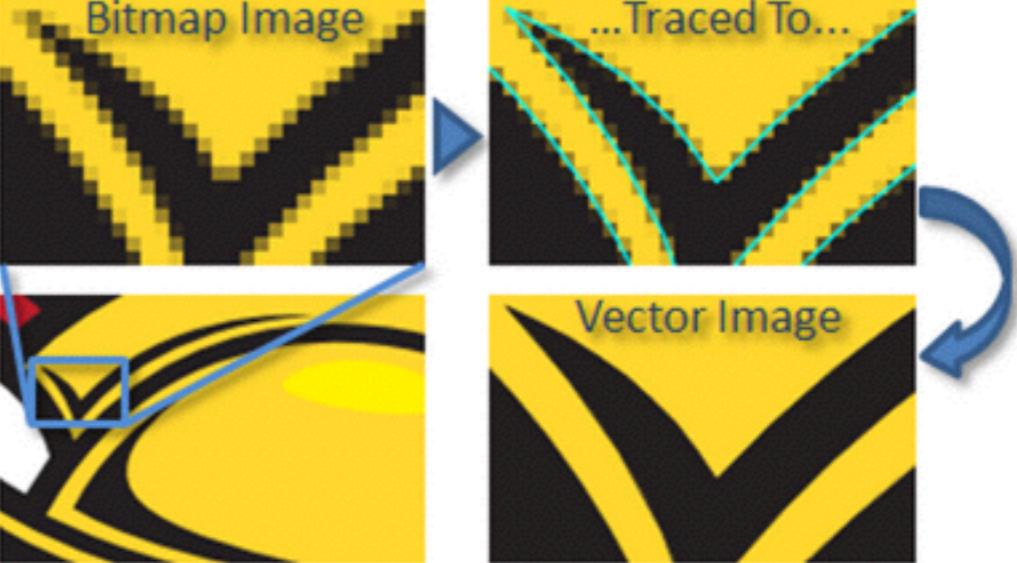
Welcome Comparisons Examples Buzz

Easily Convert Bitmap Images To Clean Vector Art

Automatically convert bitmap images like **JPEGs, GIFs and PNGs** to the crisp, clean, **scalable vector art of EPS, SVG, and PDF** with the world's best auto-tracing software.

Use the convenient service online by uploading your image below, or start tracing images offline with the [Desktop Edition](#).

Save time and money getting your artwork ready to print, embroider, cut and more!



Clown Car Technique Adaptive Images!

```
<svg xmlns="http://www.w3.org/2000/svg"
  viewBox="0 0 300 329" preserveAspectRatio="xMidYMid meet">
<title>Clown Car Technique</title>
<style>
  svg {
    background-size: 100% 100%;
    background-repeat: no-repeat;
  }
  @media screen and (max-width: 400px) {
    svg {
      background-image: url(images/small.png");
    }
  }
  ...
</style>
</svg>
```

<https://github.com/estelle/clowncar>

Vectron

Pull in SVG files with Ajax and convert them to Raphael JS.

Examples

Rendering assets/hippo.svg and assets/zebra.svg through RaphaelJS



<http://roomandboard.github.com/vectron/>

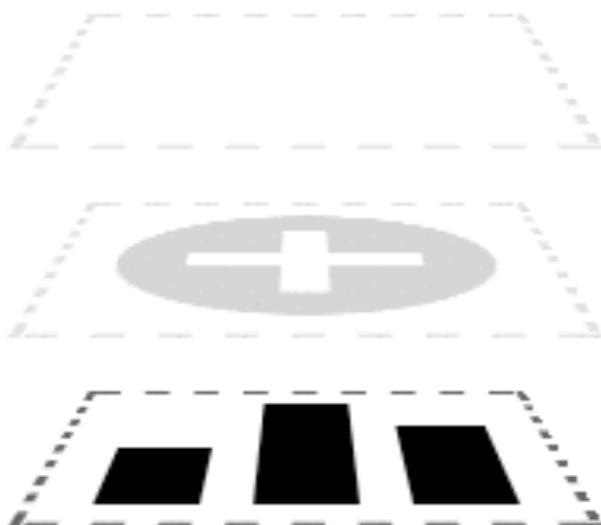
Future? SVG Stacks

css

```
background-image: url(icon.svg#chart);
```

SVG

```
.icon { display: none }  
.icon:target { display: inline }
```



```
<g id="plus" class="icon">
```

```
<g id="chart" class="icon">
```

Only works in Firefox (for now)

<http://simurai.com/post/20251013889/svg-stacks>

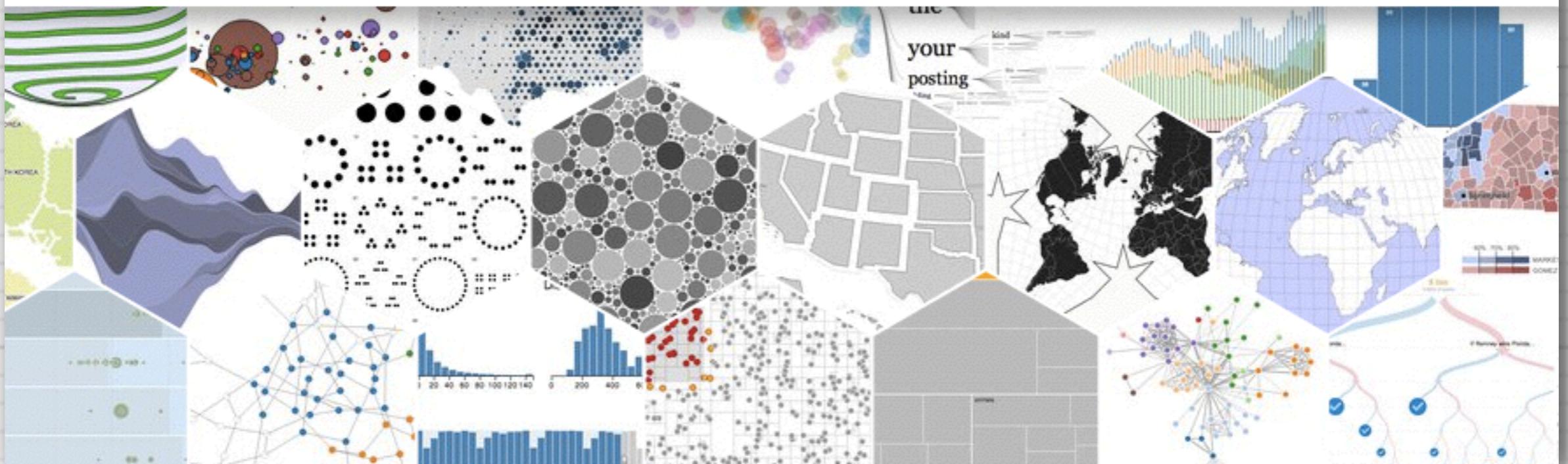
Snap.svg

The JavaScript SVG library for the modern web.





Data-Driven Documents



Marc Grabanski

@1marc