

Explorer les fichiers SVG



Le format SVG étant basé sur XML, il s’affiche tout à fait correctement dans tout éditeur de texte, éditeur dédié au XML, ou encore, beaucoup mieux, avec l’éditeur XML interne d’Inkscape.

RAPPEL XML

XML (*eXtensible Markup Language*) est un langage de balisage générique utilisé comme syntaxe de base du SVG. Ces deux formats sont plus amplement détaillés dans l’avant-propos.

Hormis l’intérêt primordial que procure la lecture d’un code pour la compréhension d’un langage, les possibilités pratiques sont nombreuses. Vous pouvez, par exemple, réorganiser rapidement et avec précision l’agencement des objets dans les plans et les calques, ajouter un objet dans un groupe (ou l’en extraire), ou encore modifier directement l’intégralité des paramètres du dessin.

Dans certains cas, vous pouvez même outrepasser certaines limites d’Inkscape, pour éviter des doublons d’identifiants lors de l’importation d’objets contenant des dégradés.

Enfin, Inkscape utilisant un SVG parfaitement conforme à la recommandation W3C, vous pouvez l’utiliser pour ajouter des éléments, y compris ceux qui ne sont pas encore pris en compte par le logiciel. Toutefois, ils risquent de ne pas s’afficher correctement sur le canevas. Vous pourrez tout de même les lire avec d’autres logiciels supportant une plus grande partie des spécifications du format.

EN DÉTAIL Le standard SVG

Cette annexe ne présente que succinctement le format SVG. Et pour cause : il représente, en version 1.1, plus de 700 pages. On a déjà vu pire, mais cela fait encore beaucoup pour parler de tout ici. Le texte intégral est disponible sur le site du W3C :

► <http://www.w3.org/TR/SVG11/>

L'éditeur XML

Pour commencer, ouvrons l'éditeur avec le menu *Édition* > *Éditeur XML...* (*Maj + Ctrl + X*). La fenêtre suivante apparaît.

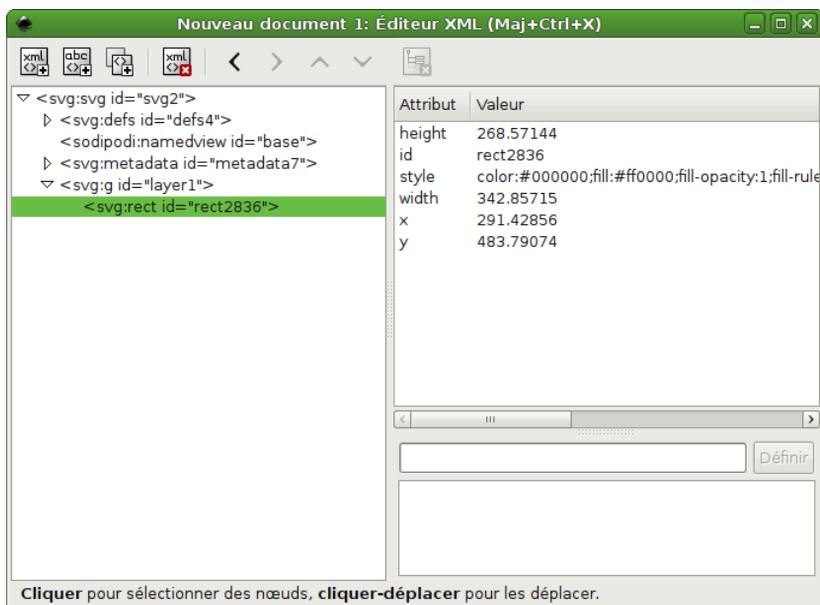


FIGURE 3-1 L'éditeur XML

Voyons comment s'organise cette fenêtre. La partie gauche de l'interface affiche les éléments (également appelés nœuds) utilisés par le dessin, sous

une forme hiérarchisée. Lorsqu'un élément en contient un ou plusieurs autres (pour former un groupe, par exemple), il est précédé d'un triangle. Pour afficher ou masquer les objets contenus sous cet élément, il suffit de le sélectionner et de cliquer sur le triangle.

À SAVOIR **Toujours synchronisé**

À l'ouverture, l'éditeur se positionne sur l'objet sélectionné sur le canevas lors de son lancement. La ligne contenant cet objet dans le code se distingue par son texte blanc sur fond bleu. Lors de l'édition, la sélection dans le SVG est synchronisée avec celle du canevas.

Sur la droite, l'éditeur XML détaille les différents attributs de l'élément sélectionné. Pour modifier la valeur d'un attribut, il suffit de le sélectionner, d'entrer la nouvelle valeur dans la zone de saisie en bas à droite et d'appuyer sur le bouton *Définir*. Pour ajouter un nouvel attribut, utilisez la même méthode en entrant son nom dans la zone de saisie à gauche du bouton.

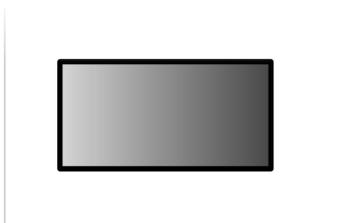
La barre placée tout en haut de l'éditeur propose un jeu d'icônes dédié à l'ajout et la suppression de nœuds, à leur déplacement, et à la suppression d'attributs. Les déplacements d'éléments, dans la partie gauche de l'éditeur, peuvent également être réalisés avec la souris, en cliquant-glissant.

À SAVOIR **Ordre des éléments**

Inkscape dessine les éléments par ordre d'apparition dans le code SVG. Ainsi, un objet placé tout en bas dans la liste de l'éditeur sera placé au-dessus des autres objets sur le canevas. Vous pouvez utiliser cette propriété pour inverser la position dans le plan de deux objets en permutant leur rang dans le code SVG.

Au cœur du SVG

Pour illustrer de façon simple un code SVG basique, prenons l'exemple d'un dessin ne comportant qu'un rectangle, dont le remplissage est dégradé (de gris clair vers gris foncé) et le contour noir.

FIGURE 3-2 *Un rectangle simple*

Ouvrons l'éditeur XML (*Maj + Ctrl + X*) et explorons le contenu du fichier.

La première ligne spécifie l'élément `svg` :

```
<svg
  width="800px"
  height="600px"
  id="svg3512"
  sodipodi:version="0.32"
  inkscape:version="0.46+devel"
  sodipodi:docname="Annexes_XML_rectangle.svg"
  inkscape:output_extension=
    "org.inkscape.output.svg.inkscape">
```

Les attributs `width` et `height` spécifient la taille du canevas. Viennent ensuite l'identifiant (`id`) du document, la version du logiciel utilisé pour son édition, le nom du fichier (`docname`) et le type de SVG enregistré (`output_extension`). Notez que certains attributs utilisent un espace de noms `sodipodi`. Il était utilisé par le logiciel Sodipodi, prédécesseur d'Inkscape, et a été conservé pour garantir la compatibilité ascendante avec les fichiers qui en étaient issus.

EN DÉTAIL Espace de noms XML

Un espace de noms est un groupe d'éléments et d'attributs particuliers, définissant un vocabulaire. Comme tout document XML, un document SVG peut utiliser des espaces de noms spécifiés pour d'autres applications. C'est le cas ici des attributs `docname` ou `output_extension`, qui n'existent pas dans le standard SVG, mais ont été ajoutés, pour les besoins d'Inkscape, respectivement dans les espaces de noms `sodipodi` et `inkscape`.

Juste en dessous de l'élément `svg` se trouve l'élément `defs` :

```
<defs
  id="defs4">
  <linearGradient
    id="linearGradient3155">
    <stop
      style="stop-color:#d8d8d8;stop-opacity:1;"
      offset="0"
      id="stop3157" />
    <stop
      style="stop-color:#424242;stop-opacity:0.97560978;"
      offset="1"
      id="stop3159" />
    </linearGradient>
    <linearGradient
      x1="50"
      y1="949.86218"
      x2="255"
      y2="949.86218"
      id="linearGradient3161"
      xlink:href="#linearGradient3155"
      gradientUnits="userSpaceOnUse" />
  </defs>
```

`defs` sert à définir toutes sortes de données qui pourront être réutilisées à plusieurs reprises dans le document. C'est le cas par exemple des dégradés, des motifs ou des marqueurs. Ici, cet élément contient deux éléments de type dégradé linéaire (`linearGradient`). Le premier, contenant dans des éléments `stop` la définition des couleurs, est unique et partagé par tous les objets qui en ont besoin. Le second indique la position du dégradé sur le canevas pour chaque objet faisant appel au dégradé. Si un autre objet utilise le même dégradé, ce dernier élément sera copié et se verra attribuer une position et un identifiant (`id`) différents.

L'élément `namedview` définit les paramètres de visualisation du document. Il s'agit, entre autres, de la taille de la fenêtre de l'application, la couleur de l'espace de travail, la grille, le niveau de zoom... Cet élément est intrinsèque à Inkscape et ne fait pas partie du standard SVG.

Vient ensuite `metadata`, qui sert à ajouter toutes sortes d'informations annexes relatives au document (licence, auteur...).

La partie qui nous intéresse commence par `g`, qui correspond à un groupe d'objets. Le voici :

```
<g
  id="layer1"
  inkscape:label="Layer 1"
  inkscape:groupmode="layer">
```

Il possède ici un attribut particulier, `inkscape:groupmode="layer"`, qui fait qu'il est interprété par Inkscape comme un calque. Sans cet attribut, le groupe prend son sens habituel : un ensemble d'objets pouvant être manipulés comme un élément unique.

Et voici enfin notre rectangle, sous le calque.

```
<rect
  style="opacity:1;fill:url(#linearGradient3161);
  fill-opacity:1;fill-rule:nonzero;stroke:#5c5c5c;
  stroke-width:5;stroke-linecap:square;
  stroke-linejoin:round;stroke-miterlimit:4;
  stroke-opacity:1;stroke-dasharray:none;
  stroke-dashoffset:0;marker:none;visibility:visible;
  display:inline;overflow:visible;
  enable-background:accumulate"
  id="rect3603"
  width="195"
  height="95"
  x="52.5"
  y="452.50003" />
```

Il contient des attributs concernant sa taille (`width` et `height`), sa position (`x` et `y`), son style et un identifiant (`id`). L'attribut `style` est particulièrement chargé car il contient tous les paramètres définis dans la boîte de dialogue *Remplissage et contour...* Vous y retrouvez en particulier, dans le paramètre `fill`, un lien vers le dégradé défini précédemment.

EN DÉTAIL Coordonnées et tailles

Vous l'avez sans doute remarqué, les attributs de taille sont légèrement différents de ceux affichés dans la barre de contrôle de l'outil. Cela s'explique par un traitement différent entre le standard SVG, qui considère que la taille ne prend pas en compte le contour (ici de 5 pixels) et Inkscape. Pour la position, la différence est encore plus marquée, puisque la position de SVG est prise au milieu du contour (d'où 2,5 pixels de différence) et dans un référentiel où l'axe des ordonnées est orienté vers le bas. Heureusement, à moins de fouiller dans le code, tout cela est parfaitement transparent...

Voilà pour notre rectangle. Le SVG connaît bien d'autres éléments, tels que les chemins, les textes, les motifs... que nous ne pouvons pas examiner en détail ici. Mais n'hésitez pas à explorer votre code, vous en apprendrez toujours plus sur le fonctionnement du format SVG et sa gestion par Inkscape.

Pour aller plus loin, référez-vous à la liste des liens proposée dans l'annexe suivante. Vous y trouverez des ressources utiles pour en apprendre davantage sur le SVG ainsi que de nombreux tutoriels pouvant servir de support à une analyse approfondie du code.

Si vous avez besoin d'une référence sur un point précis, orientez vos recherches vers la spécification SVG, à l'adresse <http://www.w3.org/TR/SVG11/>. Le document est long et peu illustré, mais clair et bien organisé. Nul doute que vous y trouverez rapidement l'information qui vous manque.