

Travaux Pratiques Web et graphisme

Octobre 2016

JC.Moissinac – JC. Dufourd - C. Concolato

Introduction

Le but de ce TP est d'étudier des notions abordées lors du [cours](#) sur le web et en particulier le web graphique.

Nous manipulerons la technologie SVG (Scalable Vector Graphics) définie par le World Wide Web Consortium (W3C) et Canvas, en association avec du Javascript lorsque ce sera nécessaire.

Nous aborderons les aspects suivants: animations, organisation spatiale, et interactivité.

Vous pourrez vous aider de la spécification SVG [ici](#).

Ce livre en ligne pourra aussi vous aider. [SVG Primer](#)

Le site [W3Schools](#) peut être une aide efficace sur les technologies HTML, CSS, SVG, Javascript.

Compte-Rendu

Pour ce TP, vous devrez rendre, sous forme numérique, un compte-rendu dans lequel vous aurez répondu aux différentes questions. Vous joindrez également les fichiers que vous aurez créés/modifiés. Envoyez le tout à moissinac@telecom-paristech.fr (par exemple sous forme de fichier zip ou rar)

Partie 1 Découverte de SVG

Dans cette partie, nous allons travailler avec des fichiers SVG directement affichés dans un navigateur. Vous pourrez ouvrir ces fichiers dans un des principaux navigateurs de votre choix. Pour créer les fichiers, le plus simple est d'utiliser l'éditeur de texte de votre choix (PSPad, Sublime, ...).

Exercice 1 – Tracé de base, groupement de primitives graphiques, rôle de la viewBox

Nous allons découvrir quelques primitives dans l'objectif de créer une représentation visuelle évoquant un bouton d'une interface : un rectangle, éventuellement avec des coins arrondis, avec un bord et une couleur, et un texte.

Le format SVG utilise une structuration arborescente pour représenter une scène.

L'arbre de scène est structuré spatialement. Pour se rendre compte de cette structuration, créer une scène au format SVG d'une taille de 800 pixels par 600 pixels, comprenant 2 rectangles et deux textes comme dans l'image [boutons.png](#).

On groupera un texte et un rectangle pour composer un 'bouton'.

Voici une [scène vide](#).

Les primitives utilisées seront groupées dans un élément <g>.

Positionnez le groupe avec l'attribut transform par exemple,

transform= "translate(10,50) "

Progressivement, vous pourrez chercher à jouer avec les paramètres de couleur, de taille de texte...

Vous pouvez aussi grouper vos éléments avec des balises <svg> imbriquées dans la première.

Positionner le bloc <svg> avec des attributs x et y.

Expliquez le fonctionnement des attributs 'viewBox' et 'transform'. Commentez ce qu'il se passe quand deux objets sont placés aux mêmes coordonnées.

Exercice 2 – Association de comportement à une représentation graphique

On va associer un comportement au 'bouton' en mettant un attribut onclick ; pour valeur simple de cet attribut, on va mettre un appel à la fonction alert du javascript : alert('ici je mets mon texte') ;

Décrire ce qui se passe.

Créer plusieurs boutons avec plusieurs comportements

Activer les attributs de gestion d'événements onmouseover, onmousedown, onmouseup. Commentez leur rôle.

Exercice 3 – Manipulation par script de la représentation graphique

On va utiliser une fonction javascript qui récupère une caractéristique graphique d'un élément retrouvé par son ID et modifie cette caractéristique.

Pour cela, on va mettre un attribut id à chaque groupement de primitives constituant un bouton.

L'attribut id peut être ajouté à tous tag SVG ; sa valeur est une chaîne de caractère, qui doit être unique dans le document. Par exemple :

```
<rect id="r1" width="220" height="100" fill="blue" stroke="rgb(255, 128, 128)"/>
```

Recopier le bout de code suivant et l'insérer dans votre fichier SVG :

```
<script>
function set(w){
d = document.getElementById("r1"); // suppose qu'on a l'id 'r1' sur un élément
d.setAttribute("width", w);
}
</script>
```

Modifier de façon analogue la position ou la couleur de la cible de votre script; que se passe-t-il pour la gestion des évènements?

Remplacer le contenu du groupe par un bout de SVG plus complexe comme celui disponible [ici](#) (ou un autre trouvé sur le web). Analyser ce qui se passe.

Voir ce qui se passe avec un bouton composé de plusieurs cercles disjoints, notamment en matière de gestion d'évènement.

Exercice 4 – Manipulation déclarative de la représentation graphique

A l'intérieur d'un des éléments des fichiers svg précédents ajouter une ligne telle que :

```
<animate begin="3s" dur="2s" repeatCount="1" attributeName="fill" from="blue" to="red" fill="freeze" />
```

Jouer sur différentes formes et différents paramètres. (ne fonctionne pas avec tous les navigateurs; voir <http://caniuse.com/#feat=svg-smil>)

Associez plusieurs animate à un même élément.

Remarque: ne passez pas trop de temps sur cet exercice; les possibilités d'animation déclarative ont changées ou vont changer sur certains navigateurs ; vous pouvez tenter d'obtenir un effet similaire avec du code javascript