Essentials of SVG

Peter Occil

Essentials of SVG

This version of the document is dated 2023-06-13.

The following short note explains the most useful things to know when writing vector graphics to SVG.

First, we start with a skeleton of an SVG file (which uses the XML data format):

```
<svg width='300px' height='300px' viewBox='0 0 100 100'
     xmlns='http://www.w3.org/2000/svg'>
     <ellipse style='stroke:none;fill:red' cx='10' cy='10' rx='2' ry='2'/>
     <path style='stroke:black;fill:none' d='M10 10L20 20 L30 10Z'/>
</svg>
```

In this simple example, we can already see several useful things:

- `width` and `height` give the size of the SVG in pixels relative to the host document.
- `viewBox` gives the size of the shape in SVG units (not pixels). It has four numbers giving the left side, top side, width, and height. Y coordinates run downwards.
- `ellipse` draws an ellipse. `cx`, `cy`, `rx`, and `ry` gives its center's X and Y coordinates and its horizontal and vertical radii, all in SVG units.
- `path` gives the shape of the path in a compact form specified as the `d` attribute. Each path is broken up into commands, which are **detailed further in the SVG specification**. The most important of these are perhaps M, L, and Z: the M command moves the pen; the L command draws with the pen, moving it to a new position; and the Z command closes the shape by drawing with the pen straight to the shape's beginning. In general, numbers given in the path specification are in SVG units, relative to the SVG document itself.
- `style` gives styling instructions for the path or ellipse. Perhaps the most important style rules are `stroke`, `fill`, and `stroke-width`, and the following are examples of the `style` attribute, which are mostly self-explanatory: `style='stroke:red;stroke-width:1px'`, `style='stroke:none;'`, `style='fill:blue;'`, `style='fill:none;'` (here `px` means an SVG unit, relative to the SVG document itself).

An SVG document can have any number of `path` elements, `ellipse` elements, or both, and each of these elements is a separate shape. Finally, each `path` and `ellipse` element can have a `transform` attribute, which describes an optional geometric transformation. The attribute's value is made up of **one or more commands** described further in the SVG specification. The commands, which are followed in the order written, include "translate(x,y)" which moves the zero-point; "rotate(d)" which rotates the coordinate system by d degrees, and "scale(x)" which scales the coordinates by a factor of x. Examples of the attribute include `transform='translate(3, 5)'`, `transform='rotate(90)'`, and `transform=' translate(2, 2) scale(2)'`.

1 License