

Scalable Vector Graphics

aus Wikipedia, der freien Enzyklopädie

Scalable Vector Graphics (**SVG**, deutsch *Skalierbare Vektorgrafiken*) ist ein Standard zur Beschreibung zweidimensionaler Vektorgrafiken in der XML-Syntax. SVG wurde im September 2001 vom W3C als Empfehlung veröffentlicht und ein Großteil des Sprachumfangs kann von den meistverwendeten Webbrowsern (prominentes Gegenbeispiel ist der Internet Explorer) von Haus aus dargestellt werden. Beim Internet Explorer ist die Darstellung durch ein Plug-in wie den SVG-Viewer von Adobe möglich.

Animationen werden von SVG mittels SMIL unterstützt. Manipulationen des SVG-DOM sind mit Hilfe eingebetteter Funktionen via Skriptsprachen möglich. Zurzeit wird die Kombination und Integration von Standards des W3C untereinander in der Arbeitsgruppe Compound Document Formats erarbeitet.

Da SVG ein XML-basiertes Dateiformat ist, können SVG-Dateien mit Hilfe eines Texteditors bearbeitet werden. Dies bedeutet, dass die Texte, die in SVG-Dateien verwendet werden, auch für gegebenenfalls erforderliche Übersetzungen mit CAT-Programmen leichter zugänglich sind. Es gibt jedoch auch spezielle Programme zur Bearbeitung, zum Beispiel die freien Vektorgrafik-Programme Sodipodi und Inkscape. Die empfohlene Dateierweiterung ist *.svg* oder, wenn die Datei mit gzip komprimiert ist, *.svgz*. Der MIME-Typ ist `image/svg+xml`.^[1]



verschiedene Graustufen in SVG
Der komplette Quelltext:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg xmlns="http://www.w3.org/2000/svg"
width="1000"
height="600"
viewBox="0 0 5 5">
  <rect id="black_stripe"
fill="#000"
width="5"
height="1"/>
  <rect id="gray_i"
fill="#444"
width="5"
height="1" y="1"/>
  <rect id="gray_ii"
fill="#888"
width="5"
height="1" y="2"/>
  <rect id="gray_iii"
fill="#ccc"
width="5"
height="1" y="3"/>
  <rect id="white"
fill="#fff"
width="5"
height="1" y="4"/>
</svg>
```

Inhaltsverzeichnis

- 1 Geschichte
- 2 Struktur
- 3 Koordinatensystem und -angabe
- 4 Elemente
 - 4.1 Grafische Primitiven von SVG
 - 4.1.1 Pfad
 - 4.1.2 Kreis
 - 4.1.3 Ellipse
 - 4.1.4 Rechteck
 - 4.1.5 Linie
 - 4.1.6 Polyline
 - 4.1.7 Polygon
 - 4.1.8 Text
 - 4.2 Animationen
 - 4.2.1 Beispiel einer Animation
 - 4.3 Grafische Effekte/Filter
 - 4.4 Scripting
 - 4.4.1 Beispiel eines Scripts

- in SVG
- 4.5 Beispiel
- 5 SVG-Unterstützung in Software
- 6 SVG-Profil
- 7 Das SVG-Logo
- 8 Konferenzen
- 9 Siehe auch
- 10 Literatur
- 11 Fußnoten
- 12 Weblinks

Geschichte

Seit August 1998 arbeitete das World Wide Web Consortium (W3C) an der Spezifikation der Vektorgrafiksprache SVG, von der im September 2001 die erste Empfehlung (engl. *recommendation*) unter dem Namen *Scalable Vector Graphics (SVG) 1.0 Specification* veröffentlicht wurde.

Aktuell ist die Spezifikation der Version 1.1, die seit dem September 2003 vom W3C empfohlen wird. Die SVG-Version 1.2 ist zur Zeit in Arbeit (Stand 2005) und steht aktuell als Entwurf zur Verfügung.

Struktur

SVG nutzt die Syntax der *Extensible Markup Language* (XML). Die definierte Grafik ist somit in einer Baumstruktur aus verschiedenen Elementen und diesen Elementen zugewiesenen Attributen aufgebaut.

Jede SVG-Datei beginnt, wie bei XML-basierten Sprachen üblich, mit der XML-Deklaration und der Dokumenttypdeklaration, die den benutzten Namensraum beschreibt. Dazu wird bei Letzterem ein Verweis auf die entsprechende DTD-Datei eingefügt. Darauf folgt das Start-Tag `<svg>`. Die SVG-Datei wird durch das Tag `</svg>` abgeschlossen.

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN" "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">

<svg xmlns="http://www.w3.org/2000/svg"
      xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:ev="http://www.w3.org/2001/xml-events"
      version="1.1" baseProfile="full"
      width="800mm" height="600mm">

<!--Inhalt der Datei -->

</svg>
```

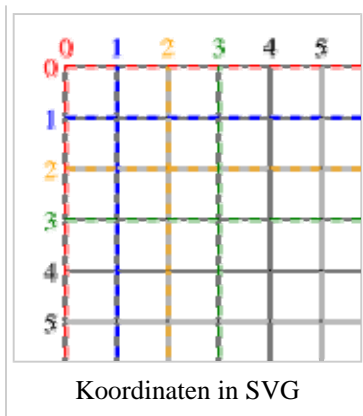
Es wird gelegentlich empfohlen, auf die Angabe des Dokumenttyps zu verzichten, da diese bei der Weiterverarbeitung unnötige Fehler produzieren kann. Ein Beispiel ist die Verwendung von RDF in den Metaangaben, die in der Spezifikation explizit erlaubt, jedoch von der DTD nicht unterstützt wird.

Mit den Attributen `width` (engl.: *Breite*) und `height` (engl.: *Höhe*) des SVG-Starttags `<svg>` wird die Größe der Grafik definiert.

Koordinatensystem und -angabe

Das Koordinatensystem hat seinen Koordinatenursprung in der linken

oberen Ecke des Zeichenbereichs. Es handelt sich dabei um ein zweidimensionales kartesisches Koordinatensystem, welches so gedreht wurde, dass die X-Achse nach rechts und die Y-Achse nach



unten weist.

Im Gegensatz zu vielen anderen Grafikformaten (z. B. Bitmaps) sind SVG-Koordinaten Gleitkomma-Zahlen und nicht Ganzzahlen. Das heißt, eine Koordinate in SVG kann quasi beliebig genau sein, während der Wertebereich ganzzahliger Pixel-Angaben in Bitmaps durch die maximale Größe des Bildes beschränkt wird.



Bei bitmapbasierten Ausgabemedien, zum Beispiel einem Monitor, bezeichnet eine SVG-Angabe wie $x = 100$ $y = 200$ nicht den ganzen Bildschirmpixel, sondern die Grenze zwischen den Pixeln. Daher bedeutet z. B. eine Angabe von **viewBox="0 0 800 600"** im öffnenden SVG-Tag eine mathematische Breite von 800 Einheiten. Bei Pixelgrafiken wäre dies eine mathematische Breite von 801 Einheiten, da die Spalten 0 und 800 mitgezählt würden. Bei dünnen Linien ist hier in SVG besondere Vorsicht geboten. Eine Angabe

```
<line x1="10" y1="300" x2="100" y2="300" stroke="#000000" stroke-width="1px" />
```

erzeugt, bei einer Skalierung von 1 : 1 im SVG-Tag, auf dem Monitor nicht etwa eine schwarze horiz. Linie von 1 Pixel Breite auf der Pixelzeile 300, sondern eine dunkelgraue Linie mit 2 Pixel Breite auf den Pixelzeilen 299 und 300, weil die Linie genau auf dem Rand zwischen den beiden Zeilen liegt. Erst eine Angabe von

```
<line x1="10" y1="300.5" x2="100" y2="300.5" stroke="#000000" stroke-width="1px" />
```

führt zum gewünschten Ergebnis.

Elemente

SVG unterstützt drei grundsätzlich unterschiedliche Typen von Elementen:

- Vektorgrafiken, aufgebaut aus grafischen Primitiven
- Rastergrafiken, also gewöhnliche Bitmap-Bilder, die extern eingebunden werden können.
- Text in einer bestimmten Schriftart, die dem Render-Programm zur Verfügung stehen muss.



Grafische Primitiven von SVG

Alle grafischen Objekte in SVG bauen auf einfachen grafischen Primitiven auf. Komplexere Objekte sind dabei aus mehreren einfachen Objekten zusammengesetzt.

Pfad

Der Pfad ist das eigentliche Grundelement in SVG. Aus ihm können alle anderen Objekte (Kreise, Rechtecke, Polygone etc.) aufgebaut werden. Da das aber teilweise sehr umständlich ist, hat man diese häufigen Formen extra mit eigenen Beschreibungen versehen, die im Anschluss aufgeführt werden.

Der Pfad wird im Attribut (d) entweder durch Koordinatenpaare (absolut) oder durch Vektoren (relativ) beschrieben. Großbuchstaben werden von Koordinaten gefolgt, Kleinbuchstaben von Vektoren. Diese Koordinaten und Vektoren bilden die Anweisungen, wie der Pfad gezeichnet werden soll.

- M/m – moveto = Stift dorthin bewegen

- L/l – lineto = mit Stift dorthin zeichnen
- Q/q – quadratische Bézierkurve (zwei Wertepaare: ein Stützpunkt, ein Zielpunkt)
- C/c – kubische Bézierkurve (drei Wertepaare: zwei Stützpunkte, ein Zielpunkt)
- Z/z – Pfad schließen

Beispiel:

Der absolut beschriebene Pfad

```
<path d="M 20 20 L 18 22 C 24 28 14 25 10 40 Q 20 45 15.33 60" />
```

ist identisch mit dem relativen

```
<path d="M 20 20 l -2 2 c 6 6 -4 3 -8 18 q 10 5 5.33 20" />
```

Für horizontale und vertikale Linien existiert eine entsprechend vereinfachte Notation, für Ellipsen und Kreise ist eine spezielle Notation mit sieben Werten erforderlich.

- H/h – horizontal line = horizontale Linie zeichnen (X-Wert)
- V/v – vertical line = vertikale Linie zeichnen (Y-Wert)
- A – elliptical arc = Bogenkurve

Kreis



Der Kreis wird mindestens durch den Radius (r) definiert. Optional kann die Position des Mittelpunktes durch die Attribute cx und cy festgelegt werden. Füllung, Art der Umrisslinie oder Transparenz werden durch das Attribut style definiert.

```
<circle cx="100" cy="100" r="50" />
```

Ellipse



Die Ellipse wird durch die zwei Halbachsenradien definiert (rx und ry). Alle anderen Attribute entsprechen denen des Kreises.

```
<ellipse cx="100" cy="100" rx="50" ry="20" />
```

Rechteck



Ein Rechteck kann auf nur eine konforme Art definiert werden. Diese Möglichkeit ist die Beschreibung durch die obere linke Ecke mit den Koordinaten (x und y) und die Breite (width) und Höhe (height).

```
<rect x="100" y="100" width="100" height="200" />
```

Möchte man die Ecken des Rechtecks abrunden, kann man das mit Hilfe der Attribute (rx und ry)

bewerkstelligen.

Linie

Eine einfache gerade Linie wird durch die beiden Endpunkte in SVG beschrieben.

```
<line x1="100" y1="100" x2="200" y2="200" />
```

Polyline

Um eine Linie über mehrere Stützpunkte zu definieren, beschreibt man sie als so genannte Polylinie. Dabei werden die einzelnen Koordinatenpaare der Stützpunkte einfach fortlaufend hintereinander geschrieben. Die Linie wird vom ersten Punkt zum zweiten Punkt gezogen, von dort weiter zum dritten Stützpunkt und so weiter.

```
<polyline points="100 100 200 200 300 40" />
```

Polygon

Ein Polygon (Vieleck) wird durch seine Eckpunkte beschrieben. Wie bei der Polyline werden auch hier alle Koordinatenpaare einfach hintereinander geschrieben.

```
<polygon points="100 100 100 200 150 200" />
```

Text

Zeichen und Text können mit Hilfe des Elements <text> definiert werden.

```
<text x="100" y="100" font-size="40px">Hallo Welt!</text>
```

Animationen

Die Tags, die für die Auszeichnung von Animationen benutzt werden, stammen aus dem XML-Multimediastandard Synchronized Multimedia Integration Language (SMIL). Zustände, welche animiert werden können, sind Transformation, Position, Sichtbarkeit, Farbe und Größe. Wichtig für das Verständnis von Animationen sind die Begriffe „Darsteller“ (das zu animierende Objekt) und „Drehbuch“ (die Zeitspanne der Animation).

Zeitangaben können von Millisekunden bis Stunden angegeben und verschieden ausgezeichnet werden. Für die meisten Animationstags wird ein Pflichtattribut benötigt, das die Art der Animation festlegt. Weitere grundlegende Komponenten sind, wie schon erwähnt, der Zeitrahmen, Zustände während der Animation (optional), der Zustand nach der Animation, und ob eine Wiederholung gestartet werden soll.

Die Eigenschaften eines Darstellers sind in SVG durch folgende Attribute festgehalten.

- attributeName (was wird animiert)
- attributeType (XML-Teile oder Formatierungen in CSS)

Für die Zeitangabe werden logische Begriffe wie *begin* (engl.: *Start, Beginn*) *end* (engl.: *Ende*) und *dur* (engl. „duration“: *Dauer*) verwendet. Start, Wiederholungen etc. können auch von Ereignissen wie

Mausberührung oder -klick abhängig gemacht werden.

Beispiel einer Animation

Quellcode der Pfadanimation:

```
<path id="Pfad" d="M50,100 C50,50 125,50 125,100 S200,150 200,100" fill="orange" stroke="black"
stroke-width="3px" stroke-linecap="round" />
<circle cx="-10" cy="-2.5" r="5" fill="yellow" stroke="white">
  <animateMotion id="kreis_ani" dur="5s" repeatCount="indefinite" begin="play.click" end="stop.click" r
  <mpath xlink:href="#Pfad" />
</animateMotion>
</circle>
```

Quellcode des Play-Buttons:

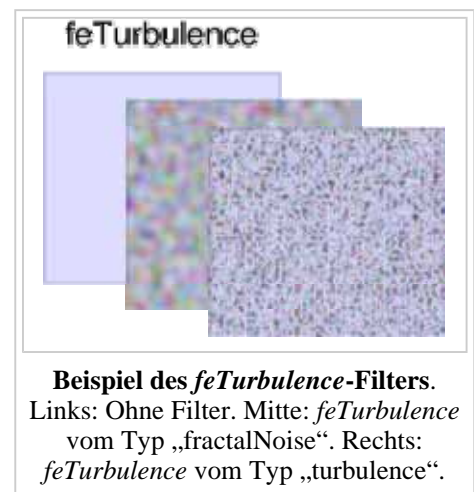
```
<polygon id="play" xlink:href="#kreis_ani" points="190,230 190,210 210,220" fill="#ff00" opacity="0.5">
  <set attributeName="opacity" attributeType="XML" to="1.0" begin="mouseover"/>
  <set attributeName="stroke" attributeType="XML" to="#fff" begin="mouseover"/>
  <set attributeName="opacity" attributeType="XML" to="0.5" begin="mouseout"/>
  <set attributeName="stroke" attributeType="XML" to="none" begin="mouseout"/>
</polygon>
```

Grafische Effekte/Filter

Bekannte Grafikfilter sind aus DTP-Programmen übernommen worden und arbeiten nach dem Matrizen-Prinzip, bei dem Operatoren und Funktionen die Form der Matrix ausdrücken. Es wird jede einzelne Bildposition mit dem definierten Filtereffekt berechnet und auf die nächste übertragen. Auf diese Weise lassen sich Bilder und Grafiken auf verschiedenste Art und Weise bearbeiten.

Einige der möglichen Filter:

- feTurbulence
- feDisplacement
- feGaussianBlur



Scripting

Scripting ist in SVG mittels ECMAScript (einer standardisierten Variante von JavaScript) möglich. Mit dem Document Object Model von SVG (SVG-DOM) kann man die XML-Struktur einer Grafik manipulieren und so zum Beispiel neue Elemente per Mausklick erschaffen.

Beispiel eines Scripts in SVG

Quellcode mit implementiertem ECMAScript-Code

```

<?xml version="1.0" encoding="utf-8"?>
<svg xmlns="http://www.w3.org/2000/svg"
      xmlns:xlink="http://www.w3.org/1999/xlink"
      xmlns:ev="http://www.w3.org/2001/xml-events"
      version="1.1" baseProfile="full"
      width="350" height="200">
  <defs><!-- Definitionsbereich, hier mit Javascript -->
    <script type="text/ecmascript"><![CDATA[
      function wachsen(e)
      {
        var re = e.target;

        var breite = re.getAttributeNS(null, 'width');
        breite = 1 + 1*breite ;
        if (breite > 250)
          breite = 10;

        re.setAttributeNS(null, 'width', breite);
      }
    ]]></script>
  </defs>
  <rect id="rechteck1" x="20" y="30" width="10" height="60"
        fill="blue" onmousemove="wachsen(evt)" cursor="pointer"/>
  <rect id="rechteck2" x="20" y="95" width="30" height="60"
        fill="red" onmousemove="wachsen(evt)" cursor="pointer"/>
</svg>

```

Im Beispiel wird eine JavaScript-Funktion `wachsen()` definiert, die bei Aufruf das betroffene Element um einen Pixel verbreitert. In dieser Funktion wird das Attribut `width` mit der Funktion `getAttributeNS()` ausgelesen, geändert und anschließend mit `setAttributeNS()` neu gesetzt. Nach dem Definitionsbereich werden zwei Rechtecke definiert. Jeweils im selben Tag wird der Eventhandler `onmousemove` an die Funktion `wachsen(evt)` geknüpft.

Das Ergebnis sind zwei farbige Fortschrittbalken, die wachsen, wenn man sie mit der Maus streichelt.

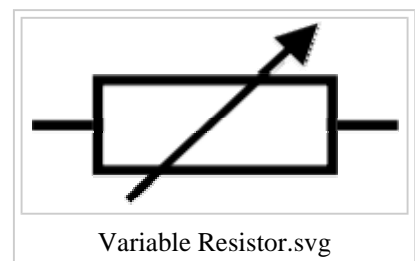
Beispiel

Ein kleines Beispiel illustriert die Verwendung der einzelnen Elemente gut. Das rechts dargestellte Bild `Variable Resistor.svg` (Schaltbild eines Potentiometers) hat den folgenden kommentierten Quelltext:

```

<?xml version="1.0" encoding="utf-8"?>
<svg xmlns="http://www.w3.org/2000/svg"
      xmlns:xlink="http://www.w3.org/1999/xlink"
      xmlns:ev="http://www.w3.org/2001/xml-events"
      version="1.1" baseProfile="full"
      width="107" height="60" viewBox="-2 -5 105 55">
  <!-- Anschlüsse links und rechts -->
  <line x1="0" y1="25" x2="100" y2="25" fill="none" stroke="black"
        stroke-width="3px"/>
  <!-- Das Widerstandsrechteck -->
  <rect x="10" y="15" width="80" height="20" fill="white" stroke="black"
        stroke-width="3px" />
  <!-- Der Schleifer -->
  <polyline points="65 5 40 40 40 50" fill="none" stroke="black"
            stroke-width="3px"/>
  <!-- Die Pfeilspitze wird gedreht -->
  <polygon points="60 5 70 5 65 -5" stroke="black" stroke-width="3px"
           transform="rotate(33.7 65 5)" />
</svg>

```



SVG-Unterstützung in Software

Viele Desktop-Environments benutzen zunehmend SVG unter anderem als Format für

Programmsymbole, Hintergrundbilder, Mauszeiger, weil SVG sich ohne Qualitätsverluste frei skalieren lässt. Bekannte Projekte sind das Oxygen Project^[2] für KDE und das Tango Desktop Project^[3] für GNOME. Generell ist zu bemerken, dass viele Betriebssysteme auf ihren grafischen Oberflächen vermehrt Vektorgrafiken benutzen; der zusätzliche Ressourcenverbrauch zum Rendern wird dabei durch steigende Hardwareressourcen kompensiert oder sogar schon von den Grafikkarten übernommen.

Programme, mit denen man SVG-Dateien erstellen und anzeigen kann, sind u. a.:

- Adobe Illustrator (proprietär, Windows/Mac OS)
- Batik (Apache License, plattformunabhängig durch Nutzung von Java)
- Canvas X nur Export (proprietär, Windows/Mac OS)
- CorelDRAW (proprietär, Windows)
- FreeHand (proprietär, Windows/Mac OS)
- Gnuplot (frei, jedoch kein GPL^[4])
- Geonext Export (frei, Linux/Windows/Mac OS)
- Graphviz Graphendarstellung (plattformunabhängig)
- Inkscape (GPL, Linux/Windows/Mac OS)
- Karbon14 und nur Export KChart (Bestandteile von KOffice)
- Microsoft Visio (proprietär, Windows)
- OpenOffice.org (LGPL, Linux/Windows/Solaris): Export nativ mit Einschränkungen, Import mit Plug-in
- Skencil (LGPL, Linux/Mac OS/Solaris/FreeBSD)
- Sketsa (proprietär, plattformunabhängig)
- Sodipodi (GPL, Linux/Windows, Entwicklung eingestellt)
- VectorMagic (proprietär, plattformunabhängige Flash-Anwendung)
- Xara Xtreme for Linux (GPL unter Linux, proprietär unter Windows)
- 3D Flash Animator (proprietär, Windows)

SVG-Unterstützung in Browsern

SVG wurde in erster Linie für das World Wide Web entworfen. Viele WWW-Browser können es ohne weiteres darstellen, d. h. sie benötigen kein SVG-Zusatzprogramm (Browser-Plug-in).

Betriebssystem	Browser	Plugin	Grundformen	Animationen	Animationen mit abs. Zeitangabe (Wallclock)	Scripting
Windows	Internet Explorer 7	nativ	keine			
		Adobe SVG-Viewer 3	vollständig	vollständig	keine	vollständig
	Internet Explorer 6	nativ	keine			
		Adobe SVG-Viewer 3	vollständig	vollständig	keine	vollständig
	Mozilla 1.7.x/Firefox 1.0.x	nativ	keine			
		Adobe SVG-Viewer 3	instabil	instabil	keine	instabil

	Firefox 1.5.x	nativ	größtenteils	keine	keine	vollständig
	Firefox 2.0.x	nativ	größtenteils	keine	keine	vollständig
	SeaMonkey 1.0	nativ	größtenteils	keine	keine	vollständig
	Opera 8	nativ	teilweise	vollständig	keine	vollständig
	Opera 9	nativ	vollständig	vollständig	keine	vollständig
	Amaya	nativ	vollständig	vollständig	keine	vollständig
Linux	Konqueror	KSVG-KPart	vollständig	teilweise	keine	vollständig
	Mozilla 1.7.x/Firefox 1.0.x	nativ	keine			
		Adobe SVG-Viewer 3	keine			
	Firefox 1.5.x	nativ	größtenteils	keine	keine	vollständig
	Firefox 2.0.x	nativ	größtenteils	keine	keine	vollständig
	Opera 8 / 9	nativ	vollständig	vollständig	keine	vollständig
	Amaya	nativ	vollständig	vollständig	keine	vollständig
Mac OS X	Safari 3.0 (beta 521.32.1)	nativ	vollständig	keine	keine	vollständig
	Safari 2.0	nativ	keine			
		WebKit [1]	vollständig	keine	keine	vollständig
	Safari 1.3	nativ	keine			
		Adobe SVG-Viewer 3	vollständig	vollständig	keine	vollständig
	Firefox 1.5.x, Camino 1.0	nativ [2]	größtenteils	keine	keine	vollständig
	Mozilla 1.7.x/Firefox 1.0.x, Camino 0.x	nativ	keine			
		Adobe SVG-Viewer 3	instabil	instabil	keine	instabil
	Amaya	nativ	vollständig	vollständig	keine	vollständig

(Tabelle Stand August 2006, Angaben Mac OS X bedürfen noch einer Bestätigung/Vervollständigung)

Die US-amerikanische Softwarefirma Adobe Systems entwickelt seit einigen Jahren einen Viewer für

SVG, der in verschiedenen Browsern auf mehreren Plattformen genutzt werden kann. Über die aktuelle Version hinaus existiert eine Betaversion des Adobe SVG-Viewers in der Version 6 (Preview 1, mit Datum Juli 2003), die eine verbesserte Unterstützung der Standards und Verbesserungen bei der Integration in Mozilla Firefox beinhaltet. Das Projekt scheint jedoch seit Sommer 2003 zu ruhen, mittlerweile hat Adobe offiziell bekanntgemacht, dass das Projekt mangels Gewinnaussichten eingestellt wurde. 2006 teilte Adobe mit, den Support des SVG-Viewers zum 1. Januar 2008 einzustellen.^[5]^[6] Beobachter gehen davon aus, dass diese Entscheidung durch die Akquise von Macromedia durch Adobe zusammenhängt. SVG in Verbindung mit SMIL wird als Konkurrenz zum proprietären Flash angesehen, für das Macromedia Bearbeitungswerkzeuge herstellte.

Als gute Alternative ist der Renesis Player zu sehen, an dem derzeit mit Hochdruck gearbeitet wird.

SVG-Profile

In SVG gibt es 3 Profile, die für die Darstellung auf unterschiedlichen Ausgabegeräten abgestimmt sind. Sie werden im Wurzelement (SVG) mit dem Attribut *baseProfile* angegeben. Sie sind abwärtskompatibel, da manche Elemente nicht implementiert sind.

- SVGF (*baseProfile*="full") ist für Computer als Ausgabegerät gedacht und unterstützt SVG komplett
- SVGB (*baseProfile*="basic") ist für PDAs
- SVGT (*baseProfile*="tiny") ist für Mobiltelefone

Der komplette Funktionsumfang für SVGT und SVGB ist hier aufgelistet.

Das SVG-Logo

Um ein einheitliches „Markenzeichen“ für SVG zu schaffen, wurde von SVG.org, unterstützt u. a. von Sun und Opera, ein Logowettbewerb ins Leben gerufen. Bis zum 1. Oktober 2006 waren Designer aufgerufen, eigene Vorschläge für ein SVG-Logo einzureichen. Dieses soll in Software und auf Webseiten, die SVG verwenden, eingesetzt werden. Am 17. Dezember 2006 wurde der Gewinner bekannt gegeben.



Konferenzen

Die SVG Open ist eine internationale Konferenz, die sich mit allem rund um das Thema SVG beschäftigt. Mittlerweile hat sie bereits fünf Mal stattgefunden.

- Zürich, Schweiz, 15.–17. Juli 2002
- Vancouver, Kanada, 13.–18. Juli 2003
- Tokio, Japan, 7.–10. September 2004
- Enschede, Niederlande, 15.–18. August 2005
- Tokio, Japan, 4.–7. September 2007

Im Jahr 2006 fand die SVG Open nicht statt. Die ursprünglich für den Zeitraum vom 16.–19. Oktober in Victoria (Kanada) geplante Konferenz wurde wegen der plötzlichen Erkrankung des Hauptorganisators abgesagt.

Siehe auch

- Adobe Flash – proprietäre Entwicklungsumgebung zur Erstellung multimedialer Inhalte

Literatur

- Alexander Adam, *SVG Scalable Vector Graphics*, Verlag FRANZIS, ISBN 3-7723-6190-0
- Marcel Salathé, *SVG Scalable Vector Graphics, ...für professionelle Einsteiger*, Verlag Markt+Technik, ISBN 3-8272-6188-0

Fußnoten

1. ↑ <http://www.w3.org/TR/SVG/intro.html#MIMEType>
2. ↑ siehe auch: „Oxygen Project“ in der englischen Wikipedia; Homepage der Entwickler
3. ↑ siehe auch: „Tango Desktop Project“ in der englischen Wikipedia; offizielle Homepage
4. ↑ <http://gnuplot.cvs.sourceforge.net/gnuplot/gnuplot/Copyright?view=markup>
5. ↑ <http://www.adobe.com/svg/viewer/install/mainframed.html>
6. ↑ <http://www.golem.de/0609/47675.html>

Weblinks



Commons: SVG-Beispiele – Bilder, Videos und Audiodateien



Wikibooks: SVG – Lern- und Lehrmaterialien

- Scalable Vector Graphics (SVG) Specification beim World Wide Web Consortium (englisch)
- Scalable Vector Graphics Tiny (SVGT) Specification beim World Wide Web Consortium (englisch)
- Links zum Thema SVG im Open Directory Project
- Tutorial zum Thema SVG mit Beispieldateien

Von „http://de.wikipedia.org/wiki/Scalable_Vector_Graphics“

Kategorien: [Computergrafik](#) | [Grafikformat](#) | [Beschreibungssprache](#) | [XML-basierte Sprache](#)

- Diese Seite wurde zuletzt am 17. Dezember 2007 um 18:40 Uhr geändert.
- Ihr Inhalt steht unter der GNU-Lizenz für freie Dokumentation.
Wikipedia® ist eine eingetragene Marke der Wikimedia Foundation Inc.