

3 Jahre Scalable Vector Graphics @ Hochschule Merseburg – Präsentation von Forschungsergebnissen für den Zeitraum 01/2002 bis 01/2005

**Posterbeitrag / Beitrag zum Tagungsband der
6. Nachwuchswissenschaftlerkonferenz der Fachhochschulen Mitteldeutschlands
am 20.01.2005 in Merseburg**

von Dr. rer. nat. Thomas Meinike
Lehrkraft für besondere Aufgaben
Hochschule Merseburg (FH)
Fachbereich Elektrotechnik, Informationstechnik und Medien / Studiengang Kommunikation und Technische
Dokumentation
Geusaer Straße, 06217 Merseburg
Tel.: 03461/463055, Fax: 03461/462900
E-Mail: thomas.meinike@et.fh-merseburg.de
Homepage: <http://www.et.fh-merseburg.de/person/meinike/>

1. Überblick

Im September 2001 wurde vom World Wide Web Consortium (W3C) die Spezifikation für XML-basierte Vektorgrafiken mit der Bezeichnung „Scalable Vector Graphics“ (SVG) in der Version 1.0 verabschiedet. SVG 1.1 folgte im Januar 2003 und Mitte 2005 soll die finale Version 1.2 vorliegen [W1].

SVG stellt bekannte geometrische Grundformen wie u. a. Rechteck, Kreis und Polygon zur Verfügung. Es lassen sich aber auch komplexere Objekte durch Pfade beschreiben. Hinzu kommen mächtige Techniken wie Animationen, Transformationen, Filter, Formatierung von Inhalten mittels Cascading Stylesheets (CSS) und die Unterstützung von Skriptsprachen wie JavaScript. Einige typische SVG-Anwendungsbereiche sind Kartografie und Geo-Informationssysteme, Konstruktion / CAD und die Visualisierung wissenschaftlich-technischer Sachverhalte, z. B. für Dokumentationszwecke.

SVG-Dokumente lassen sich nicht nur statisch erzeugen, sondern auch unter Anwendung von serverseitigen Technologien wie beispielsweise PHP. Clientseitig lässt sich Dynamik und Interaktion durch Manipulationen über das Document Object Model (DOM) realisieren. SVG führt somit viele offene Web-Technologien zusammen und entwickelt sich ständig weiter. Zu nennen ist dabei vor allem der Bereich Mobile Computing mit Endgeräten wie Handys oder PDAs.

Seit 2002 ist SVG ein Lehr- und Forschungsthema im Studiengang Kommunikation und Technische Dokumentation. Im Rahmen von mehreren Diplomarbeiten wurden bisher Grundlagen studiert sowie praktische Anwendungen umgesetzt. Außerdem sind im genannten Zeitraum Publikationen entstanden, die unter Literatur und Ressourcen vermerkt sind. Im Folgenden werden einige Grundlagen, Anwendungen und weitere Ergebnisse als Ergänzung zum Posterbeitrag [P4] dargestellt.

2. Grundlagen

An dieser Stelle kann nur ein kleiner Einblick in die SVG-Spezifikationen gegeben werden, da diese in gedruckter Form mehrere hundert Seiten einnehmen. Einen guten Einstieg vermitteln die Websites [W2] und [W3]. Jedes SVG-Dokument ist nach diesem Prinzip aufgebaut:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.0//EN"
  "http://www.w3.org/TR/2001/REC-SVG-20010904/DTD/svg10.dtd">
<svg xmlns="http://www.w3.org/2000/svg">
  <title>Titel des Dokuments</title>
  <desc>Beschreibungstext</desc>
  <defs>Bereich für die Einbindung von Referenzen
    wie Style- und Skriptvorlagen, Filtern usw.</defs>

  <!-- Inhalte der Grafik -->

</svg>
```

Nach dem aus XML- und Dokumenttyp-Deklaration bestehenden Prolog folgt das Wurzelement `svg`, welches alle folgenden Elemente einschließt. Die eigentlichen Grafikinhalte werden mit weiteren Elementen und Attributen definiert (Auswahl):

```
<rect x="..." y="..." width="..." height="..." />
<circle cx="..." cy="..." r="..." />
<ellipse cx="..." cy="..." rx="..." ry="..." />
<line x1="..." y1="..." x2="..." y2="..." />
<polyline points="x1,y1 ... xn,yn" />
<polygon points="x1,y1 ... xn,yn" />
<text x="..." y="...">Textinhalt</text>
...
```

Im Detail werden u. a. Koordinaten, Längenangaben (Breiten, Höhen, Radien), Textinhalte sowie das visuelle Erscheinungsbild über Präsentationsattribute oder CSS-Definitionen (z. B. Füllfarben, Umrandungen, Schriftarten- und -größen) festgelegt. Speziellere Techniken wie Animationen oder Filter lassen sich mit weiteren SVG-Sprachkonstrukten realisieren [A1, A5, D1, D3]. Für die Darstellung der Grafiken in einem Web-Browser werden zurzeit noch separat zu installierende Plug-ins benötigt. Das bekannteste und leistungsfähigste Plug-in für verschiedene Plattformen ist der Adobe SVG Viewer (ASV), der unter [W4] zum Download für Windows, MacOS und Linux angeboten wird. Abbildung 1 zeigt eine mit SVG realisierte Übersicht zu den wichtigsten Objekten und Effekten.

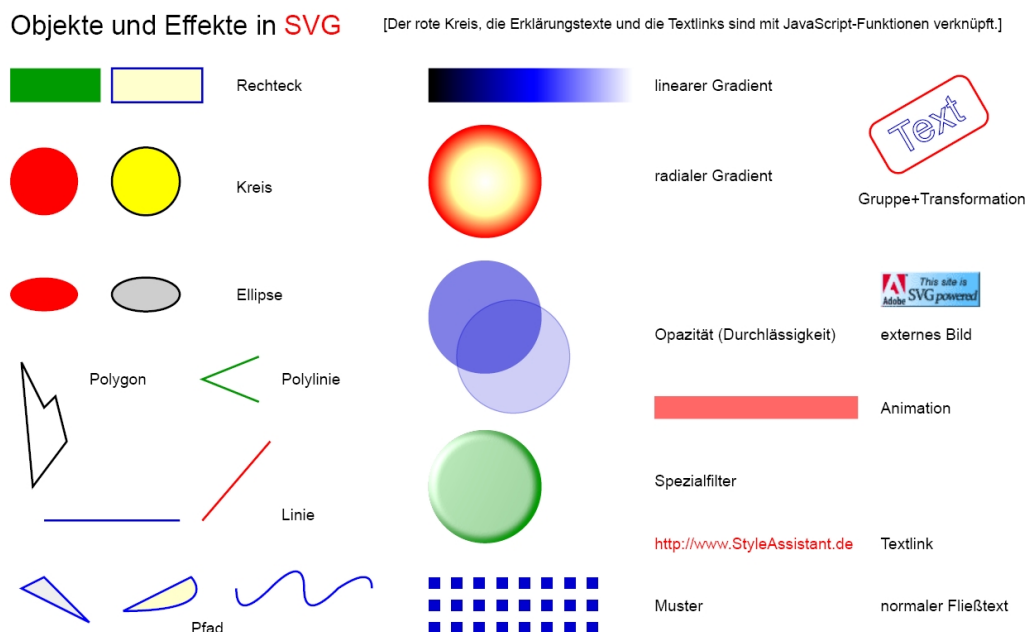


Abb. 1: Demonstration von Objekten und Effekten in SVG (online unter [W5] verfügbar).

3. Ausgewählte Anwendungen und Ergebnisse

Das in Abbildung 1 umgesetzte Dokument ist Bestandteil der Website „SVG – Learning By Coding“ [W6], die vom Autor seit 2002 betrieben wird. Insbesondere finden sich dort ca. 200 umfangreiche Beispiele mit dem dazugehörigen Sourcecode. Code und Grafiken lassen sich beim Vorhandensein eines entsprechenden SVG-Viewers direkt im Browser anzeigen (vgl. Abbildung 2).

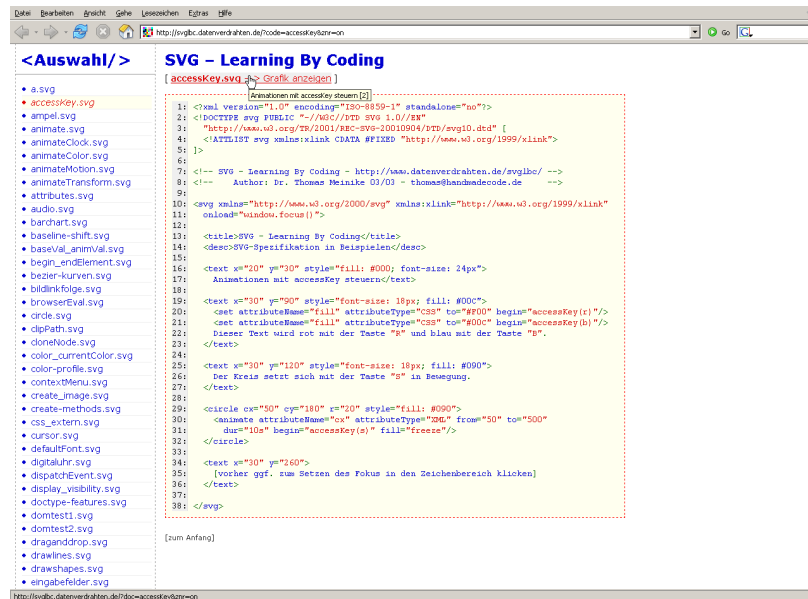


Abb. 2: Website „SVG – Learning By Coding“ (online verfügbar unter [W6]).

In der technischen Dokumentation wird XML bereits umfassend verwendet und auch das Interesse an SVG nahm in den letzten Jahren deutlich zu, wie die Resonanz auf mehrere vom Autor gehaltene Vorträge [V1] bis [V4] gezeigt hat. Im Rahmen von fünf Diplomarbeiten wurde das Thema SVG eigenständig [D1, D3-D5] oder anwendungsbezogen behandelt [D2]. Es wurden umfangreiche Erfahrungen zur praktischen Entwicklung von SVG-Anwendungen mit unterschiedlichen zusammenwirkenden Technologien gesammelt und über die genannten Vorträge hinaus in Form von Artikeln und Posterbeiträgen publiziert. Anmerkung: Einige nicht in diesem Text zitierte Arbeiten beschäftigen sich mit allgemeinen oder speziellen Grundlagen und Techniken im SVG-Umfeld.

Ausgewählte praktische Ergebnisse sind in Form von Abbildungen mit Kurzbeschreibung als Bildunterschrift aufgeführt.

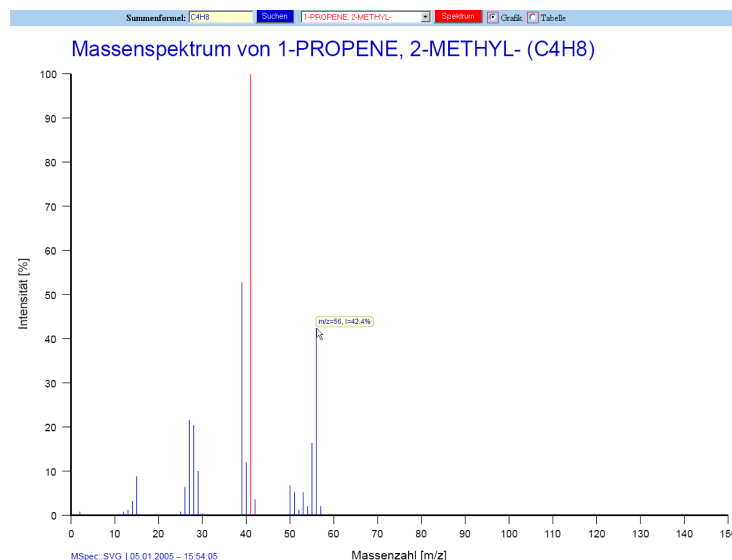


Abb. 3: Dynamisch im SVG-Format erzeugte Massenspektren für die chemische Analytik ausgehend von einer Datenbank mit den Spektrendaten zu 50.000 chemischen Verbindungen. Verwendete Technologien: CSS, HTML, JavaScript, MySQL, PHP, SVG, XML, XSLT (zu Details siehe [A2], [P1], [V2]; Anwendung online verfügbar unter [W7]).

Periodensystem der Elemente mit Online-Datenabfrage

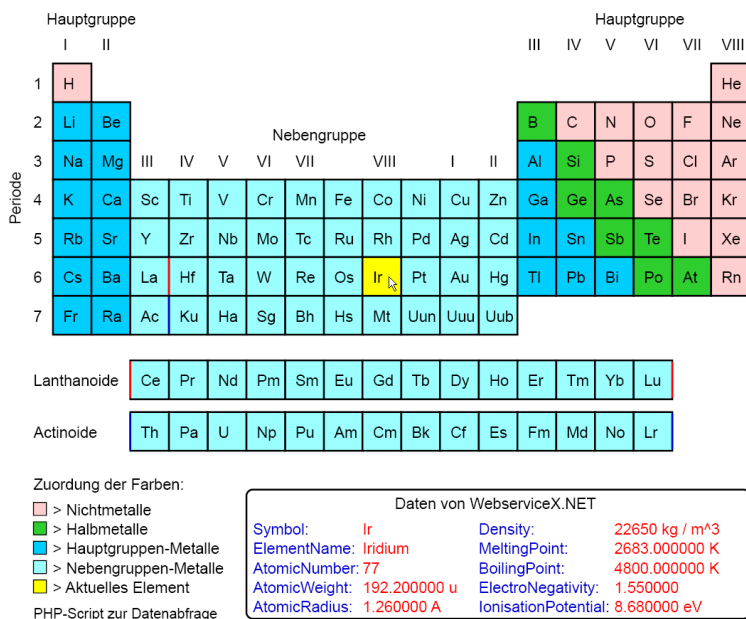
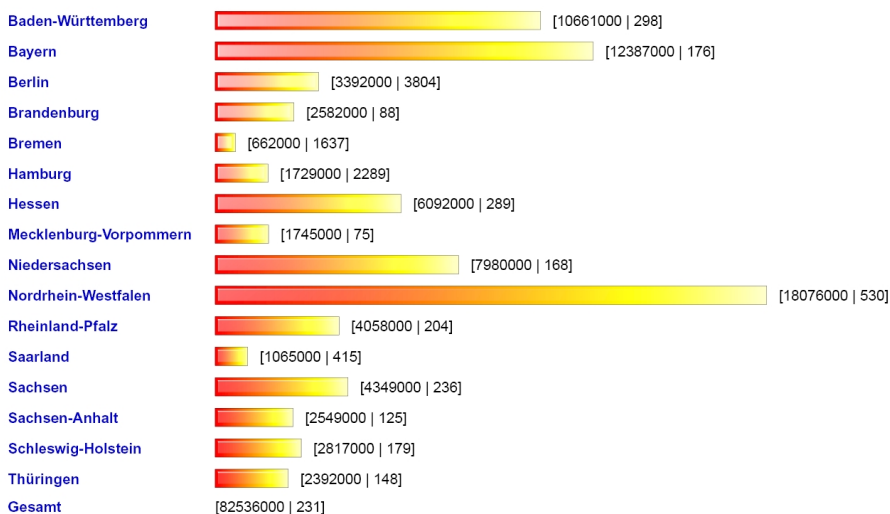


Abb. 4: Anwendung zur Demonstration des interaktiven Zugriffs auf von einem XML-Webservice gelieferte Daten aus einem SVG-Dokument heraus. Verwendete Technologien: CSS, HTML, JavaScript, PHP, SVG, XML (zu Details siehe [A7], [P2], [V3]; Anwendung online verfügbar unter [W8]).

Einwohnerzahlen am Stichtag 31.12.2002

Legende: [Einwohner gesamt | Einwohner pro km²]



Diese Grafik wurde aus XML-Daten mittels XSLT erzeugt.
Datenquelle: Statistisches Bundesamt Deutschland

Abb. 5: Visualisierung von XML-Daten mittels SVG über XSL-Transformationen und Ausgabe von Rastergrafiken (JPEG, PNG) für nicht-SVG-fähige Browser. Verwendete Technologien: CSS, HTML, Java (Batik-Rasterizer), SVG, XML, XSLT (zu Details siehe [A8], [P3]; Anwendung online verfügbar unter [W9]).



Abb. 6: Interaktive SVG-Kartenanwendung mit Datenbankabfrage. Verwendete Technologien: CSS, HTML, JavaScript, MySQL, PHP, SVG (zu Details siehe [D5]; Anwendung online verfügbar unter [W10]).

4. Zusammenfassung und Ausblick

In den letzten Jahren hat sich das Thema SVG zu einem Arbeitsschwerpunkt des Autors entwickelt. Mit den hier dokumentierten Ergebnissen soll auch die an Fachhochschulen durchaus mögliche Beschäftigung mit innovativen Technologien über den Lehralltag hinaus unterstrichen werden. Die bisher gesammelten Erfahrungen bilden die Basis für künftige Projekte. Gegenwärtig befindet sich die viel versprechende SVG-Version 1.2 im Stadium eines Working Drafts, der in [D4] und [V4] näher betrachtet wurde. Weitere Studien- und Diplomarbeiten befinden sich in Bearbeitung bzw. Vorbereitung. Ein wichtiges Thema ist dabei der Einsatz von SVG auf mobilen Endgeräten. Aktuelle Handys sind mit entsprechender Anzeigesoftware ausgestattet und bieten somit Potenzial für die Erschließung und Umsetzung von neuen Anwendungen. Abbildung 7 zeigt eine Grafik im SVG-Tiny-Format auf dem S65-Handy von SIEMENS.



Abb. 7: SVG auf einem aktuellen Handy-Modell (Anwendung online verfügbar unter [W11]).

Da SVG noch ein relativ neues Arbeitsgebiet ist und die bisherigen Anwendungen sich nur auf wenige Bereiche konzentrieren, wäre es für die Zukunft wünschenswert, mit weiteren Partnern aus der Industrie und anderen Hochschulen zusammen zu arbeiten. Eine entsprechende Kontaktaufnahme mit dem Autor dieses Beitrages ist stets willkommen.

5. Literatur und Ressourcen

Hinweis: Zu einigen Artikeln, Postern und Vorträgen stehen auf der Website des Autors [W12] HTML- oder PDF-Versionen zur Verfügung.

Artikel in Fachzeitschriften [A]

- [A1] Meinike, T.: „Grafik-Tagwerk – Statische und dynamische SVG-Grafiken erstellen“; Internet Professionell 7/2002, S. 77-79.
- [A2] Meinike, T.: „Technologie-Mix [MSpec::SVG – Generierung von Massenspektren im SVG-Format]“; XML & Web Services Magazin 02.2002, S. 68-72.
- [A3] Meinike, T.: „TESTDRIVE XStudio Next 1.1 (SVG-Authoring-Lösung)“; Internet Professionell 04/2003, S. 25.
- [A4] Meinike, T.: „SVG ruft Datenbank – Anwendung von Zusatzfunktionen des Adobe SVG Viewers“; XML & Web Services Magazin 03.2003, S. 66-69 und PHP Magazin 04.2003, S. 79-81.
- [A5] Meinike, T.: „Mobile Vektoren – Animationen und Effekte mit SVG“; Internet Professionell 06/2003, S. 72-75.
- [A6] Meinike, T.: „Operation Matrix – Anwendung von Matrizen-Techniken bei SVG-Transformationen“; XML & Web Services Magazin 05.2003, S. 29-32 und PHP Magazin 01.2004, S. 79-82.
- [A7] Meinike, T.: „Chemiestunde – Entwicklung von dynamischen Datenzugriffstechniken für Scalable Vector Graphics“; XML & Web Services Magazin 01.2004, S. 42-44.
- [A8] Meinike, T.: „Auto-Grafik – XML-Daten grafisch darstellen“; Internet Professionell 05/2004, S. 69-71.
- [A9] Meinike, T.: „Rezension zu ‚SVG Reporting‘“; XML & Web Services Magazin 03.2004, S. 11 und dot.net magazin 06.2004, S. 14.
- [A10] Meinike, T.: „Bitte um Auskunft – Analysetool zur Ermittlung von Eigenschaften und Methoden von SVG-Objekten“; XML & Web Services Magazin 03.2004, S. 80-81.
- [A11] Meinike, T.: „Vektordynamik – SVG-Programmierung mit JavaScript“; Internet Professionell 01/2005, S. 82-85.

Betreute Diplomarbeiten an der HS Merseburg mit Bezug zu SVG [D]

- [D1] Plitz, M.: „Scalable Vector Graphics (SVG) – Erstellung und Anwendung XML-basierter Vektorgrafiken“ (2002, abgeschlossen); Betreuer: Dr. T. Meinike / Prof. F. Hofmann.
- [D2] Gräfe, A.: „Visualisierung in LON-Netzwerken“ (2004, abgeschlossen); Betreuer: Prof. Dr. F. Sokollik / Dr. T. Meinike.
- [D3] Becke, K.: „Techniken zur Verarbeitung und Publikation von Scalable Vector Graphics“ (2004/05, eingereicht); Betreuer: Dr. T. Meinike / Prof. F. Hofmann.
- [D4] Menzel, U.: „Scalable Vector Graphics 1.2 – Neuerungen und Anwendungen“ (2004/05, eingereicht); Betreuer: Dr. T. Meinike / Prof. F. Hofmann.
- [D5] Boche, M.: „Scalable Vector Graphics – Realisierung von Dynamik und Interaktivität“ (2004/05, eingereicht); Betreuer: Dr. T. Meinike / Prof. F. Hofmann.

Posterpräsentationen [P]

- [P1] Meinike, T.: „Realisierung eines XML-Workflows von der Datenhaltung bis zur Visualisierung – Projekt zur Demonstration des Zusammenwirkens von Web-Technologien“; Tag der Forschung an der FH Merseburg in Merseburg (13.11.2003).
- [P2] Meinike, T.: „Entwicklung von dynamischen Datenzugriffstechniken für Scalable Vector Graphics – Projekt zur Programmierung von XML-basierten Client-Server-Anwendungen“; Tag der Forschung an der FH Merseburg in Merseburg (13.11.2003).

[P3] Meinike, T.: „Visualisierung von XML-Daten unter Verwendung von XSLT und SVG – Projekt zur Demonstration des Zusammenwirkens von XML-Technologien“; Tag der Forschung an der FH Merseburg in Merseburg (11.11.2004).

[P4] Meinike, T.: „3 Jahre Scalable Vector Graphics @ Hochschule Merseburg – Präsentation von Forschungsergebnissen für den Zeitraum 01/2002 bis 01/2005“; 6. Nachwuchswissenschaftlerkonferenz in Merseburg (20.01.2005).

Vorträge auf Konferenzen [V]

[V1] Meinike, T.: „Scalable Vector Graphics (SVG) – ein XML-basierter Grafikstandard für 2D-Vektorgrafiken“; Tag der Forschung an der FH Merseburg in Merseburg (30.05.2002).

[V2] Meinike, T.: „'On-the-fly'-Generierung von SVG-Dokumenten auf dem Webserver“. In: tekomp, Gesellschaft für technische Kommunikation e. V., Tagungsband zur Jahrestagung 2002 in Wiesbaden, S. 52-55 (20.11.2002).

[V3] Meinike, T.: „SVG-Aktionsprogrammierung – mit DOM-Methoden vom Ereignis zum Effekt“. In: tekomp, Gesellschaft für technische Kommunikation e. V., Tagungsband zur Jahrestagung 2003 in Wiesbaden, S. 297-301 (20. und 21.11.2003).

[V4] Meinike, T.: „SVG auf dem Weg zum reichhaltigen Medienformat – Neuerungen in Version 1.2“. In: tekomp, Gesellschaft für technische Kommunikation e. V., Tagungsband zur Jahrestagung 2004 in Wiesbaden, S. 337-340 (11. und 12.11.2004).

Web-Adressen [W]

[W1] <http://www.w3.org/Graphics/SVG/>

[W2] <http://www.scale-a-vector.de/>

[W3] <http://www.svgx.org/>

[W4] <http://www.adobe.com/svg/>

[W5] <http://svglbc.datenverdrahten.de/svg/TMs10kSVGDemo.htm>

[W6] <http://svglbc.datenverdrahten.de/>

[W7] <http://www.et.fh-merseburg.de/person/meinike/mspecsvg/>

[W8] <http://www.et.fh-merseburg.de/person/meinike/ptablesvg/>

[W9] <http://ktd.et.fh-merseburg.de/~tm/xml2grafik/>

[W10] <http://www.about-svg.de/le1.html>

[W11] <http://svgmob.datenverdrahten.de/>

[W12] <http://www.et.fh-merseburg.de/person/meinike/>