

Svg animasjon

Torsdag 26. Mai

svg

- SVG - Scalable vector graphic er et XML-basert filformat for markeringsspråk som beskriver todimensjonal vektorgrafikk.
- Svg tillater tre typer grafiske objekter: vektor grafiske former, bilder og tekst.
- Svg elementer kan også animeres ved bruk av css og SMIL Synchronized Multimedia Integration Language, eller JavaScript
- Her skal vi se på svg aminering ved bruk av javascript og svg jQuery.

Svg

- Svg animasjoner gjøres på svg elementer.
- Dette kan enten gjøres ved å lage egne svg elemeter i html

```
<svg width="200" height="200">  
  <rect id="rectangle1" x="50" y="50" rx="0" ry="0" width="100"  
    height="100" style="fill:red;stroke-width:5;opacity:0.5" />  
</svg>
```

Eller

- Du kan animere elementer fra en svg fil.

Basics

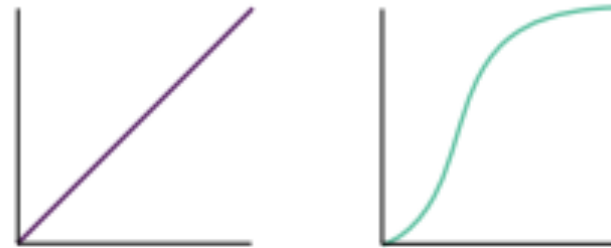
- SVG animasjon handler i hovedsak om endring av et svg elementets attributter. Dette for å endre form, plassering eller lignende.

```
$('#figurID').animate({  
  //endring av attributt  
},tid);
```

Timing

```
$('#figurID').animate({  
  //endring av attributt  
},tid);
```

```
$('#figurID').animate({  
  //endring av attributt  
},{easing:'linear', duration: animationSpeed});
```



Easing sier noe om hastigheten på animasjonen - linear og swing(default)

Flere animasjoner

```
$( '#figurID' ).animate({  
  //endring av attributt,      ← separeres med komma  
  //endring av attributt  
}, {easing: 'linear', duration: animationSpeed});
```

```
$( '#figurID' ).animate({  
  svgWidth: '100px',  
  svgHeight: '100px',  
}, {easing: 'linear', duration: animationSpeed});
```

Påfølgende animasjoner

- Dersom du ved avslutningen av en animasjon ønsker å fortsette med en ny handling kan du:

```
$( '#figurID' ).animate({  
    //endring av attributt  
}, {easing: 'linear',  
    duration: animationSpeed,  
    complete: function(){  
        //neste animasjon eller handling her.  
    }});
```

Animere flere objekter

```
$('#figurID1, #figurID2').animate({  
    svgWidth: '200px'  
}, tid);
```


Eksempler på animasjoner

```
$( '#figurID' ).animate({  
    svgWidth: '200px'  
}, tid);
```

```
$( '#figurID' ).animate({  
    svgTransform: 'translate(x, y)'  
}, tid);
```

svgTransform

- Enkel transform som flytter figuren til gitt punkt

svgTransform: `'translate(100,0)'`

- Enkel animasjon som roterer figuren til gitt vinkel

svgTransform: `'rotate(' + deg + ', 100, 100)'`

-

Kombinerer de overstående animasjonene:

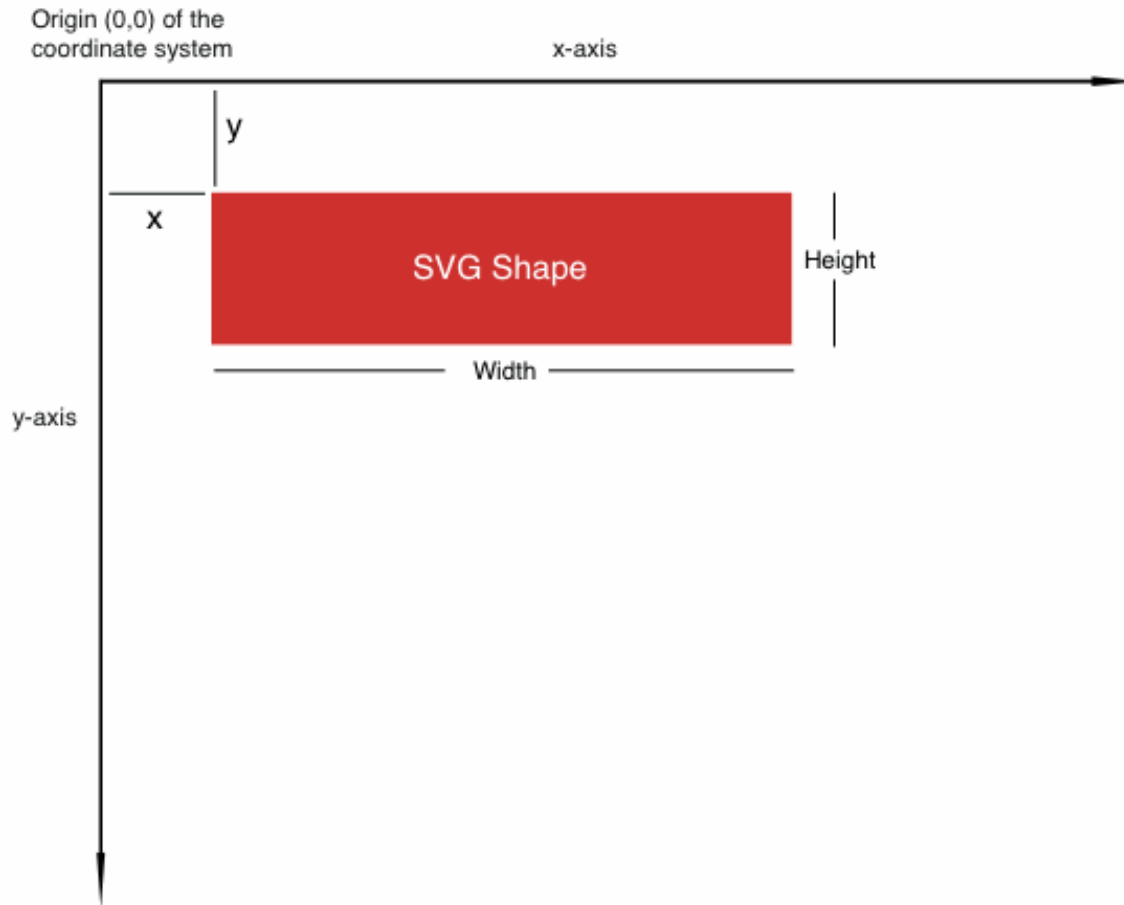
svgTransform: `'translate('+cx1+',0) rotate(' + deg + ', 100, 100)'`

ViewBox

- Magi som hjelper deg å gjøre et vektorelement skalerbart.
- Definerer koordinatsystemet og lager en ramme elementene dine skal passe inn i.
- På denne måten kan du wrappe svg figuren din i en viewBox og e.kes endre høyde og bredde på denne som igjen vil påvirke skalaen på figuren din.

```
<svg width="500" height="200" id="figur1"
viewBox="0 0 50 20">
  <rect x="20" y="10" width="10" height="5"
    style="fill:red;" />
</svg>
```

Svg koordinatsystem



Whats next

- Velocity - <http://julian.com/research/velocity/>
- Snap - <http://snapsvg.io/>
- SMIL - <https://css-tricks.com/guide-svg-animations-smil/>
- MDN (svg css styling) - <https://developer.mozilla.org/en-US/docs/Web/SVG/Element>