

Scalable Vector Graphics: des images en XML

L3Pro BDISE – XML

Mathieu Sassolas

IUT de Sénart Fontainebleau
Département Informatique

Année 2016-2017
Cours 6



SVG

M. Sassolas
L3Pro
Cours 6

Motivation

Dessiner

defs et use

Décorer

Animer

Trucs et
astuces

- 1 Dessiner textuellement
- 2 Les outils de base du SVG
- 3 Définir et réutiliser des choses
- 4 Décorer
- 5 Animer
- 6 Trucs et astuces

2 / 36

SVG

M. Sassolas
L3Pro
Cours 6

Motivation

Dessiner

defs et use

Décorer

Animer

Trucs et
astuces

- 1 Dessiner textuellement
- 2 Les outils de base du SVG
- 3 Définir et réutiliser des choses
- 4 Décorer
- 5 Animer
- 6 Trucs et astuces

3 / 36

SVG

M. Sassolas
L3Pro
Cours 6

Motivation

Dessiner

defs et use

Décorer

Animer

Trucs et
astuces

- ▶ On peut zoomer à l'infini.
- ▶ Pas de « pixellisation » à l'agrandissement.

Utilisation

- ▶ Bien pour les logos, les diagrammes. . .
- ▶ Pas adapté aux photos.

4 / 36

- ▶ Les photos (BMP, JPEG, etc) gardent (une compression de) valeur de couleur pour chaque pixel.
- ▶ Dans une image vectorielle, il n'y a pas de pixel.
- ▶ L'image est donnée par une description textuelle qui permet à l'afficheur de la retracer (en pixels).
- ▶ Par exemple **Postscript** dans le PDF et les imprimantes, **SVG** sur internet.

- ▶ Format basé sur **XML** (comme XHTML).
- ▶ Standard du W3C depuis 1998, la dernière version stable date de 2011 (SVG 1.1).
- ▶ Donne une description de l'image avec des **formes** et des **chemins** (*path*).
- ▶ Comme c'est du XML, peut être associé à des feuilles de style (CSS, XSLT).
- ▶ Peut également être généré à partir d'une transformation XSLT.

- 1 Dessiner textuellement
- 2 Les outils de base du SVG
- 3 Définir et réutiliser des choses
- 4 Décorer
- 5 Animer
- 6 Trucs et astuces

- ▶ On va évoquer les manières de déclarer objets, textes, et chemin.
- ▶ On va voir quelques éléments de **style**.
- ▶ On va toucher les limites du SVG à la main (càd écrit textuellement).

Le canevas : balise svg

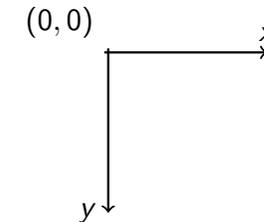
Définit une zone de dessin. Ce qui est hors du canevas ne sera pas affiché.

Un canevas A4

```
<svg width="21cm" height="29.7cm" version="1.1"
      xmlns="http://www.w3.org/2000/svg" >
  ...
</svg>
```

On peut définir des sous canevas en imbriquant des svg.
On omet alors la version et le xmlns mais on précise une position avec x et y.

- ▶ Le canevas est en fait infini : on peut dessiner hors de ses bords sans erreur (ce ne sera pas visible, mais ça existera).
 - ▶ Les dimensions fixent la manière par défaut de l'afficher (le tracer).
 - ▶ Les dimensions fixent le **système de coordonnées** à appliquer dans le canevas : dans une zone de 600px × 800px, le milieu a pour coordonnées 300px et 400px, même si on a zoomé pour afficher l'image en 1080p.
- ↔ Éviter de **mélanger** les systèmes de coordonnées.
- ▶ Le repère est orthonormal indirect :



```
<rect x="5cm" y="10cm" width="3cm" height="7cm"
      style="fill:rgb(54,123,42);stroke-width:3;
            stroke:#E42535"/>
```

```
<circle cx="5cm" cy="20cm" r="2cm" stroke-width="3"
        stroke="rgb(54,123,42)" fill="#E42535"/>
```

```
<ellipse cx="3cm" cy="7cm" rx="2cm" ry="1cm"/>
```

Notez que le style peut être spécifié de diverses manières.

```
<rect x="5cm" y="10cm" width="3cm" height="7cm"
      style="fill:rgb(54,123,42);stroke-width:3;
            stroke:#E42535"/>
```

```
<circle cx="5cm" cy="20cm" r="2cm" stroke-width="3"
        stroke="rgb(54,123,42)" fill="#E42535"/>
```

```
<ellipse cx="3cm" cy="7cm" rx="2cm" ry="1cm"/>
```

Notez que le style peut être spécifié de diverses manières.

Lignes, lignes brisées, polygones

SVG

M. Sassolas
L3Pro
Cours 6

Motivation

Dessiner

defs et use

Décorer

Animer

Trucs et
astuces

```
<polyline points="10,200 20,700 550,250 200,470"
  stroke="green" stroke-width="8" fill="blue"
  stroke-linejoin="round"/>
```

```
<polyline points="200,10 250,10 325,50 200,400
  100,700 100,100" stroke="red"
  stroke-width="5" fill="none"
  stroke-linecap="square"/>
```

```
<polygon points="450,600 300,750 500,500 250,750"
  stroke="orange" stroke-width="5" fill="lime"/>
```

```
<line x1="100" y1="10" x2="500" y2="753"
  stroke="purple"/>
```

12 / 36



Lignes

SVG

M. Sassolas
L3Pro
Cours 6

Motivation

Dessiner

defs et use

Décorer

Animer

Trucs et
astuces

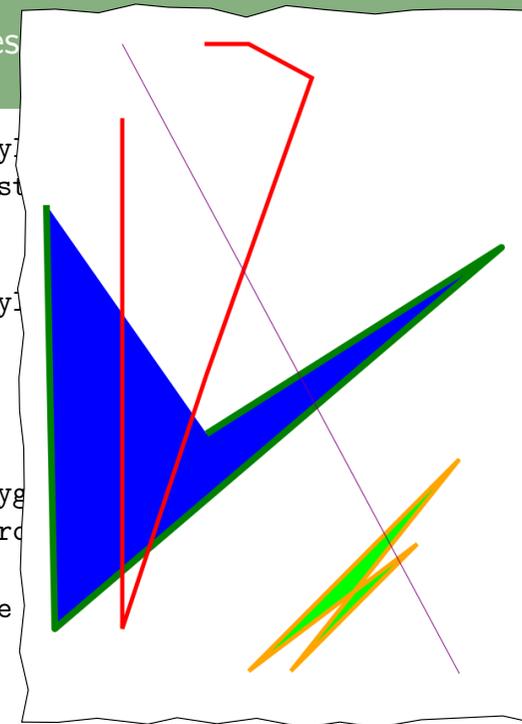
```
<polyline points="10,200 20,700 550,250 200,470"
  stroke="green" stroke-width="8" fill="blue"
  stroke-linejoin="round"/>
```

```
<polyline points="200,10 250,10 325,50 200,400
  100,700 100,100" stroke="red"
  stroke-width="5" fill="none"
  stroke-linecap="square"/>
```

```
<polygon points="450,600 300,750 500,500 250,750"
  stroke="orange" stroke-width="5" fill="lime"/>
```

```
<line x1="100" y1="10" x2="500" y2="753"
  stroke="purple"/>
```

12 / 36



Chemins

SVG

M. Sassolas
L3Pro
Cours 6

Motivation

Dessiner

defs et use

Décorer

Animer

Trucs et
astuces

- ▶ Manière la plus générale de dessiner.
- ▶ Généralise les `polyline` avec en plus des arcs de cercles et des courbes de Bézier.
- ▶ Le principe : une suite de commandes qui définit le chemin.
- ▶ Balise : `<path d="..." />`.

Les actions en majuscule entendent les coordonnées en absolu, celles en minuscule en relatif par rapport au point courant.

- M,m** *Moveto* déplace le curseur (sans dessiner).
- L,l** *Lineto* trace un trait.
- V,v** *Vertical lineto* trace un trait vertical.
- H,h** *Horizontal lineto* trace un trait horizontal.
- Z,z** Clôt le sous-chemin (la partie du chemin depuis le dernier *moveto*).

13 / 36



Chemins – exemple

SVG

M. Sassolas
L3Pro
Cours 6

Motivation

Dessiner

defs et use

Décorer

Animer

Trucs et
astuces

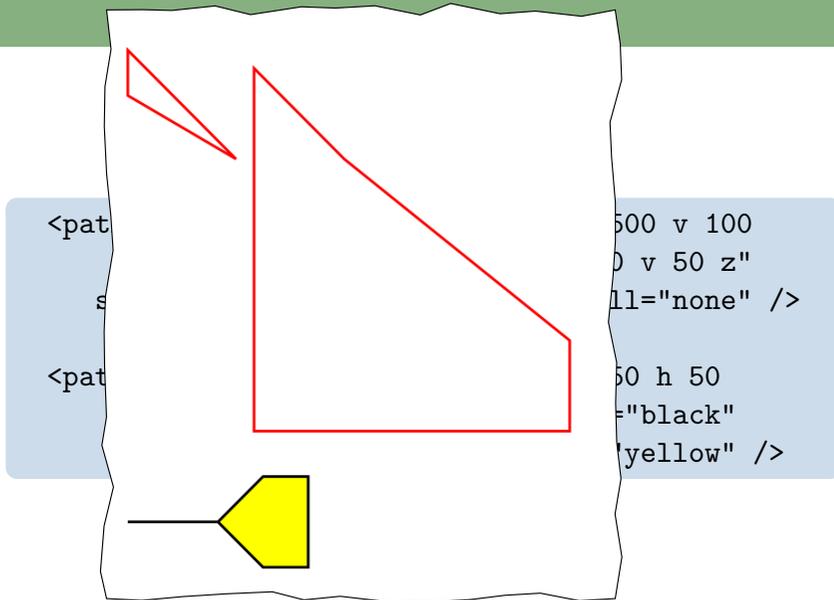
```
<path d="M 150 200 l 100 100 L 500 500 v 100
  h -350 z m -20 100 l -120 -120 v 50 z"
  stroke="red" stroke-width="3" fill="none" />
```

```
<path d="M 10 700 h 100 m 0 0 l 50 50 h 50
  v -100 h -50 z" stroke="black"
  stroke-width="3" fill="yellow" />
```

14 / 36



Chemins – exemple



Chemins – suite

- A,a** Arc trace un arc d'ellipse.
- C,c** Cubic Bézier Curve trace la courbe de Bézier (cubique).
- S,s** Smooth cubic Bézier Curve trace la courbe de Bézier (cubique) symétrique.
- Q,q** Quadratic Bézier Curve trace la courbe de Bézier (quadratique).
- T,t** Smooth quadratic Bézier Curve trace la courbe de Bézier (quadratique) symétrique.

Arcs de cercles et d'ellipses

- A** `rx ry angle large-arc sweep x y`
- rx et ry** Rayons de l'ellipse.
 - angle** Angle de rotation de l'ellipse.
 - x et y** Coordonnées du point d'arrivée.
 - large-arc** Choix de l'arc : 1 – grand (θ parcourt $360 - \alpha$) – ou 0 – petit (θ parcourt α).
 - sweep** Choix du sens de l'arc (par quel côté la ligne droite entre l'origine et la destination va être contournée).

Arcs de cercles et d'ellipses – exemple

- Départ et arrivée**
- ```
<circle cx="150" cy="100" r="5" fill="green"/>
<circle cx="250" cy="150" r="5" fill="blue"/>
```
- Large-arc=0, Sweep=0**
- ```
<path d="M 150,100 a 100,50 0 0,0 100,50" style="fill:none; stroke:red; stroke-width:6"/>
```
- Large-arc=1, Sweep=0**
- ```
<path d="M 150,100 a 100,50 0 1,0 100,50" style="fill:none; stroke:red; stroke-width:6"/>
```
- Large-arc=0, Sweep=1**
- ```
<path d="M 150,100 a 100,50 0 0,1 100,50" style="fill:none; stroke:red; stroke-width:6"/>
```

Arcs de cercles et d'ellipses – exemple

SVG

M. Sassolas
L3Pro
Cours 6

Motivation

Dessiner

defs et use

Décorer

Animer

Trucs et
astuces

17 / 36

Départ et arrivée

```
<circle cx="150" cy="100" r="5" fill="green"/>
<circle cx="250" cy="150" r="5" fill="blue"/>
```

Large-arc=0, Sweep=0

```
<path d="M 150,100 a 100,50 0 0,0 100,50"
      style="fill:none; stroke:red; stroke-width:6"/>
```

Large-arc=1, Sweep=0

```
<path d="M 150,100 a 100,50 0 1,0 100,50"
      style="fill:none; stroke:red; stroke-width:6"/>
```

Large-arc=0, Sweep=1

```
<path d="M 150,100 a 100,50 0 0,1 100,50"
```

Arcs de cercles et d'ellipses – exemple

SVG

M. Sassolas
L3Pro
Cours 6

Motivation

Dessiner

defs et use

Décorer

Animer

Trucs et
astuces

17 / 36

Départ et arrivée

```
<circle cx="150" cy="100" r="5" fill="green"/>
<circle cx="250" cy="150" r="5" fill="blue"/>
```

Large-arc=0, Sweep=0

```
<path d="M 150,100 a 100,50 0 0,0 100,50"
      style="fill:none; stroke:red; stroke-width:6"/>
```

Large-arc=1, Sweep=0

```
<path d="M 150,100 a 100,50 0 1,0 100,50"
      style="fill:none; stroke:red; stroke-width:6"/>
```

Large-arc=0, Sweep=1

```
<path d="M 150,100 a 100,50 0 0,1 100,50"
```

Arcs de cercles et d'ellipses – exemple

SVG

M. Sassolas
L3Pro
Cours 6

Motivation

Dessiner

defs et use

Décorer

Animer

Trucs et
astuces

17 / 36

Départ et arrivée

```
<circle cx="150" cy="100" r="5" fill="green"/>
<circle cx="250" cy="150" r="5" fill="blue"/>
```

Large-arc=0, Sweep=0

```
<path d="M 150,100 a 100,50 0 0,0 100,50"
      style="fill:none; stroke:red; stroke-width:6"/>
```

Large-arc=1, Sweep=0

```
<path d="M 150,100 a 100,50 0 1,0 100,50"
      style="fill:none; stroke:red; stroke-width:6"/>
```

Large-arc=0, Sweep=1

```
<path d="M 150,100 a 100,50 0 0,1 100,50"
```

Courbes de Bézier quadratiques

SVG

M. Sassolas
L3Pro
Cours 6

Motivation

Dessiner

defs et use

Décorer

Animer

Trucs et
astuces

18 / 36

q cx cy x y

cx et cy Coordonnées du point de contrôle.

x et y Coordonnées du point d'arrivée.

t x y

Pas de point de contrôle à donner, utilise le symétrique du dernier utilisé par rapport au point de départ.

```
<path d="M 50 200 q 100,50 200,25 100,250 200,25
        t 200,25" style="fill:none; stroke:orange;
        stroke-width:3"/>
```

Courbes de Bézier quadratiques

SVG

M. Sassolas
L3Pro
Cours 6

Motivation

Dessiner

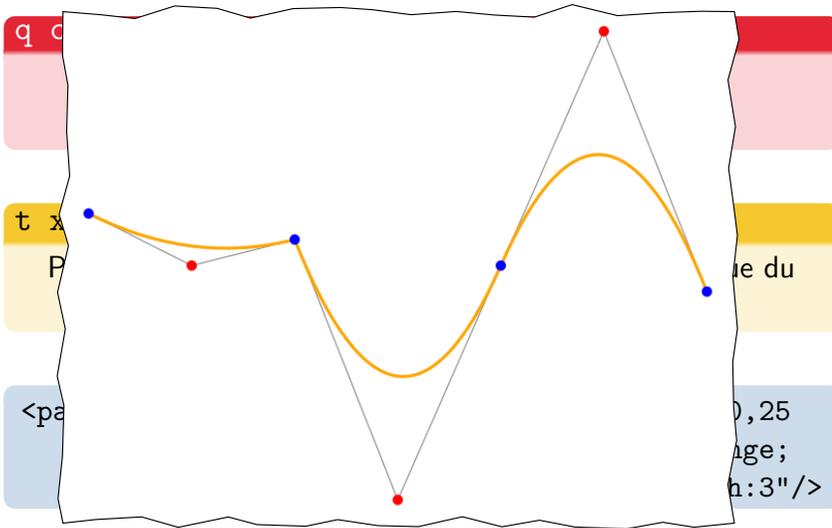
defs et use

Décorer

Animer

Trucs et
astuces

18 / 36



Courbes de Bézier cubiques

SVG

M. Sassolas
L3Pro
Cours 6

Motivation

Dessiner

defs et use

Décorer

Animer

Trucs et
astuces

19 / 36

c cx1 cy1 cx2 cy2 x y

cx1 et cy1 Coordonnées du premier point de contrôle.

cx2 et cy2 Coordonnées du second point de contrôle.

x et y Coordonnées du point d'arrivée.

s cx cy x y

Un seul point de contrôle à donner, utilise comme premier point le symétrique du dernier utilisé par rapport au point de départ.

```
<path d="M 50 100 c 100,200 150,100 200,0
          -20,250 100,250 125,100
          s 100,-100 200,25" style="fill:none;
          stroke:green; stroke-width:3"/>
```

Courbes de Bézier cubiques

SVG

M. Sassolas
L3Pro
Cours 6

Motivation

Dessiner

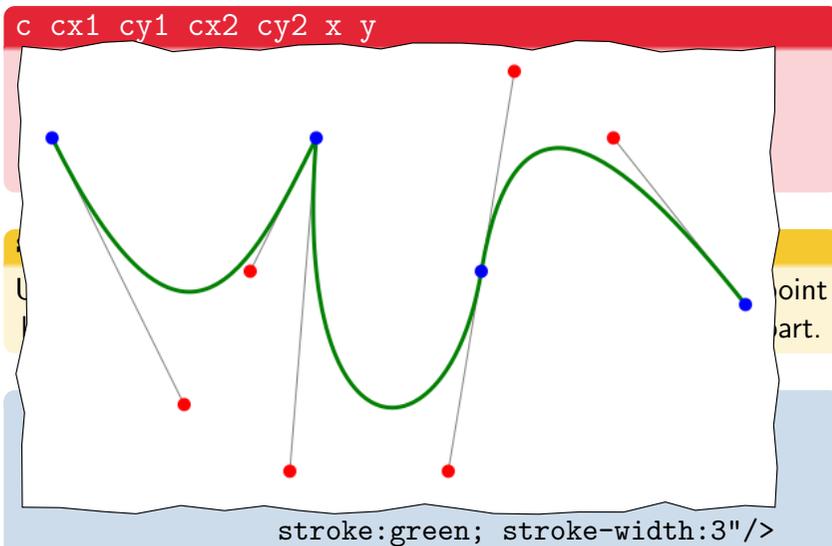
defs et use

Décorer

Animer

Trucs et
astuces

19 / 36



Caveat

SVG

M. Sassolas
L3Pro
Cours 6

Motivation

Dessiner

defs et use

Décorer

Animer

Trucs et
astuces

20 / 36

- ▶ Les courbes de Bézier sont assez difficiles à construire textuellement (on ne « voit » pas à l'avance le résultat).
- ▶ Il est intéressant pour cela d'utiliser des logiciels de dessin vectoriel : **Inkscape** (libre) ou **Adobe Illustrator** (pas libre et cher).
- ▶ Attention, ces logiciels produisent du SVG « sale » qu'il est difficile de modifier à la main ensuite.

- 1 Dessiner textuellement
- 2 Les outils de base du SVG
- 3 Définir et réutiliser des choses
- 4 Décorer
- 5 Animer
- 6 Trucs et astuces

Syntaxe

```
<svg ...>
  <defs>
    <!-- Ici on définit des choses -->
    <path ... id="unId" />
  </defs>
  <!-- Ici a vraiment le dessin -->
  <use xlink:href="#unId"/>
</svg>
```

Remarques

- ▶ C'est l'attribut id qui sert à faire le lien.
- ▶ On utilise xlink:href, il faut avoir déclaré le namespace xlink auparavant (à la racine).

```
<svg width="700px" height="500px" version="1.1"
      xmlns="http://www.w3.org/2000/svg"
      xmlns:xlink="http://www.w3.org/1999/xlink" >
  <defs>
    <path d="M 50 200 q 100,50 200,25
           100,250 200,25 t 200,25" id="monChemin"/>
  </defs>
  <text>
    <textPath xlink:href="#monChemin"
              startOffset="21">
      Lorem ipsum dolor sit amet, consectetur
      adipiscing elit. Sed quis metus quam.
    </textPath>
  </text>
</svg>
```

```
<svg width="700px" height="500px" version="1.1"
      xmlns="http://www.w3.org/2000/svg"
      xmlns:xlink="http://www.w3.org/1999/xlink" >
  <img alt="A diagram showing a wavy path with text following its curve. The text is 'Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed quis metus quam.' The path is defined by a path element with a 'd' attribute and a 't' attribute. The text is placed inside the path using a textPath element with a 'xlink:href' attribute pointing to the path and a 'startOffset' attribute." data-bbox="598 675 905 890"/>
  <textPath xlink:href="#monChemin"
            startOffset="21">
    Lorem ipsum dolor sit amet, consectetur
    adipiscing elit. Sed quis metus quam.
  </textPath>
</text>
</svg>
```

- 1 Dessiner textuellement
- 2 Les outils de base du SVG
- 3 Définir et réutiliser des choses
- 4 Décorer
- 5 Animer
- 6 Trucs et astuces

Chaque chemin ou objet peut avoir un attribut `style` qui comprend des propriétés à la CSS.

Principales propriétés

`stroke` couleur de la ligne.

`stroke-width` largeur du tracé.

`fill` couleur de remplissage (attention : défaut `black`, remplacer par `none` pour ne pas remplir).

Il faut définir les dégradés avant de les utiliser.

Syntaxe

```
<defs>
  <linearGradient id="unId" x1="..%" y1="..%"
                  x2="..%" y2="..%">
    <stop offset="..%" style="stop-color:...;
                          stop-opacity:.."/>
  </linearGradient>
  <radialGradient id="unAutreId" cx="..%" cy="..%"
                    r="..%" fx="..%" fy="..%">
</defs>
< ... style="fill:url(#unId)"/>
< ... style="fill:url(#unAutreId)"/>
```

```
<defs>
  <linearGradient id="grad1" x1="0%" y1="0%" x2="100%" y2="0%">
    <stop offset="0%"
            style="stop-color:rgb(255,0,255);stop-opacity:1" />
    <stop offset="20%"
            style="stop-color:rgb(255,255,0);stop-opacity:0.5" />
    <stop offset="40%"
            style="stop-color:rgb(255,255,0);stop-opacity:1" />
    <stop offset="100%"
            style="stop-color:rgb(255,0,0);stop-opacity:1" />
  </linearGradient>
</defs>
<rect x="75" y="150" width="170" height="55"
      stroke="black" fill="red" />
<rect x="75" y="95" width="170" height="55"
      stroke="black" fill="black" />
<ellipse cx="160" cy="150" rx="85" ry="55"
         stroke="black" fill="url(#grad1)" />
```

Exemple de dégradé linéaire

SVG

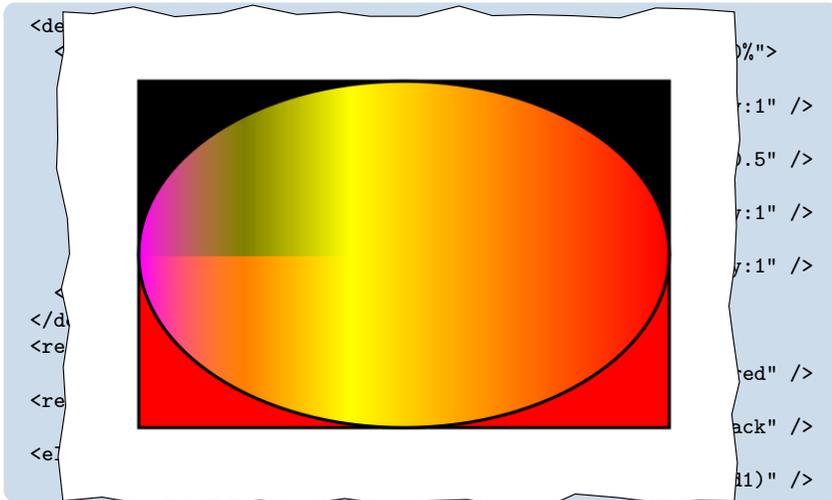
M. Sassolas
L3Pro
Cours 6

Motivation
Dessiner
defs et use

Décorer

Animer

Trucs et
astuces



27 / 36

Exemple de dégradé radial

SVG

M. Sassolas
L3Pro
Cours 6

Motivation
Dessiner
defs et use

Décorer

Animer

Trucs et
astuces

```

<defs>
  <radialGradient id="ball" cx="50%" cy="40%"
    r="70%" fx="30%" fy="30%">
    <stop offset="0%" style="stop-color:red"/>
    <stop offset="100%" style="stop-color:black"/>
  </radialGradient>
</defs>
<circle cx="150" cy="200" r="100"
  fill="url(#ball)"/>

```

28 / 36

Exemple de dégradé radial

SVG

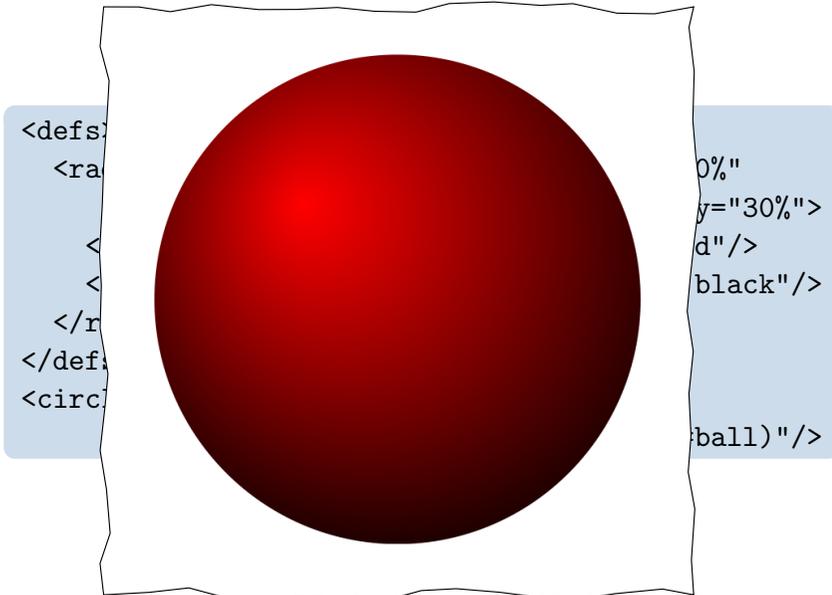
M. Sassolas
L3Pro
Cours 6

Motivation
Dessiner
defs et use

Décorer

Animer

Trucs et
astuces



28 / 36

Plan de la séance

SVG

M. Sassolas
L3Pro
Cours 6

Motivation
Dessiner
defs et use

Décorer

Animer

Trucs et
astuces

- 1 Dessiner textuellement
- 2 Les outils de base du SVG
- 3 Définir et réutiliser des choses
- 4 Décorer
- 5 Animer
- 6 Trucs et astuces

29 / 36

- ▶ Des éléments **animate** au sein des objets et chemins.
- ▶ Chaque animate spécifie l'attribut qui est modifié (coordonnée, taille, couleur).
- ▶ On spécifie les valeurs prises par l'attribut par from et to ou par values="val1;val2...".
- ▶ On spécifie le départ et durée de l'animation par begin et dur; on peut la répéter *n* fois avec repeatCount (indefinite pour répéter infiniment).
- ▶ Beaucoup de choses sont « animables », par exemple la couleur d'un stop d'un dégradé...

```
<path d="M 50 200 L 150,250 250,225"
      style="fill:none; stroke:grey; stroke-width:1">
  <animate attributeName="d" begin="0s" dur="12s"
            keySplines=".5 0 .5 1" repeatCount="indefinite"
            values="M 50 200 L 100,250 250,225;
                  M 50 200 L 200,350 250,225;
                  M 50 200 L 100,250 250,225"/>
</path>
<path d="M 50 200 Q 150,250 250,225"
      style="fill:none; stroke:orange; stroke-width:3">
  <animate attributeName="d" begin="0s" dur="12s"
            keySplines=".5 0 .5 1" repeatCount="indefinite"
            values="M 50 200 Q 100,250 250,225;
                  M 50 200 Q 200,350 250,225;
                  M 50 200 Q 100,250 250,225"/>
  <animate attributeName="stroke" attributeType="CSS" begin="0s"
            dur="12s" repeatCount="indefinite"
            values="orange;lime;green;lime;orange"/>
</path>
<circle cx="50" cy="200" r="5" fill="blue"/>
<circle cx="150" cy="250" r="5" fill="red">
  <animate attributeName="cx" begin="0s" dur="12s"
```

Syntaxe

```
<animateMotion path="..." />
```

```
<defs>
  <radialGradient id="ballM" cx="50%" cy="40%" r="70%"
                  fx="30%" fy="30%">
    <stop offset="0%" stop-color="red">
      <animate attributeName="stop-color" begin="0s" dur="3s"
                repeatCount="indefinite" values="red;blue;red"/>
    </stop>
    <stop offset="100%" style="stop-color:black"/>
  </radialGradient>
</defs>
<circle cx="150" cy="100" r="100" fill="url(#ballM)">
  <animateMotion path="M 0,0 c 100,200 150,100 200,0 -20,250
                    100,250 125,100 s 100,-100 200,25 z"
                begin="0s" dur="7s" repeatCount="indefinite"/>
</circle>
```

- 1 Dessiner textuellement
- 2 Les outils de base du SVG
- 3 Définir et réutiliser des choses
- 4 Décorer
- 5 Animer
- 6 Trucs et astuces

- ▶ On peut utiliser des **liens** hypertextes avec la balise `a` autour d'objets.
- ↪ Attention, utiliser `xlink:href="..."`.
- ▶ On peut utiliser **JavaScript** pour lire/éditer/contrôler le DOM du SVG.
- ↪ Des bibliothèques font ça très bien : Raphaël qui permet même de construire le SVG via du JavaScript.

- ▶ Lire la doc de SVG
(<http://www.w3.org/TR/SVG/Overview.html>).
- ▶ Je n'ai pas évoqué :
 - les motifs (*patterns*),
 - les filtres (*filters*),
 - les changement de coordonnées (*transformations*),
 - plein d'autres choses...

↪ C'est l'heure du TP ↩