

# AWA1

## ANIMATION WEB

# Snap.svg

# Contenu

1. Introduction
2. Objet paper
3. Formes et tracés
4. Attributs
5. Groupes
6. Masques
7. Transformations
8. Animation
9. Importation SVG

# 1. Introduction Snap.svg

## Qu'est-ce que le Snap.svg ?

- Une librairie javascript pour créer et animer du contenu SVG.
- Basé sur Raphael.js (et crée par le même auteur Dmitry Baranovskiy)
- Ciblé pour les browsers récents
- Permet en plus que Raphael.js le maskage, clipping, patterns...
- Ou le trouver :
- <https://github.com/adobe-webplatform/Snap.svg/archive/v0.4.1.zip>

## 2. Objet paper

Insérer le script : `<script src="/script/snap.svg.js"></script>`

Soit on créer un nouvel objet paper dans le DOM :

```
var paper=Snap(largeur,hauteur);
```

Soit on a un svg existant :

```
<svg id="svg">...</svg>
```

et on l'utilise :

```
var paper=Snap("#svg");
```

```
var monElement = paper.select("#monElement").attr({fill:"#ff0000"});
```

### 3. Formes et tracés

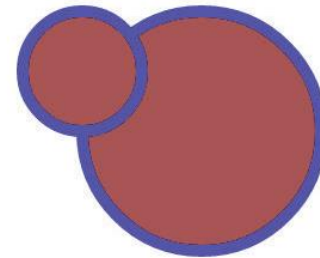
- `paper.circle(cx,cy,r)`
- `paper.line(x1,y1,x2,y2)`
- `paper.path("M10 10L90 90")`
- `paper.rect(x,y,w,h,rx,ry)`
- `paper.ellipse(x,y,rx,ry)`
- `paper.polyline(array of points)`
- `paper.polygon(array of points)`
- `paper.text(x,y,string)`
- `paper.image(src,x,y,width,height)`

## 4. Attributs

```
var circ=paper.circle(100, 100, 50);  
circ.attr( {  
  fill:"#ff0000",  
  stroke:"#0000ff",  
  strokeWidth:10  
});
```

## 5. Groupes

```
var paper = Snap(svgElement);  
var circ1 = paper.circle(100, 100, 50);  
var circ2 = paper.circle(200, 150, 100);  
group1 = paper.group(circ2, circ1);  
group1.attr( {  
  fill: "#ff0000",  
  stroke: "#0000ff",  
  strokeWidth: 10  
})
```



# 7. Transformations

// Définir une transformation

```
var t = new Snap.Matrix();
```

```
t.translate(100, 100);
```

```
t.rotate(20, 100, 100);
```

// Définir un objet ( ici un texte)

```
var text = paper.text(10, 10, "Mon texte");
```

// Appliquer la transformation a l'objet

```
text.transform(t);
```



## 8. Animation

Syntaxe :

```
Element.animate(attrs,duration,easing,callback);
```

Exemple :

```
var circ1 = paper.circle(100, 100, 50);  
circ1.animate({r:100},5000,mina.bounce);
```

Que se passe-il ?

le rayon du cercle passe de 50px a 100px en 5s avec un effet rebond.

Callback:

```
circ1.animate ({r:100, fill:"#b12626"},5000,mina.elastic ,function(){  
    circ1.attr({strokeWidth:1,transform:"t10,10s2,2});  
});
```

## 8. Animation (exemples)

<http://tympanus.net/Development/AnimatedSVGIcons/>

<http://tympanus.net/Development/ElasticSVGElements/button.html>

<http://tympanus.net/Tutorials/ShapeHoverEffectSVG/index3.html>

<http://tympanus.net/Development/WobblySlideshowEffect/>

<http://codepen.io/mattsince87/pen/snqLy> Animation sur un tracé

<http://codepen.io/ollieRogers/pen/rKoqs> Graphique animé

<http://codepen.io/houbly/pen/warado> Graphique paramétrable animé

## 9. Importation SVG externe

```
Snap.load("dessin.svg",dessinpret);  
  
// execute onDrawingLoaded quand le fichier est chargé  
  
function dessinpret(d){  
  var yellow=d.select("#yellowRect");  
  paper.append(yellow);  
  yellow.animate({width:600,height:600},4000);  
}
```

# Références

<http://snapsvg.io/docs/> tutorial pour démarrer

<http://snapsvg.io/docs/> documentation snap.svg