SVG Crashkurs

1 Einleitung

Was ist SVG?

SVG steht für *Scalable Vector Graphics* und ist eine Beschreibungssprache (der XML-Familie) für Vektorgrafiken und gleichzeitig auch ein Grafikformat.

Als eigenständiges Grafikformat hat SVG die Endung `.svg`.

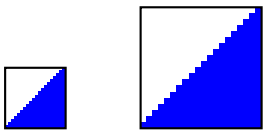
SVG-Grafiken können von den meisten Browsern und vielen Grafikprogrammen dargestellt werden.

Rastergrafiken

Um den Begriff *Vektorgrafik* besser verstehen zu können muss man zunächst wissen, was *Rastergrafiken* sind.

Rastergrafiken speichern Bilder mittels Bildpunkten (Pixeln) ab, die gewissermassen die Atome des Bildes sind.

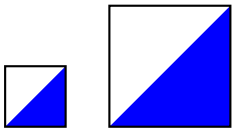
Bei niedriger Pixeldichte (oder hoher Vergrösserung) wird diese Struktur sichtbar.



Vektorgrafiken

Vektorgrafiken definieren ein Bild durch grafische Primitiven wie Linien, Kreise oder Splines. Anstatt alle Bildpunkte einer Strecke zu definieren, werden die Koordinaten der Endpunkte und die Darstellungsparameter (Liniendicke, Linenfarbe, ...) angegeben.

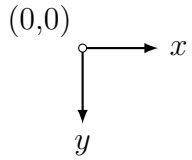
Diese Darstellung ermöglicht es, Grafiken ohne Qualistätsverlust zu skalieren. Auch der meist geringere Speicherbedarf ist ein Vorteil gegenüber Rastergrafiken



Da Bildschirme und Drucker auf Pixelbasis arbeiten, ist am Ende dennoch eine Umwandlung in ein Rasterbild erforderlich.

Das Koordinatensystem

Die x-Koordinate nimmt von links nach rechts, die y-Koordinate von oben nach unten zu.



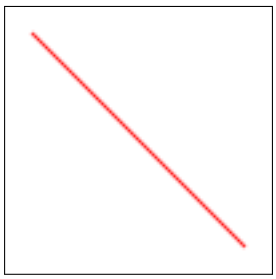
Die Dokumentstruktur

Da SVG-Dateien spezielle XML-Dokumente (Extensible Markup Language) sind, müssen sie eine bestimmte Struktur haben.

```
1 <?xml version ="1.0"?>
2 <svg width="100" height="100"
3      xmlns="http://www.w3.org/2000/svg">
4   <line x1="10" y1="10" x2="90" y2="90" stroke="red" />
5 </svg>
```

- Zeile 1 besagt, dass es sich um ein XML-Dokument handelt.
- `<svg ...> ... </svg>` ist das *Wurzelelement*, das zur Anzeige der Grafik führt. In der öffnenden Markierung (*tag*) befinden sich *Attribute* der Form `key="value"`.
- Das Attribut auf Zeile 3 definiert den XML-Namensraum.
- Zeile 5 enthält das *Kindelement* `<line ... />`.

Das Öffnen der obigen SVG-Datei führt zur folgenden Ausgabe:



Zahlen

Für die Angabe von Koordinaten können ganze und gebrochene Zahlen verwendet werden. Als Dezimaltrennzeichen wird der Punkt verwendet. Auch die vom Taschenrechner bekannte E-Schreibweise ist erlaubt. Beispiele:

- `510`
- `-22`
- `24.53`
- `-0.825` oder `-.825`
- `7.9E2` oder `7.9e2`

Einheiten

Üblich sind unter anderem:

- `px` (Pixel)
- `cm` (Zentimeter)
- `mm` (Millimeter)
- `in` (Zoll)
- `pt` (1/72 Zoll)

Längen

Eine Länge bzw. eine Längenangabe besteht aus einer Zahl und einer Längeneinheit. Beispiele:

- `10px`
- `2.54cm`
- `-18mm`

Ohne Angabe einer Längeneinheit wird die Basiseinheit `px` (Pixel) verwendet.

Farben

Farben können durch die Angabe von Farbwerten (0, 1, . . . , 255) für den Rot-, den Grün- und den Blauanteil definert werden

Farbname	RGB-Darstellung	Hex-Darstellung
red	`rgb(255,0,0)`	`#FF0000`
green	`rgb(0,255,0)`	`#00FF00`
blue	`rgb(0,0,255)`	`#0000FF`
black	`rgb(0,0,0)`	`#000000`
white	`rgb(255,255,255)`	`#FFFFFF`
cyan:	`rgb(0,255,255)`	`#00FFFF`
magenta:	`rgb(255,0,255)`	`#FF00FF`
yellow:	`rgb(255,255,0)`	`#FFFF00`

2 Grundformen

Linie

`<line ... />`

Attribute:

- `x1="`*Länge*`"`: x-Koordinate des Startpunkts
- `y1="`*Länge*`"`: y-Koordinate des Startpunkts
- `x2="`*Länge*`"`: x-Koordinate des Endpunkts
- `y2="`*Länge*`"`: y-Koordinate des Endpunkts
- `stroke="`*Farbe*`"`: Linienfarbe
- `stroke-width="`*Länge*`"`: Linienbreite

Kreis

`<circle ... />`

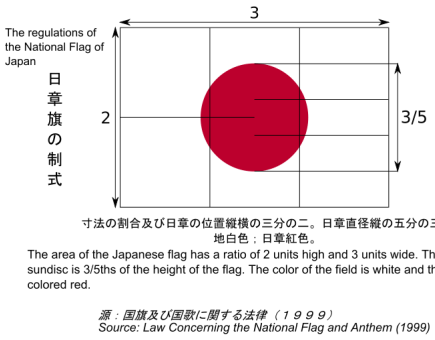
Attribute:

- `cx="`*Länge*`"`: x-Koordinate des Mittelpunkts
- `cy="`*Länge*`"`: y-Koordinate des Mittelpunkts
- `r="`*Länge*`"`: Kreisradius
- `stroke="`*Farbe*`"`: Linienfarbe
- `stroke-width="`*Länge*`"`: Linienbreite
- `fill="`*Farbe*`"`: Linienfarbe

`stroke="none"` bzw. `fill="none"` verhindert das Zeichnen des Randes bzw. das Ausfüllen der Figur.

Aufgabe 1

Erzeuge eine Grafik der Flagge Japans mit folgenden Angaben:



Rechteck

`<rect ... />`

Attribute:

- `x="`*Länge*`"`: x-Koordinate der linken oberen Ecke
- `y="`*Länge*`"`: y-Koordinate der linken oberen Ecke
- `width="`*Länge*`"`: Breite des Rechtecks
- `height="`*Länge*`"`: Höhe des Rechtecks
- `stroke="`*Farbe*`"`: Linienfarbe
- `stroke-width="`*Länge*`"`: Linienbreite
- `fill="`*Farbe*`"`: Linienfarbe

`stroke="none"` bzw. `fill="none"` verhindert das Zeichnen des Randes bzw. das Ausfüllen der Figur.

Aufgabe 2

Erzeuge eine Grafik der Flagge Schottlands mit folgenden Angaben:

- Seitenverhältnis: 3 : 5
- Breite der Kreuzbalken: ein Fünftel der Flaggenhöhe
- Farbe des Andreaskreuzes: weiss
- Hintergrundfarbe: rgb(0,110,182) bzw. #006EB6

Ellipse

`<ellipse ... />`

Attribute:

- `cx="`*`Länge`*`"`: x-Koordinate des Mittelpunkts
- `cy="`*`Länge`*`"`: y-Koordinate des Mittelpunkts
- `rx="`*`Länge`*`"`: Länge der horizontalen Halbachse
- `ry="`*`Länge`*`"`: Länge der vertikalen Halbachse
- `stroke="`*`Farbe`*`"`: Linienfarbe
- `stroke-width="`*`Länge`*`"`: Linienbreite
- `fill="`*`Farbe`*`"`: Linienfarbe

`stroke="none"` bzw. `fill="none"` verhindert das Zeichnen des Randes bzw. das Ausfüllen der Figur.

Offene Linienzüge

`<polyline ... />`

Attribute:

- `points="`*`Liste von Punkten`*`"`: Folge der Eckpunkte
- `stroke="`*`Farbe`*`"`: Linienfarbe
- `stroke-width="`*`Länge`*`"`: Linienbreite
- `fill="`*`Farbe`*`"`: Linienfarbe

`stroke="none"` bzw. `fill="none"` verhindert das Zeichnen des Randes bzw. das Ausfüllen der Figur.

x- und y-Koordinate eines Eckpunktes werden durch ein Komma getrennt; die Punkte selbst durch (mindestens) ein Leerzeichen.

Beispiel: `points="0,0 20,0 20,20, 0,20"`

Geschlossene Linienzüge

`<polygon ... />`

Attribute:

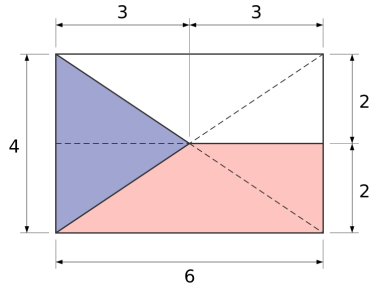
- `points="`*`Liste von Punkten`*`"`: Folge der Eckpunkte
- `stroke="`*`Farbe`*`"`: Linienfarbe
- `stroke-width="`*`Länge`*`"`: Linienbreite
- `fill="`*`Farbe`*`"`: Linienfarbe

`stroke="none"` bzw. `fill="none"` verhindert das Zeichnen des Randes bzw. das Ausfüllen der Figur.

Die Folge der Eckpunkte wird wie bei den offenen Linienzügen definiert. Der erste und der letzte Punkt werden automatisch verbunden.

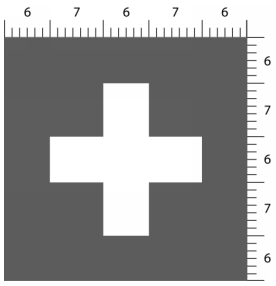
Aufgabe 3

Erzeuge eine Grafik der Flagge Tschechiens mit den folgenden Abmessungen:



Aufgabe 4

Erzeuge eine Grafik der Schweizerfahne mit den Abmessungen, wie sie im *Bundesgesetz über den Schutz des Schweizerwappens und anderer öffentlicher Zeichen* angegeben sind:



3 Pfade

Mit dem `<path />`-Element können allgemeine Linien und Kurvenformen erzeugt werden.

- `d="`*`Liste von Kommandos und Parametern`*`"`: Pfaddaten
- `stroke="`*`Farbe`*`"`: Linienfarbe
- `stroke-width="`*`Länge`*`"`: Linienbreite
- `fill="`*`Farbe`*`"`: Linienfarbe

Pfaddaten

- `M` *`x y`*: Startpunkt des Pfades
- `L` *`x y`*: gerade Linie zu (x, y)
- `C` *`x1 y1 x2 y2 x y`*: kubische Bézier-Kurve zu (x, y) mit den Kontrollpunkten (x_1, y_1) und (x_2, y_2)
- `Q` *`x1 y1 x y`*: quadratische Bézier-Kurve zu (x, y) mit dem Kontrollpunkt (x_1, y_1)
- `Z`: Ende des Pfades (ohne Angabe eines Punkts)

Es gibt noch Pfadkommandos für horizontale Linien (`H`), vertikale Linien (`V`), elliptische Bogenelemente (`A`) und Kurzformen für das Fortsetzen von kubischen (`S`) bzw. quadratischen (`T`) Bézierkurven.

4 Transformationen

Mit dem Attribut

`transform="`*`Liste von Transformationen`*`"`

werden die Transformationen in der angegebenen Reihenfolge auf das grafische Objekt angewendet. Dabei sind:

- `translate(x y)`: Verschiebung um (x, y)
- `scale(kx ky)`: Skalierung mit kx und ky
- `rotate(w x y)`: Drehung um Winkel w um Punkt (x, y)
- `skewX(w)`: Scherung um Winkel w in x-Richtung
- `skewY(w)`: Scherung um Winkel w in y-Richtung

Aufgabe 5

Erzeuge eine Grafik mit einem zweidimenisionalen Bild eines Hauses und führe damit einige der oben aufgezählten Transformationen durch.

5 Weiterführende Information

In diesem Dokument sind bei Weitem nicht alle Möglichkeiten von SVG dokumentiert. Unter den folgenden Webadressen findet man zusätzliche Informationen.

- `https://de.wikibooks.org/wiki/SVG`
- `https://www.w3schools.com/graphics/svg_intro.asp`
- `https://www.w3.org/2002/Talks/www2002-svgtut-ih/hwtut.pdf`
- `http://svg.tutorial.aptico.de/`
- `https://www.w3.org/TR/SVG11/` (die Spezifikation des W3C)