

## SCALABLE VECTOR GRAPHICS

### SVG Tiny 1.2

```
<?xml version="1.0" encoding="utf-8"?>
<!-- Generator: Adobe Illustrator 12.0.1, SVG Export Plug-In . SVG
Version: 6.0.0 Build 51448) -->
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd" [
  <!ENTITY ns_flows "http://ns.adobe.com/Flows/1.0/">
  <!ENTITY ns_extend "http://ns.adobe.com/Extensibility/1.0/">
  <!ENTITY ns_ai "http://ns.adobe.com/AdobeIllustrator/10.0/">
  <!ENTITY ns_graphs "http://ns.adobe.com/Graphs/1.0/">
  <!ENTITY ns_svg "http://www.w3.org/2000/svg">
  <!ENTITY ns_xlink "http://www.w3.org/1999/xlink">
]>
<svg version="1.1" id="Layer_1" xmlns:x="&ns_extend;" xmlns:i="&ns_ai;"
xmlns:graph="&ns_graphs;"
  xmlns="&ns_svg;" xmlns:xlink="&ns_xlink;"
  xmlns:a="http://ns.adobe.com/AdobeSVGViewerExtensions/3.0/"
  width="150.684" height="169.572" viewBox="0 0 150.684 169.572"
  overflow="visible" enable-background="new 0 0 150.684 169.572"
  xml:space="preserve">
```

profile

LAHDEN AMMATTIKORKEAKOULU

Mediatekniikan koulutusohjelma

Teknisen visualisoinnin suuntautumisvaihtoehto

Opinnäytetyö

27.11.2007

Anni Kotisalo

# SCALABLE VECTOR GRAPHICS

LAHDEN AMMATTIKORKEAKOULU  
Mediatekniikan koulutusohjelma  
Teknisen visualisoinnin suuntautumisvaihtoehto  
Opinnäytetyö  
27.11.2007  
Anni Kotisalo

**KOTISALO, ANNI: Scalable Vector Graphics**

Teknisen visualisoinnin opinnäytetyö, 48 sivua, 1 liitesivu

Syksy 2007

**TIIVISTELMÄ**

---

Tämä opinnäytetyö käsittelee SVG:tä eli skaalautuvaa vektorigrafiikkaa. Aihetta tarkastellaan tekniseltä kannalta selventäen formaatin rakennetta, soveltuvuutta ja luomista.

Skaalautuvaan vektorigrafiikkaan tutustuminen aloitetaan kertomalla sen kehityshistoriasta: mistä se on saanut alkunsa ja miten se on matkan varrella muokkautunut nykyiseen muotoonsa. Lyhyesti kerrotaan myös mitä ovat XML ja vektorigrafiikka, koska nämä yhdistyvät SVG-formaatissa. Seuraavaksi käydään läpi standardin eri profiilit, mitä ne ovat ja mihin tarkoituksiin niistä kukin on laadittu.

Pääosa tekstistä käsittelee SVG:n teknistä rakennetta. Tarkastelu etenee alkaen perusteista, kuten dokumentin määrittelyt, käytettävissä olevat mittayksiköt ja yksinkertaisimmat graafiset elementit, jatkaen graafisten tehosteiden kanssa ja lopuksi tutustutaan interaktiivisuutta tuoviin ominaisuuksiin, kuten animointi ja ohjelmointi.

Soveltamisesta kertovassa luvussa käydään läpi yleisimmät käyttökohteet ja pohditaan SVG:n soveltuvuutta näihin tarkoituksiin. Työssä tarkastellaan myös ohjelmistoja, joilla skaalautuvaa vektorigrafiikkaa on mahdollista tuottaa ja muokata, sekä tutkitaan katseluohjelmien ja selaimien SVG-tukea.

Lopuksi kerrotaan pienimuotoisen casen toteuttamisesta ja tuloksista; casen tarkoituksena on kokeilla hieman käytännössä miten SVG toimii.

Avainsanat: SVG, skaalautuvuus, vektorigrafiikka, XML, mobiili

**Lahti University of Applied Sciences**  
**Faculty of Technology**  
**Degree Programme in Media Technology**

**KOTISALO, ANNI: Scalable Vector Graphics**

Bachelor's Thesis in Visualisation Engineering, 48 pages, 1 appendix

Autumn 2007

## **ABSTRACT**

---

This thesis deals with SVG, i.e. Scalable Vector Graphics. The purpose is to look at the subject from a technical point of view by studying the structure, applicability and creating of the format.

The thesis starts by exploring the development history of Scalable Vector Graphics, where it began and how it has been modified to its current form. XML and vector graphics are also explained shortly, as both these formats are combined in SVG. The next part describes the different Profiles of the standard, explaining what they are and which purposes each is made for.

The main part of the text discusses the technical structure of SVG documents. The analysis starts with the basics, such as document definitions, available measure units and the simplest graphic elements; continuing with graphic effects and ending with features that allow interactivity, such as animation and scripting.

The chapter about application deals with the most common uses and the applicability of SVG in them. There is also a look at software with which Scalable Vector Graphics can be created and modified, as well as the SVG support of viewers and browsers.

At the end there is a report of the implementation and results of a small case, which is intended simply to show how SVG works in practise.

Keywords: SVG, scalability, vector graphics, XML, mobile

# SISÄLLYS

1	JOHDANTO.....	1
2	TAUSTA .....	2
2.1	Kehityshistoria .....	2
2.1.1	XML.....	3
2.1.2	Vektorigrafiikka.....	3
2.2	SVG:n standardit .....	4
2.2.1	SVG 1.1.....	4
2.2.2	SVG Mobile.....	4
2.2.3	SVG Print.....	5
2.2.4	SVG 1.2 .....	6
3	TEKNINEN RAKENNE.....	7
3.1	Perusrakenne ja käsitteet.....	7
3.1.1	SVG-tiedoston yleisrakenne .....	7
3.1.2	Peruskäsitteet .....	8
3.1.3	Yksiköt ja arvoalueet .....	9
3.1.4	Geometria .....	9
3.2	Grafiikan määrittelyt .....	10
3.2.1	Muodot.....	10
3.2.2	Värit ja täytöt .....	11
3.2.3	Teksti.....	12
3.2.4	Bittikartat ja maskit.....	14
3.2.5	Suodattimet.....	14
3.3	Linkittäminen, HTML ja CSS.....	15
3.4	Animointi.....	16
3.5	Ohjelmointi.....	17
3.5.1	XML.....	17
3.5.2	DOM ja JavaScript.....	18
4	SOVELTAMINEN .....	20
4.1	Käyttökohteista .....	20
4.1.1	Websovellukset ja grafiikka.....	20
4.1.2	Pelit ja interaktiivisuus .....	21
4.1.3	Tekniset piirustukset .....	22
4.1.4	Karttasovellukset .....	23
4.1.5	Käyttöliittymät.....	24
4.2	Editorit .....	25
4.2.1	Perinteiset vektorigrafiikkaohjelmistot.....	25
4.2.2	SVG-pohjaiset ohjelmistot.....	26
4.2.3	Konversiotyökalut .....	27

4.3	Mobiili SVG .....	27
4.3.1	SVG:n soveltaminen mobiililaitteissa.....	27
4.3.2	Mobiili-viewerit ja toolkitit.....	28
4.4	Selaintuki ja viewerit .....	30
5	CASE: SVG-kuvan toteutus .....	31
5.1	Case:n kuvaus.....	31
5.2	Toteutusvaiheiden määrittely.....	32
5.3	Toteutus.....	33
5.3.1	Testi A: Toteutustapojen vertailu.....	34
5.3.2	Testi B: Katseluohjelmien ja SVG-versioiden testaus .....	36
5.4	Lopputulosten arviointia ja yhteenveto .....	39
6	YHTEENVETO.....	40
	LÄHTEET .....	41
	LIITTEET .....	46

## LYHENTEET JA TERMIT

3GPP

API – Application programming interface, ohjelmointirajapinta

Attribuutti – elementtikohtainen tai yleinen määrittävä termi

CSS – Cascading Style Sheet, tyylitiedosto

DOM – Document Object Model, alusta- ja kieliriippumaton ohjelmointirajapinta

DTD – Document Type Definition, määrittelee dokumentin rakenteen

Elementti – dokumentin osa

(X)HTML – eXtensible Hypertext Markup Language, html-kielestä kehitetty xml-vaatimukset täyttävä merkintäkieli

J2ME – Java 2 Mobile Edition

J2SE – Java 2 Standard Edition

JavaScript – web-ohjelmointikieli

Moduuli, modulaarisuus – itsenäinen osa rakennetta

Skaalautuva, skaalautuvuus – kokoaan muuttava

SMIL –

Spesifikaatio – tekninen standardi

SVG – Scalable Vector Graphics, skaalautuva vektorigrafiikka

Profiili – spesifikaatiosta jäsennetty standardimääritelmä

URI – Uniform Resource Identifier

Viewer – katseluohjelma, itsenäinen tai liitännäinen

W3C – World Wide Web Consortium, työryhmä joka kehittää web-standardeja

XML – Extensible Markup Language, merkintäkieli, rakenteellinen kuvauskieli

XSL – Extensible Stylesheet Language, CSS:n kaltainen tiedosto jolla määritellään XML-dokumenttien tyyliä ja muunnoksia

## 1 JOHDANTO

"Things to watch: SVG – Scalable Vector Graphics – at last, graphics which can be rendered optimally on all sizes of device."

- Tim Berners-Lee, World Wide Webin kehittäjä

Scalable Vector Graphics (SVG) on XML-pohjainen W3C:n suosittama standardi, jolla voidaan toteuttaa webissä toimivaa vektorigrafiikkaa. Standardi on avoin ja tukee monia standardeiksi tulleita tekniikoita, joista mainittakoon CSS, Java ja XML. Näitä yhdistelemällä voidaan tuottaa hyvinkin monipuolisia sovelluksia erilaisiin tarkoituksiin. Teknisten püirrosten ja karttasovellusten lisäksi SVG soveltuu myös vaikkapa pelien ja käyttöliittymien pohjaksi.

Webin lisäksi SVG:tä voidaan soveltaa myös mobiililaitteissa. Tätä varten on määritelty kaksi mobiiliprofiilia SVG Basic ja SVG Tiny, jotta standardia voidaan hyödyntää näytön ja tehon ominaisuuksilta rajoitetuissa laitteissa. SVG-tuki mobiililaitteissa, erityisesti matkapuhelimissa, kasvaa koko ajan, ja matkapuhelinteollisuus onkin ottanut SVG:n yhdeksi vaadittavista grafiikkaformaateista uuden sukupolven puhelimiin. Multimediaa käytetään viestinnässä yhä enemmän puhelinmallien uudistuessa ja käyttäjien vaatimusten ja tarpeiden muuttuessa, joten tarvitaan monipuolisempia sovelluksia ja erilaisia tekniikoita niiden toteuttamiseen.

SVG:n perusominaisuudet – skaalautuvuus, pieni tiedostokoko, avoimuus ja yhdisteltävyys muiden standardien kanssa – tekevät siitä houkuttelevan vaihtoehdon monipuolisten sovellusten kehittämiseen.



## 2 TAUSTA

### 2.1 Kehityshistoria

Scalable Vector Graphics eli SVG on World Wide Web Consortiumin (W3C) kehittämä ja suosittelema avoin web-standardi. Se on kehitetty näytöllä skaalautuvan (kokoa muuttavan) vektorigrafiikan esittämiseen; toisin sanoen, saman kuvan voi näyttää missä koossa tahansa, ilman että sen laatu kärsii. (Tuikka & Kanala 2001.)

W3C:n SVG Working Group alkoi kehittää SVG:tä vuonna 1998 sen jälkeen, kun Macromedia ja Microsoft esittelivät VML:n (Vector Markup Language) ja Adobe Systems ja Sun Microsystems kilpailevan formaatin PGML (Precision Graphics Markup Language). Sekä VML että PGML ovat XML-pohjaisia vektorigrafiikankuvauskieliä, mutta niitä ei hyväksytty W3C:n suosituksiksi (W3C Recommendation) ja niiden pohjalta alettiin kehittää SVG-standardia. XML on myös W3C:n alla kehitetty standardi. (Scalable Vector Graphics 2007.)

SVG on siis XML:ään perustuva tekstipohjainen grafiikkakieli, jolla kuvataan tekstiä, vektorigrafiikkaa ja upotettuja bittikarttakuvia. Muita XML-pohjaisia merkintäkieliä ovat: WML (Wireless Markup Language), SMIL (Synchronized Multimedia Integration Language), MathML (Mathematical Markup Language) ja CML (Chemical Markup Language). (Tuikka ym 2001.)

Ensimmäinen SVG:n W3C-suositus SVG 1.0 julkaistiin vuonna 2001 ja seuraava suositus SVG 1.1 vuonna 2003. SVG Mobile-profiilit SVG Tiny ja SVG Basic ovat SVG 1.1:n alaprofiileja. Tällä hetkellä SVG 1.2 Tiny on W3C:n suosituskandidaatti (2006), ja SVG 1.2 Full on työn alla. (Scalable Vector Graphics 2007.)

#### 2.1.1 XML

Extensible Markup Language (XML) on yksinkertainen ja joustava tekstiformaatti, joka on johdettu SGML:stä (Standard Generalized Markup Language). Rakenteeltaan XML on yksinkertaistettu versio SGML:stä, josta kehittyi HTML vuonna 1992 ja XML vuonna 1998 (W3C:n suositus). Alun perin XML suunniteltiin vastaamaan ison skaalan elektronisen julkaisemisen haasteisiin, mutta sillä on myös kasvava rooli monenlaisen tiedon välittämisessä internetissä ja muualla.

XML-dokumentit koostuvat rakenteisesta tiedosta. Toisin kuin HTML, se ei kuvaa miten tieto esitetään, vaan sillä kuvataan tietoa tiedosta. Rakenteisuus mahdollistaa tiedon luokittelun ja hakemisen. XML-dokumentissa tieto ja ulkoasu on erotettu toisistaan. Ulkoasun määrittely toteutetaan tyylikielillä XSL ja CSS.

Esimerkki XML-koodista:

```

<?xml version="1.0" encoding="ISO8859-1" ?>
  <catalog>
    <cd>
      <title>I</title>
      <artist>Led Zeppelin</artist>
      <year>1969</year>
    </cd>
    <cd>
      <title>II</title>
      <artist>Led Zeppelin</artist>
      <year>1969</year>
    </cd>
    <cd>
      <title>III</title>
      <artist>Led Zeppelin</artist>
      <year>1970</year>
    </cd>
  </catalog>
  ...

```

(Tuikka ym 2001; Extensible Markup Language (XML) 1.0 (Fourth Edition) 2006; Extensible Markup Language (XML) 2007; Wikipedia, Extensible Markup Language 2007; XML 10 kohdan tiivistelmänä 2003.)

### 2.1.2 Vektorigrafiikka

Toisin kuin pikseleistä koostuvat bittikarttakuvat, vektorigrafiikka perustuu koordinaatistoon sidottuihin geometrisiin primitiiveihin, kuten pisteet, viivat, ympyrät, (bézier-) käyrät ja monikulmiot, jotka kaikki perustuvat matemaattisiin yhtälöihin. Jokainen elementti on erillinen objekti, joka sisältää tietoa sen muodosta, väristä ja sijainnista, ja jota voi muokata loputtomiin.

Matemaattisuus mahdollistaa kuvan koon ja muodon muuttamisen ilman, että kuvan laatu kärsii. Vektorigrafiikka on resoluutio-riippumatonta, minkä ansiosta kuva ei näytä karkealta suurennettaessa, ja sen tiedostokoko on suhteessa pienempi bittikarttakuvaan verrattuna. Vektorigrafiikkakuva on kuitenkin tallennettava bittikarttamuodossa (JPEG, PNG, GIF), jotta sen voi näyttää esimerkiksi selaimessa.

Pieni osa koodista joka on vektorigrafiikkaa EPS-formaattiin tallennettuna:

```
E98D09D760A3C22CF119F9DC699A22C35B529FBC319140A0049F057F42052E
3A8D9AE5679F78981811AF51C2768AFEF74C3FB094B830F80CDDE1A4468794
30C01F07A76D3070FC382366FDE4922A7E2887FC440409174165DC0E5DB934
306DC606D3
B7B01A93A226A85F45F9E1CA97A8D4B0AF250F6391EA095F49C543CD6BF92A
0085A93936BB2FE277F3FD9C9636479A6EE4483E4C2D2EEC184E757DD2F737
4EC2D01FC0266A42B51641F914ED88AED066569E17215A1364E77B23C22D23
599BD3E278
E67D2CCF0D66EF88DF7D7DB55382093EC7F106B47184BE74430E6ACB842D38
38C1F104EAFDAEAD963C3A690561DAB46A659D0369456325AF3E2D0F461086
C3A88E05BEF21C28A20A6D0B7A90FB6F998811CE2124971AD27339192C591C
6CEF3451D8
```

(Vector Graphics, 2007; About bitmap and vector graphics, 2007; Chastain, 2007; Caplin ja Banks, 2003.)

## 2.2 SVG:n standardit

### 2.2.1 SVG 1.1

SVG 1.1 on ositettu ja moduloitu päivitys SVG 1.0:sta; moduloinnin ansiosta alemman tason standardit voidaan määritellä sen profiileina. SVG 1.1:n profiilit SVG Tiny ja SVG Basic (SVG Mobile profiilit) julkaistiin W3C suosituksina vuonna 2003.

SVG:n pääprofiili on kokoelma kaikista moduuleista, jotka mahdollistavat tietynlaisen toiminnallisuuden. Moduuleja voidaan yhdistellä keskenään ja toisissa spesifikaatioissa (kuten XHTML) määriteltyjen moduulien kanssa. Näin syntyy SVG:n ”osajoukko- ja lisäke” -dokumenttityyppejä, jotka kelpuutetaan SVG-dokumenttien tyyppiperheeseen. (Scalable Vector Graphics, 2007; Scalable Vector Graphics (SVG) 1.1 Specification, 2003.)

### 2.2.2 SVG Mobile

Teollisuuden vaatimukset, laaja SVG-työryhmän tuki ja kehittäjien pyynnöt ovat vakiinnuttaneet sen tosiasian, että jonkinlaista SVG:n muotoa tarvitaan näyttämään vektorigrafiikkaa pienillä laitteilla. Lisäksi SVG 1.0:n lähtökohtainen tarkoitus on ottaa pienet laitteet vektorigrafiikan näyttämisen kohderyhmäksi. Vastatakseen näihin vaatimuksiin SVG-työryhmä on keskittynyt sinnikkäästi pyrkimykseen luoda profiilispesifikaatio, jossa kohteena ovat mobiililaitteet.

Yksi ainut profiili ei riitä käsittelemään mobiililaitteiden kirjoa, koska jokaista mobiililaitetta koskevat erityispiirteet CPU:n nopeudesta, muistin koosta ja väri-

en tuesta. Kaksi profiilia täytyy määritellä vastaamaan erilaisia laiteperheitä. Ensimmäinen matalatasoinen profiili, SVG Tiny (SVGT), soveltuu rajoitetuille mobiililaitteille, kun taas toinen profiili, SVG Basic (SVGB) on kohdistettu korkeamman tason mobiililaitteille. Mobiililaitteiden muistin vähyyden, alhaisen CPU:n tehon, ja rajoitetun näyttöalueen takia Mobile SVG profileissa on rajoituksia sisällön, attribuutti-tyyppien, ominaisuuksien ja käyttäjätapahtumien suhteen (Liite 1).

Mobiiliprofiilien rajoittavia tekijöitä:

- Kahden profiilin avulla SVG rendautuu erilaisissa mobiililaitteissa, joissa on rajattu muisti, teho ja kaista.
- Mobile SVG-profiilien yhteensopivuus maksimoidaan SVG 1.0:n kanssa, jotta voidaan näyttää olemassa olevaa sisältöä.
- SVG 1.0:n kuvausmallista säilytetään aito osajoukko.
- Mobile SVG on suunniteltu helpottamaan export-toimintoa ohjelmistoista.
- Mobile SVG on suunniteltu niin, että SVG 1.1 voidaan muuntaa SVGB ja SVGT -muotoon säilyttäen mahdollisimman paljon skaalautuvuudesta.
- Jotta sisällön ja ohjelmistojen, jotka hyväksyvät eri profiilien käytön, yhteentoimivuus säilyy, SVGT on määritelty SVGB-profiilin alajoukoksi, ja SVGB on vastaavasti SVG 1.1:n alajoukko.

(Mobile SVG Profiles: SVG Tiny and SVG Basic, 2003.)

### 2.2.3 SVG Print

Toisin kuin SVG Mobile, SVG Print ei ole oma profiilinsa vaan ohjeistus, miten käyttää SVG:n ominaisuuksia tulostamisessa. SVG kehitettiin alun perin kuva-standardiksi web-sivuille. Sellaisenaan se on kehittänyt lukuisia graafisia ominaisuuksia, joita voidaan soveltaa erilaisiin tarkoituksiin. SVG:n versioiden 1.0 ja 1.1 kehityksen aikana pääpainopiste standardien kehitysprosessissa keskittyi näyttöpohjaisiin laitteisiin, kuten PCt, PDAt ja matkapuhelimet. SVG:n versioiden 1.1 ja 1.2 kehityksen aikana otettiin huomioon tulostamisen tarpeet, jotta voitiin tuoda esille SVG:n ominaisuuksia ja käytettävyyttä printtiympäristöissä.

Tämän ohjeistuksen tarkoitus on siis luetella, millä tavoin SVG-tiedostoja voidaan käyttää tulostamisessa. Kaikki SVG-tiedostot pitäisi olla tulostettavissa sopivalla SVG Print -laitteella. SVG 1.2:n määritelmiin on esitetty tiettyjä toiminnallisuksia auttamaan tulostamista. Tämä yleiskäyttöinen toiminnallisuus soveltuu myös muihin käyttöihin kuin tulostus, kuten esimerkiksi dia-esitykset.

Tarkan värihallinnan lisäksi SVG-kuvat tulostuvat suurimmalla mahdollisella resoluutiolla millä tahansa tulostimella. SVG-dokumentit ovat tulostettavissa samoilla grafiikkaominaisuuksilla kuin näytöllä on nähtävissä, joten tulostetuissa

SVG-kuvissa on sileät reunat ja puhtaat liukuvärit. Upotetut fontit ja epätavalliset merkit sekä suodatinefektit tulostuvat. Jos kuvaa on zoomattu tai siirretty tai kuvan animaatio on pysäytetty, sen hetkinen tila tulostuu. (SVG Print 1.2, 2007; Tutorial – printing, 2007.)

#### **2.2.4 SVG 1.2**

SVG-suosituksen kehitteillä oleva seuraava versio 1.2 on jäsennetty SVG Tiny 1.2:n mukaisesti. (Ossi Nykänen, luku 2.4.) SVG Tiny 1.2 on ollut W3C:n suositus-kandidaatti lokakuusta 2006 lähtien. Alun perin SVG Tiny 1.2 julkaistiin profiilina, ja myöhemmin jäsennettiin uudestaan kokonaiseksi spesifikaatioksi sisältämään kaikki SVG 1.1:n ja SVG 1.2:n osiot. Täten SVG Full 1.2 tullaan julkaisemaan SVG Tiny 1.2:n laajenuksena. Merkittävä lisäys SVG Full 1.2:ssa tulee olemaan multipage document -syntaksi, sekä tuki äänen ja videon lisäämiseen. (World Wide Web Consortium julkaisee SVG Tiny 1.2 -spesifikaation alustavana suositusehdotuksena, 2006; Multimedia, 2007; Multiple pages, 2004; Scalable Vector Graphics (SVG) Full 1.2 Specification, 2005.)

## 3 TEKNINEN RAKENNE

### 3.1 Perusrakenne ja käsitteet

#### 3.1.1 SVG-tiedoston rakenne

SVG on xml-kieli jolla kuvataan kaksiulotteista grafiikkaa. Se sallii kolmentyyppisiä graafisia objekteja: vektorigrafiikkamuotoja, kuvia ja tekstiä. Näitä graafisia objekteja voidaan ryhmittää, muokata ja järjestää aiemmin renderöityihin objekteihin. Teksti voi olla missä tahansa sovellukseen sopivassa xml-nimiavaruudessa, mikä parantaa SVG-grafiikan haettavuutta ja saatavuutta. Ominaisuuksiin kuuluvat sisäkkäiset muunnokset, leikkauspolut, alpha-maskit, suodatin-efektit, template-objektit ja laajennettavuus.

SVG-grafiikka voi olla dynaamista ja interaktiivista. SVG:n Document Object Model, joka sisältää täyden XML DOM-käyttöliittymän, sallii vektorigrafiikan helpon ja tehokkaan animoinnin skriptauksen avulla. Sen yhteensopivuuden ja muiden web-standardien hyödyksikäytön ansiosta ominaisuudet, kuten skriptaus, voidaan toteuttaa SVG- sekä muille xml-elementeille samanaikaisesti eri nimiavaruuksista samalla web-sivulla. (About SVG, 2004.)

XML-esittely, joka kertoo merkkikoodauksen tyyppiin:

```
<?xml version="1.0" encoding="iso-8859-1"?>
```

Tyypiesittely:

```
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"  
  "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
```

svg -elementti, jossa määritellään versio ja katseluikkunan ominaisuudet:

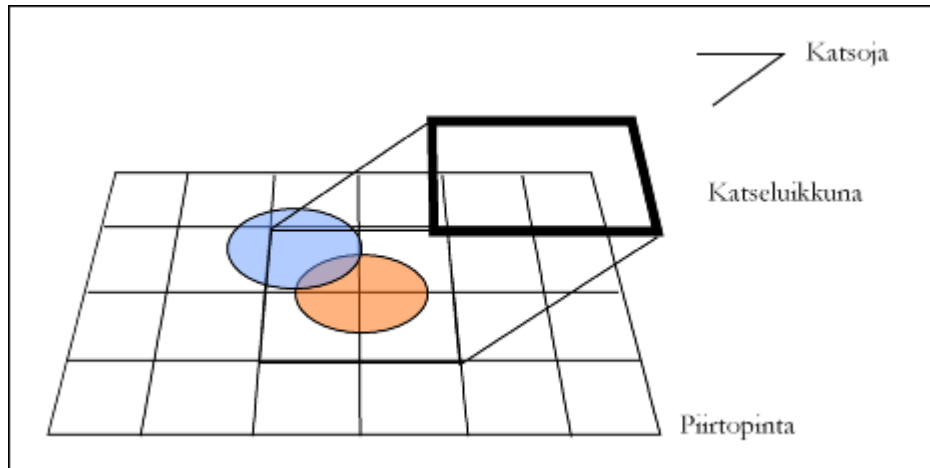
```
<svg version="1.1" viewBox="0 0 100 100"  
  xmlns="http://www.w3.org/2000/svg">  
  ...</svg>
```

Yllä esitelty svg -elementti sisältää graafiset muodot ja muut elementit, joita kuvassa halutaan esittää sekä seuraavia määritteleviä elementtejä: ?xml-stylesheet, title, defs, desc, g (group) ja use. (Nykänen 2007.)

SVG-tiedoston rakenteen voi testata validaattorilla, joka havaitsee dokumentissa esiintyvät virheet. Validoinnin voi suorittaa joko sopivan XML-jäsentimen ja SVG 1.1 tyypimäärittelyn avulla tai käyttämällä online-palvelua. On huomattava, että validaattori testaa dokumentin siinä ilmoitetun tyypiesittelyn mukaan. Ongelmia voi syntyä myös, kun käytössä on useampia scripti-kieliä samassa dokumentissa. (Nykänen 2007.)

### 3.1.2 Peruskäsitteet

Yksinkertaistettuna SVG-kuva piirretään koordinaatistoon mielivaltaisen kokoiselle piirtopinnalle, jota katsotaan katseluikkunan määrittämän alueen läpi (kuva 1). Koordinaatiston nollapiste (origo; 0, 0) on oletuksena katseluikkunan vasemmassa yläkulmassa.



Kuva 1. Katseluikkunan idea (Nykänen, 2007)

Katseluikkunan määrittäminen tapahtuu `svg`-elementissä `height` ja `width` sekä `viewBox`-attribuuteilla. Oletuksena mittayksikkönä on `px` (pikseli), joka on käytössä, jos arvolle ei anneta mittayksikköä ollenkaan, mutta attribuuteille voidaan myös antaa muita yksiköitä. Mittayksikön määrittäminen `svg`-elementissä ei vaikuta muissa elementeissä annettuihin mittayksiköihin, ja niitä voidaan käyttää sekaissä. (Nykänen 2007.)

”Katseluikkuna rajataan ja skaalataan oletusarvoisesti koordinaatiston suhteella 1:1 siten, että koko `viewBox`-alue on näkyvässä ja koko `viewBox`-laatikko keskitetään katseluikkunan sisälle maksimikorkeuden tai -leveyden perusteella. Tätä vastaa oletusarvo `preserveAspectRatio="xMidYMid meet"`.” (Nykänen 2007, s.64)

Kuvasuhdetta voidaan säätää (skaalata) antamalla `preserveAspectRatio`-attribuutille toisenlaisia arvoja, jotka jaotellaan `align`- ja `meetOrSlice`-termien alle.

`Align`-termin `none` sekä `x`- ja `y`-koordinaattien alku-, väli- ja loppupisteiden mukaan ilmoitettavat arvot (`xMinYMin`, `xMidYMin`, `xMaxYMin`, `xMinYMid`, `xMaxYMid`, `xMinYMax`, `xMidYMax` ja `xMaxYMax`) määrittävät kuvasuhteen, eli miten `viewBox` asettuu katseluikkunaan. Termin `meetOrSlice` arvot `meet` ja `slice` kertovat, miten `viewBox` täyttää katseluikkunan. (Nykänen 2007.)

### 3.1.3 Yksiköt ja arvoalueet

SVG:n pituusyksiköt jaotellaan absoluuttisiin (pt [piste], pc [pica], cm [sentti], mm [milli], in [tuuma]) ja suhteellisiin (em [fontin koko], ex [fontin x-korkeus], ja % [pituus suhteessa kontekstiin]). Suhteellisella mittayksiköllä määritellyn arvon absoluuttinen koko riippuu kontekstista, näytön tarkkuudesta tai esim. selaimen valitusta fontikoosta.

Käytettävissä olevia arvoalueita ovat:

- o luvut: kokonaisluvut, desimaaliluvut, pituusarvot, koordinaatit
- o arvolistat – erotin joko tyhjämerkki tai pilkku
- o kulmat
- o suhdeluvut
- o väri (rgb-arvoilla tai avainsanoilla ilmoitettuna)
- o resurssit
- o taajuudet.

(Nykänen 2007.)

### 3.1.4 Geometriset muunnokset

Erilaisille piirto-objekteille tai ryhmille voidaan tehdä geometrisia muunnoksia attribuutilla `transform`. Sillä on joukko erilaisia muunnosoperaatioita (`matrix`, `translate`, `scale`, `rotate`, `skewX` ja `skewY`), joilla voi siirtää, kiertää ja vinon-taa objekteja eri suuntiin. Lopputulos on operaatioiden yhteisvaikutus, ja niiden järjestys voi myös vaikuttaa muunnoksen tulokseen.

Muunnokset asetetaan `defs`-osiossa määritellylle objektille käyttämällä elementtiä `use` ja viittaamalla siinä objektiin (tai ryhmään). Attribuutilla `transform` määritellään käytettävä muunnos ja sille annetaan arvo tai arvoparit, joiden mukaan muunnos tapahtuu. `Matrix`-operaatio tosin saa yleisen 3x3 muunnosmatriisin mukaan kuusi arvoa. (Nykänen 2007; Concepts SVG 1.1, 2003.)



## 3.2 Grafiikan määrittelyt

### 3.2.1 Muodot

Vektorigrafiikka perustuu erilaisiin geometrisiin perusmuotoihin. Näitä SVG-kuvissa toistuvasti tarvittavia perusmuotoja eli (graafisia) objekteja ovat: suorakulmio, ympyrä, ellipsi, viivat, monikulmio ja yleinen polku (path).

SVG-elementin sisään määritellään suorakulmio `rect`-elementillä:

```
<rect x="400" y="100" width="400" height="200"
      fill="lightblue" stroke="navy" stroke-width="15"
      stroke-linejoin="round"/>
```

**Taulukko 1 (Nykänen, taulukko 4.1, s.78)**

Elementin <code>rect</code> keskeiset attribuutit:	
Nimi	Kuvaus
<code>x</code>	Sijainti x-akselin suhteen
<code>y</code>	Sijainti y-akselin suhteen
<code>width</code>	Leveys
<code>height</code>	Korkeus
<code>rx</code>	Pyöristetyn suorakaiteen kulman x-säde
<code>ry</code>	Pyöristetyn suorakaiteen kulman y-säde

Elementillä `rect` on Taulukon 1 mukaiset attribuutit. Lisäksi siihen määriteltävissä ovat täytön ja reunaviivan määrittelyt `fill` ja `stroke`.

Vastaavasti määritellään ympyrä `circle`-elementillä. Ellipsi piirretään samoin, mutta lisäämällä toinen säde.

```
<circle cx="600" cy="200" r="100"
        fill="lightblue" stroke="navy" stroke-width="15"
        stroke-dasharray="30px 20px 10px 40px"/>
```

Erilaisten viivojen ja monikulmioiden piirtäminen tapahtuu yksinkertaistettuna määrittämällä alku-, (väli-) ja loppupisteet. Näiden elementtien (`line`, `polyline`, `polygon`) keskeiset attribuutit ovat `x1`, `x2`, `y1`, `y2` tai `points`.

Yleinen polku (`path`) määritellään hieman monimutkaisemmin. Attribuutille `d` (`draw`) annetaan lista piirtokomentoja, joiden mukaan piirretään murtoviivoja, sekä määritellään apupisteitä käyriä varten.

Pisteiden määrittelyyn käytetään seuraavanlaisia komentoja:

- kynän asemointi: moveto (M, m)
- polun sulkeminen: closepath (Z tai z)
- viivan piirtäminen: lineto (L, l, H, h, V, v)
- käyrän piirtäminen: curveto (C, c, S, s, Q, q, T, t, A, a).

Iso kirjain tarkoittaa absoluuttista ja pieni kirjain suhteellista komentoa. (Nykänen 2007.)

### 3.2.2 Värit ja täytöt

Graafisille objekteille on määriteltävissä piirto-ominaisuuksia CSS2-suosituksen mukaan joko elementin sisällä inline-koodina tai dokumenttikohtaisen tyylin tai ulkoisen tyylitiedoston avulla. Pääasiassa objektin täytteelle ja reunaviivalle voidaan antaa erilaisia piirto-ominaisuuksia, joita ovat: yksi väri, läpinäkyvä väri, liukuväri, täyttökuvio (vektori- tai bittikarttakuva) tai SVG:n laajennusominaisuuksiin perustuva täyteoperaatio. Täyte ja reunaviiva voidaan myös jättää piirtämättä, jolloin tausta näkyy läpi ja SVG-kuvan rakenteen mukaisesti toisten muotojen ominaisuudet periytyvät objektille.

Täytteen ominaisuudet määritellään seuraavasti:

**Taulukko 2 (Nykänen, taulukko 4.9, s.93)**

Täytteen ominaisuudet		
Ominaisuus	Arvot	Kuvaus (oletusarvo)
fill	<paint>	Täyteväri (black)
fill-rule	nonzero   evenodd   inherit	Asetetaan monimutkaisen muodon sisäpuoli (nonzero)
fill-opacity	<opacity-value>   inherit	Täytteen peittävyys (1)

Jossa <paint> voi saada hyvinkin monimutkaisia arvoja:

```
none | currentColor | <color> [icc-color( <name>[, <iccvalue>]*)] | <uri> [ none | currentColor | <color> [icc color(<name>[, <iccvalue>]*)]] | inherit.
```

SVG-dokumentit käyttävät tiettyä väriprofiilia näyttämään värit oikein, selaimesta tai näytöstä riippumatta. (Nykänen 2007; Introduction to SVG, 2003; Color SVG 1.1, 2003.)

## Liukuväri

Gradient eli liukuväri määritellään svg-dokumentin määrittelyosiossa `defs`-elementissä, johon viitataan objektin täyttövärissä. Suoraviivainen liukuväri (`linearGradient`) täyttää piirtoalansa samansuuntaisilla, erivärisillä suorilla, kun taas säteittäinen liukuväri (`radialGradient`) täyttää piirtoalansa erivärisillä samankeskisillä kaarilla. Määrittelyosiossa kuvataan liukuväri, johon viitataan elementin `fill`-attribuutissa:

```
<linearGradient id="myG">
<stop offset="0%" stop-color="black" />
<stop offset="100%" stop-color="white" />
</linearGradient>
```

```
...fill="url(#myG)"...
```

## Täytekuviot

Vastaavasti kuin liukuväri, myös täytekuvio määritellään svg-dokumentin määrittelyosiossa. Elementin `defs` sisällä `pattern`-elementissä määritellään toistuva kuvio, jota halutun objektin täyttö käyttää. Pattern-elementti sisältää piirtoobjektit, joista täytekuvio muodostuu. Kuvion sijaintiin ja kokoon vaikuttavat attribuutit `x`, `y`, `width`, `height` ja `viewBox`. (Nykänen 2007.)

### **3.2.3 Teksti**

Kaksi keskeistä elementtiä, joilla määritellään tekstiobjekteja SVG-dokumentissa, ovat `text` ja `tspan`. Nämä vastaavat käytöltään (X)HTML-kielen `p` ja `span` elementtien käyttöä. Niiden sisällä määritellään esimerkiksi fontin koko ja tyyppi, korostus, väri, merkkiväli ja asemointi. SVG:n fonttiominaisuuksiin ei tässä tarkemmin syvennyttä. Näiden lisäksi elementeillä on seuraavat keskeiset attribuutit: `x`, `y`, `dx`, `dy`, `rotate`, `textLength` ja (vain `text`-elementillä) `lengthAdjust` (Taulukko 3).

Taulukko 3 (Nykänen, taulukot 6.1 ja 6.2, s.110 ja 113)

text- ja tspan-elementtien keskeiset ominaisuudet	
Nimi	Kuvaus
x	Sijainti x-akselin suhteen
y	Sijainti y-akselin suhteen
dx	Tekstin siirto kontekstin suhteen x-suunnassa
dy	Tekstin siirto kontekstin suhteen y-suunnassa
rotate	Tekstin kierto
textLength	Vihje tekstin pituudesta
lengthAdjust	Pituuden hienosäädön tyyppi (vain text-elementissä)

XML-tekstiformaattien tapaan SVG perustuu kansainväliseen Unicode-merkistöön, joka on oletuksena UTF-8. Tämän ansiosta SVG-dokumenttiin voidaan koodata tekstiä periaatteessa millä kielellä tahansa, sekä lisätä merkkejä dokumenttiin ns. merkkiviittausten avulla. Lisäksi dokumentissa esiintyvä tekstin luonnollinen kieli voidaan ilmoittaa attribuutin `xml:lang` avulla, esim:

```
<svg xml:lang="fi" ...>
...
</svg>
```

### Monirivinen teksti

SVG 1.1 ei tue rivitettyä tekstiä suoraan – ts. se ei osaa tasata tekstiä annetun suorakaiteen sisälle – mutta versiossa 1.2 tämä tulee korjaantumaan. Monirivistä tekstiä voidaan toteuttaa `text` ja `tspan` elementtien avulla joko usealla `text`-elementillä tai yhdellä `text`-elementillä, jonka sisällä on useita `tspan`-elementtejä. Attribuutit `dx`, `dy` ja `rotate` `tspan`-elementissä mahdollistavat tekstin rakenteen ja piirto-ominaisuuksien määrittelyn, ja sen ominaisuudet tietyissä rajoissa ohittavat `text`-elementissä määritellyt ominaisuudet. (Nykänen 2007.)

### Asemointi ja ositus

Teksti, kuten muutkin objektit, asemoidaan koordinaatistoon x- ja y-akselien suhteen. Tekstin x-korkeus (`ex`) on myös avuksi sijoittelussa. Eri fonteilla x-korkeus tosin vaihtelee, jolloin pistekooltaan samankokoiset fontit näyttävät erikokoisilta. Elementin `text` sisällä voidaan myös viitata aiemmin `defs`-osiossa määriteltyyn tekstiin. Viittaus tapahtuu `tref`-elementillä attribuutin `xlink:href` avulla.

## Polkua pitkin

Elementillä `textPath` määritellään polku, jota pitkin teksti kulkee. Polun näkyvyys riippuu siitä, määritelläänkö polku `defs`-osiossa vai `text`-elementissä.

**Taulukko 4 (Nykänen, taulukko 6.3, s.117)**

textPath-elementin spesifit attribuutit		
Attribuutti	Arvot	Kuvaus (oletusarvo)
<code>startOffset</code>	<code>&lt;length&gt;</code>	Tekstin alkupiste polulla (0)
<code>method</code>	<code>align   stretch</code>	Venytetäänkö teksti polulle ( <code>align</code> )
<code>spacing</code>	<code>auto   exact</code>	Kuvakkeiden välistys ( <code>exact</code> )
<code>xlink:href</code>	<code>&lt;uri&gt;</code>	Viittaus polkuun (e/o)

(Nykänen 2007.)

### 3.2.4 Bittikartat ja maskit

Pikseleistä koostuvan bittikartan koko ja värien määrä määrittelevät bittikarttakuvan tarkkuuden ja tilantarpeen. Suurentaminen tekee kuvasta karkean, kun taas kuvan tarkkuuden lisääminen vie enemmän muistitilaa. SVG:ssä bittikarttakuvan pikseli koostuu neljästä komponentista (lukuarvosta); kuvat ovat ns. RGBA-muotoisia. Neljäs komponentti on Alpha-kanava, joka määrää kuvapisteen läpinäkyvyyden.

Bittikarttakuvan lisääminen SVG-dokumenttiin tehdään `image`-elementillä, jossa määritellään kuvan sijainti ja koko. Kuva voi olla PNG-, JPG- tai SVG-formaatissa, ja siihen viitataan `xlink:href` attribuutin avulla.

Leikkauspolkujen (`clipping path`) avulla voidaan leikata osia pois bittikarttakuvasta tai vektoriobjekteista tai niiden yhdistelmistä. Polku, joka määritellään `defs`-osiossa `clipPath`-elementillä, määrittelee kuvion, joka rajaa kuvasta näytettävän osan. Leikkauskuviona voidaan käyttää SVG:n perusmuotoja, yleisiä polkuja tai tekstiobjekteja.

Leikkauspolkuja laajempi tehoste on maski; apukuvio, jonka mukaan kohdekuvion tai kuvan läpinäkyvyys määrittyy. Maski määritellään myös `defs`-osiossa, elementillä `mask`, ja siihen voi sisältyä erilaisia ominaisuuksia `clipPath`-elementin tavoin. (Nykänen 2007.)

### 3.2.5 Suodattimet

Kuvankäsittelyohjelmista tuttuja suodattimia (filter) hyödyntäen SVG-kuvaan voidaan tehdä erilaisia tehosteita. SVG:n suodatin koostuu säiliöelementistä `filter` (Taulukko 5), joka sisältää graafisia operaatioita suorittavia suodatinprimitiivejä. Tuloksena on suodatinten mukainen graafinen muunnos.

**Taulukko 5 (Nykänen, taulukko 8.1, s.138)**

filter-elementin keskeiset attribuutit	
Nimi	Kuvaus
<code>filterUnits</code>	Koordinaatisto ( <code>userSpaceOnUse</code>   <code>objectBoundingBox</code> )
<code>primitiveUnits</code>	Primitiivien koordinaatisto ( <code>userSpaceOnUse</code>   <code>objectBoundingBox</code> )
<code>x</code>	Sijainti x-akselin suhteen
<code>y</code>	Sijainti y-akselin suhteen
<code>width</code>	Leveys
<code>height</code>	Korkeus
<code>xlink:href</code>	Viittaus kuvan esittävään resurssiin
<code>filterRes</code>	Suodattimen tarkkuus x- ja y-suunnissa ( <code>arvo x-pixels [y-pixels]</code> )
<code>xlink:href</code>	Viittaus toiseen suodattimeen

Suodatinprimitiivien komentoja, eli millaisia efektejä suodattimilla saadaan:

- o valonlähteen määrittely
- o kahden kuvan sekoittaminen
- o värien muunnokset
- o kuvan kirkkauden, kontrastin jne hallinta
- o kuvien sumentaminen, reunan korostaminen, terävöittäminen jne
- o kuvan pikselien liikuttelu
- o kuvan sumentaminen
- o kuvakerrosten yhdistäminen
- o kuvan lihavoittaminen tai ohentaminen
- o kuvan liikuttelu kokonaisuutena
- o alueen monistaminen kuvilla
- o keinotekoisten tekstuurien rakentaminen
- o apukomennot.

Maskien tapaan suodattimet määritellään `defs`-osiossa, ja siihen viitataan `use`-elementin sisällä `filter`-attribuutilla. (Nykänen 2007.)

### 3.3 Linkittäminen, HTML ja CSS

SVG-kuva voidaan liittää osaksi toista dokumenttia. Linkittäminen toimii SVG-dokumentissa samoin kuten HTML-dokumentissa. Kohteen lähde (linkin määränpää) määritellään URI:lla `a`-elementin `xLink href`-attribuutilla (Taulukko 6). SVG soveltaa `XLink`-suosituksesta yksinkertaisen linkin käsitettä.

Taulukko 6 (Nykänen, taulukko 2.2, s.42)

a-elementin keskeiset attribuutit	
Nimi	Kuvaus
<code>xlink:type</code>	Linkin tyyppi <code>XLink</code> -suosituksen mukaan (kiinteä arvo "simple")
<code>xlink:role</code>	Linkin rooli <code>XLink</code> -suosituksen mukaan; käytännössä nimi
<code>xlink:arcrole</code>	Monensuuntaisen linkin kaaren rooli <code>XLink</code> -suosituksen mukaan
<code>xlink:show</code>	Avataanko uusi ikkuna linkin kohteelle ( <code>new</code> ) vai korvataan nykyinen sisältö ( <code>replace</code> )
<code>xlink:actuate</code>	Linkin prosessointiajankohta <code>XLink</code> -suosituksen mukaan (kiinteä arvo <code>onRequest</code> )
<code>xlink:href</code>	Linkin kohteen ilmaiseva URI-osoite
<code>target</code>	Linkin kohde silloin, kun se ei ole suoraan konteksti; linkin kohde voi olla esim. toinen ikkuna

HTML-dokumentissa voidaan viitata SVG-dokumenttiin `img`, `object` tai `applet` (viittaa kuvan käsittelijään) elementillä, sekä hypertekstilinkillä (`a href`). Periaatteessa SVG-kuva voidaan liittää kohdedokumenttiin myös inline-koodina liittämällä SVG-tekstikoodi suoraan esim. HTML-sivun lähdekoodiin, mutta nykyinen XHTML-suositus ei vielä tue tätä tapaa. (Linking SVG 1.1, 2003.)

CSS on yksinkertainen mekanismi, jolla lisätään tyylejä HTML-dokumentteihin. SVG hyödyntää CSS2-standardin ja HTML-sovelluksista tuttuja tyylisääntöjä asettamaan kuvien ominaisuuksia elementillä `style`. Näiden lisäksi myös XML-dokumenttien muunnoksia ja esitystä määrittelevät XSL-tyylit ovat mahdollisia.

Tyylit voidaan siis määritellä `style`-elementissä SVG-dokumentin sisällä. SVG:stä voidaan myös viitata ulkoiseen CSS-tiedostoon:

```
<?xml-stylesheet href="tyyli.css" type="text/css"?>
```

(Nykänen 2007; Cascading Style Sheets, 2007; The Extensible Stylesheet Language Family (XSL), 2007.)

### 3.4 Animointi

”SVG-animaatiolla tarkoitetaan kuvassa ajassa tapahtuvaa muutosta, joka on määritelty kuvailevan merkkauksen keinoin. Animaatiot voivat muuttaa piirto-objektien sijaintia, muotoa sekä piirto-ominaisuuksia. Animaatiot voidaan ajastaa, ne voidaan ajoittaa alkamaan peräkkäin ja animaatio voi reagoida erilaisiin tapahtumiin.” (Nykänen)

SVG:ssä toteutettavissa olevat animaatiot määritellään seuraavasti:

**Taulukko 7 (Nykänen, taulukko 9.1, s.148)**

Animoinnin eri tekniikat	
Nimi	Kuvaus
animate	Objektin yksittäisen valitun, lähinnä numeerisen ominaisuuden animointi
Set	Objektin yksittäisen ominaisuuden asettaminen; mahdollistaa paloittain etenevät animaatiot
animateMotion	Objektin liikuttaminen polkua pitkin
animateColor	Objektin väriominaisuuksien animointi
animateTransform	Objektin geometristen ominaisuuksien animointi

SVG:ssä animaatiot perustuvat SMIL-suosituksen (Synchronized Multimedia Integration Language) määritelmiin multimediaesitysten koostamisesta. SMIL kuvaa animaatorakenteen sekä joukon XML-animaatioelementtejä, joita voidaan yhdistää XML-dokumentteihin. SVG:n animaatiomääritelmä pohjautuu myös SMIL 1.0 ajastusmalliin, muutamien lisäkkeiden kera, ja on SMIL 2.0:n osajoukko. Tämä tarjoaa ponnahduslaudan monimutkaisiin toteutuksiin sovelluksille, joissa halutaan käyttää SMIL-yhteensopivaa animaatiota, muttei sisällyttää aikaisäiliöitä (time containers).

SVG:ssä on viisi tapaa tuottaa animaatiota: animate, animateMotion, animateTransform, animateColor ja set (Bring Scalable Vector Graphics to life with built-in animation elements, 2003).

Elementtejä, joita voidaan animoida, ovat: g, defs, use, image, switch, path, rect, circle, ellipse, line, polyline, polygon, text, clipPath, mask, a ja foreignObject. Jokaisen elementin sisältämiä ominaisuuksia voidaan myös animoida. (Nykänen 2007; SMIL Animation, 2001; Synchronized Multimedia Integration Language (SMIL) 1.0 Specification, 1998; Animation SVG 1.1., 2003.)



## 3.5 Ohjelmointi

### 3.5.1 XML

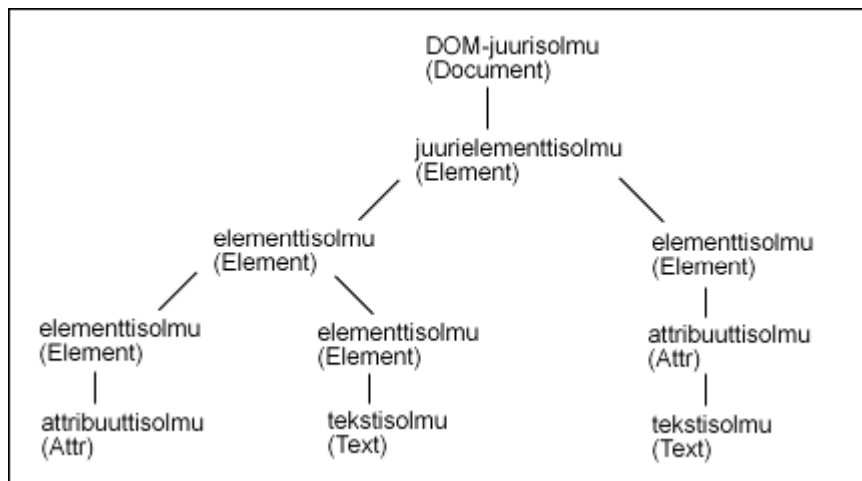
(SVG-)ohjelmointi laajentaa SVG-kuvien käyttömahdollisuuksia. SVG-tiedostot ovat tavallaan xml-tiedostoja, joten XML on luonteva keino SVG-sovellusten luomiseen. XML on tiedon tekstipohjainen väline ja rajapinta erilaisten sovellusten välillä, ts. se on tiedon kuvaamista ja siirtämistä sovelluksesta toiseen.

Kun tietoa kuvataan tiettyyn visuaaliseen muotoon, sitä kutsutaan datalähtöiseksi grafiikaksi. Tieto voi olla minkä tahansa tyyppisen tietovarannon tietoa tai jonkin sovelluksen kuvaamaa tietoa, esimerkiksi taulukkomuotoon kirjoitettuna XML-dokumentissa. Ideana on automatisoida tiedon tuottaminen kuvalliseen muotoon. Tietoa voidaan koota useammasta lähteestä samaan tulodokumenttiin, jossa tieto ja sen esitystapa yhdistetään.

Kun halutaan muuntaa tai kuvata yksi tai useampi XML-dokumentti yhteen tekstidokumenttiin, HTML-sivuun tai yleiseen XML-dokumenttiin, käytetään apuna XSL-muunnoksia. Nämä muunnokset määrittelevät joukon sääntöjä, jotka viittaavat lähdedokumenttien rakenteeseen ja sisältöön, ja niiden avulla voidaan tyylilien lisäksi määritellä erilaisia tekstisuodattimia. XML-muunnosten tuottamiseen käytetään muunninohjelmaa, joita ovat esimerkiksi Java-pohjaiset Saxon ja Xalan. (Nykänen 2007; Extensible Markup Language (XML) 1.0 (Fourth Edition), 2006; Extensible Markup Language (XML), 2007.)

### 3.5.2 DOM ja JavaScript

SVG-sovellusten toteuttamiseen voidaan käyttää myös DOM-rajapintaa hyödynnäviä keinoja. DOM (Document Object Model) on matalan tason ohjelmointirajapinta, jonka avulla päästään XML-dokumentin tietorakenteeseen tutkimaan dokumentin tietosisältöä ja rakennetta sekä muokkaamaan dokumentin tietoja. DOM:in jäsennetty objektipuu ja sen solmut (node) vastaavat XML-dokumentin loogista rakennetta, joten siinä on elementtisolmuja, attribuuttisolmuja ja tekstisolmuja (kuva 2).



Kuva 2. DOM-puun rakenne (Nykänen, s.191)

DOM-ohjelmointi toteutetaan aina jollakin tietyllä ohjelmointityökalulla ja -kielellä. Yleisin ohjelmointikieli tässä tarkoituksessa on JavaScript, mutta DOM-rajapinnan toteuttavia XML-jäsentimiä on saatavilla monille eri ohjelmointikielille. JavaScript on nykyisellään oliomuotoinen komentosarjakieli jota käytetään pääasiassa web-ympäristössä, standardoituna versiona sitä kutsutaan ECMAScriptiksi.

Interaktiivisuutta SVG-sovelluksiin on mahdollista toteuttaa muokkaamalla SVG-dokumentin tietosisältöä katseluohjelman ohjelmointirajapinnan läpi, jolloin katseluohjelman havaitsemiin tapahtumiin voidaan reagoida reaaliaikaisesti.

Skriptit lisätään SVG-dokumenttiin `script`-elementillä tai viittaamalla ulkoiseen kooditiedostoon.

```

<script type="text/javascript">
<![CDATA[
...
]]>
</script>

```

```

<script xlink:href="etsin.js"
type="text/javascript" />

```

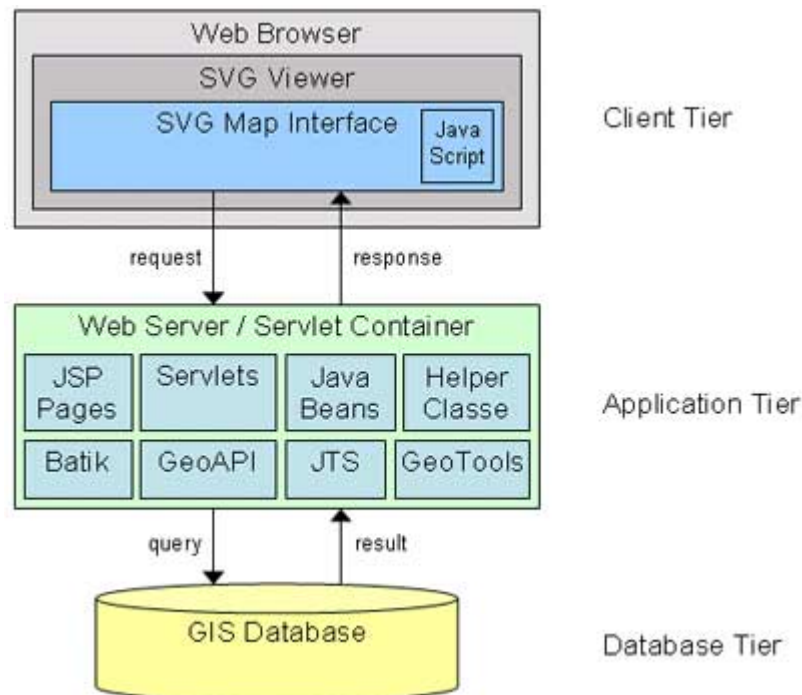
(Nykänen 2007; Scripting SVG 1.1, 2003; Document Object Model, 2005; SVG Document Object Model (DOM) SVG 1.1, 2003.)

## 4 SOVELTAMINEN

### 4.1 Käyttökohteista

#### 4.1.1 Websovellukset ja grafiikka

Web-pohjaisten sovellusten suosio kasvaa jatkuvasti, mutta selaimien yhteensopivuusongelmat ja puuttuvat toiminnallisuudet rajoittavat kehitystyötä. SVG:n vahvuutena on, että se on sekä avoin formaatti että XML-pohjainen. Avoimuuden ansiosta SVG on sopiva toimimaan erilaisten ohjelmistojen komponenttien kanssa. Se ei ole sidoksissa mihinkään tiettyyn käyttöön tai ohjelmistoon. XML-kielisyys ansiosta SVG:tä on helppo tuottaa ja muokata ohjelmistokomponenteilla, jotka käyttävät XML-merkkausta. SVG:tä voidaan käyttää alustana, jolle rakennetaan graafisia sovelluksia ja käyttöliittymiä, koska se tukee skriptausta tehokkaasti.



Kuva 3 Web-pohjaisen SVG-karttasovelluksen rakenne (GIS Development 2007.)

Web-palvelut painottuvat laajalti yhteentoimivuuteen ja XML-merkkaukseen, joten SVG soveltuu luontevasti palvelujen käyttämäksi formaatiksi esittämään graafista sisältöä. Avoimuuden ja XML-pohjaisuuden lisäksi SVG:llä on monipuolinen rakenne, ja semantiikka säilyy deskriptiivisten elementtien ja metadatan ansiosta. Tämä monipuolisuus tarjoaa mahdollisuuden web-palveluille generoida, muokata tai etsiä graafisesta sisällöstä. Esimerkiksi tätä voisi hyödyntää palvelu, joka etsii SVG-karttoja, jotka sisältävät tietyn kaupungin nimen. (About SVG, 2004; Using SVG to Create Compelling User Interfaces for Web Services, 2005;

Using GIS data intelligence on the web with Scalable Vector Graphics (SVG), 2005.)

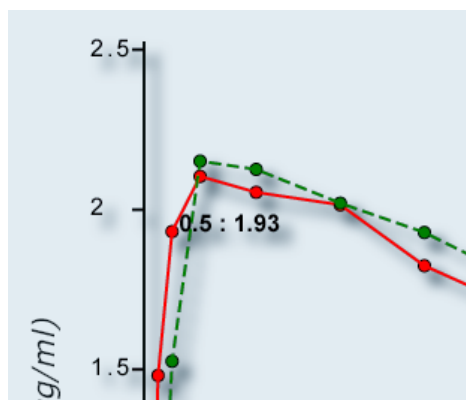
Formaatin skaalautuvuus puolestaan tulee esille, kun suunnitellaan grafiikkaa jota halutaan näyttää erikokoisilla ja -resoluutioisilla näytöillä. Sama kuvatiedosto, esimerkiksi logo, näkyy aina selkeänä koosta riippumatta.

#### 4.1.2 Pelit ja interaktiivisuus

SVG-sisältö voi olla interaktiivista (eli vastaanottavainen käyttäjä-aloitteisiin tapahtumiin) hyödyntämällä seuraavia ominaisuuksia:

- Käyttäjä-aloitteiset tapahtumat, kuten napin-painallukset osoittimella (kuten hiiri), käynnistävät animaation tai skriptin.
- Käyttäjä voi avata hyperlinkkejä uusille web-sivuille tapahtumilla, kuten hiiren klikkaus, kun osoitin sijaitsee tiettyjen graafisten elementtien päällä.
- Monissa tapauksissa, riippuen zoomAndPan-attribuutin arvoista svg-elementissä ja selaimen ominaisuuksista, käyttäjä voi zoomata ja liikkua svg-sisällössä.
- Käyttäjän tekemät liikkeet osoittimella voivat aiheuttaa muutoksia kursoriin joka näyttää osoittimen senhetkisen sijainnin.

(Interactivity SVG 1.1, 2003.)

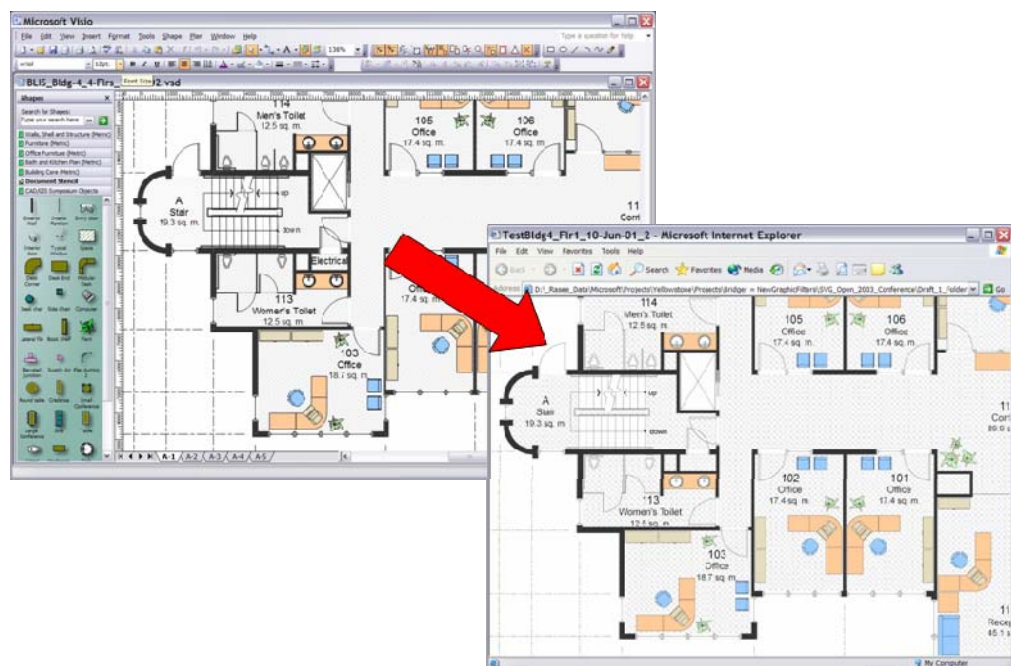


Kuva 4 SVG:llä toteutettu tieteellinen kuvaaja (Clinical Data to SVG scientific graphs 2004.)

Näitä ominaisuuksia käyttämällä ja soveltamalla voidaan tuottaa SVG:llä esimerkiksi pelejä sekä muita interaktiivisuutta hyödyntäviä sovelluksia, kuten tieteelliset visualisoinnit (kuva 4), kun SVG:n ominaisuudet yhdistetään XML:n kanssa. Esimerkiksi tieteellisen tutkimuksen tulokset syötetään sovellukseen, joka analysoi tiedon ja rakentaa siitä visuaalisen esityksen. SVG:llä toteutetut pelit voivat olla kuten mitkä tahansa Java- tai Flash-pelit.

### 4.1.3 Tekniset piirustukset

SVG soveltuisi monilta ominaisuuksiltaan hyvin tekniseen dokumentointiin, jota (ilmailu- ja avaruus, kuljetus ja liikenne, sekä telekommunikaatio) eri alojen teollisuus käyttää. XML:n laajennettavuus sallii SVG-kaavioissa upotetun metadatan sisällyttämisen sovelluskohtaisissa formateissa ilman, että se vaikuttaa esitykseen. Esimerkiksi CAD-ohjelmistosta voisi export-toiminnolla tallentaa kuvia SVG-muodossa esittämiseen netissä, ja upottaa tiedostoon tietoa joka helpottaa myöhempää muokkaamista. Sen lisäksi sitä voidaan käyttää ohjelmistojen välisenä siirtotiedostoformaattina, koska monissa suunnittelutyökaluissa on import-export tuki SVG:lle.



**Kuva 5 Microsoft Visiosta SVG-muotoon tallentaminen (SVG Scenarios using Microsoft Office Visio 2003.)**

Kun perinteisiä teknisiä sovelluksia tuodaan nettiin, ne tuovat mukanaan informaatioresursseja, jotka yhdistävät tekstiä ja dataa erittäin teknisten graafisten komponenttien kanssa. Tekniset kuvat sovelluksissa, kuten lentoalusten ylläpito- ja huoltomanuaalit, ovat tyypillisesti laajoja, monimutkaisia, todella tarkkoja, yhteensopivuus-rajoitteisia ja pitkäikäisiä. W3C:llä on kaksi standardia tällaiselle web-pohjaiselle grafiikalle, WebCGM ja SVG. Tiedon ja tiedostojen määrän takia teollisuuden nykyinen standardiformaatti on CGM (Computer Graphics Metafile), josta WebCGM on johdettu teknisten dokumenttien nettikäyttöön. (About SVG, 2004; Computer Graphics Metafile, 2007; Applicability of CGM versus SVG for technical graphics, 2006.)

#### 4.1.4 Karttasovellukset

GIS-karttajärjestelmillä on todella tarkat vaatimukset graafisten ominaisuuksien, vektori- ja rasterisisällön tuen ja suuren datamäärän käsittelysuhteen. SVG täydentää tässä hyvin OpenGIS-yhtymän GML-formaattia, erityisesti ominaisuuksilla sisällyttää metadatan ja SVG:n laajennettavuudella. GML (Geography Markup Language), joka on myös XML-pohjainen, kuvaa maantieteellisiä (graafisia) elementtejä kuten jokia ja teitä. GML-tiedostot voidaan muuntaa SVG-muotoon XML:n avulla näytettäväksi internetissä. Lisäksi moni GIS-järjestelmä tukee SVG:n export-toimintoa. (About SVG, 2004.)



Kuva 6 Lähikuva SVG-karttasovelluksesta (GIS Development 2007.)

Web-pohjaisten GIS-järjestelmien idea on yksioikoinen – tarjota reaaliaikainen pääsy tietokantoihin sekä työkalut rakentaa karttoja, hakea oleellista informaatiota ja lähettää päivityksiä takaisin tietokantaan. Kuka tahansa, jolla on internet-selain käytössään, voi suorittaa näitä tehtäviä ilman GIS-koulutusta. (GeoMedia WebMap, 2007.)

Hyvä esimerkki karttasovelluksesta, joka käyttää hyväkseen SVG:tä, on Google Maps, joka päivitti SVG-tuen karttapalveluunsa keväällä 2006. Tämä ominaisuus toimii selaimissa, joissa on natiivi svg-tuki (kuten Firefox). (Google Maps SVG Support, 2006.)

#### 4.1.5 Käyttöliittymät

Monilla upotetuilla järjestelmillä on erityisiä resurssirajoituksia, kuten pienempi näyttö, rajoitettu muistin määrä ja pienempi prosessointi-kapasiteetti, verrattuna tyypillisiin desktop-järjestelmiin. SVG Mobile -spesifikaatio suunniteltiin tällaisille laitteille ja mahdollistaa graafisten käyttöliittymien kehittämisen upotetuille järjestelmille. (About SVG, 2004.)

Matkapuhelinvalmistajista mm. Nokia ja Sony Ericsson ovat olleet mukana SVG Working Groupin alaisessa työryhmässä kehittämässä SVG:n mobiiliprofiileja ja tukevat niitä monissa puhelinmalleissaan. (SVG Phones, 2007.)



Kuva 7 MobileTV, jossa on SVG-käyttöliittymä (Beeweb Technologies 2007.)

## 4.2 Editorit

### 4.2.1 Perinteiset vektorigrafiikkaohjelmistot

#### Adobe Illustrator

Illustrator on Adobe Systems Incorporated:in kehittämä ammattilaisten suosima vektorigrafiikkaohjelmisto. Ominaisuuksiin kuuluvat alan standardit työkalut, joustava värien ja fonttien hallinta sekä tiivis yhteensopivuus muiden Adoben ohjelmien kanssa. Sillä voi tuottaa grafiikkaa printtiin, nettiin ja mobiililaitteisiin. Illustratorissa on ollut versiosta 9 asti SVG-tuki. Siitä voi suoraan tallentaa SVG ja SVGZ (SVG Compressed) -tiedostomuotoon. (Adobe SVG Zone, 2007.)

#### Adobe (Macromedia) Freehand

Freehand, nykyisin Adoben omistama Macromedian tuote, on luovan suunnittelun työkalu monisivuisien dokumenttien tuottamiseen sekä grafiikan luomiseen printtiin, nettiin ja Flashiin. Freehand ei tue SVG:tä, mutta siitä voi tallentaa esim. EPS-formaatissa vektorigrafiikkaa ja käsitellä Illustratorissa, josta tallentaa edelleen SVG-muotoon. (Freehand MX at a glance, 2007.)

#### CorelDRAW

Corel-tuoteperheessä on paljon erilaisia työkaluja grafiikan tekemiseen. Ohjelmisto-paketin tuotteilla voi luoda mm. logoja, web-grafiikkaa sekä monisivuisia esitteitä. Tuki SVG:lle alkoi versiosta 9 export-toiminnon kautta, ja versiosta 10 lähtien ohjelmistossa on täysi SVG-tuki. (CorelDRAW Graphics Suite X3, 2007.)

#### GIMP

GIMP on avoimen lähdekoodin ohjelma, joka soveltuu kuvien käsittelyyn, prosessoitiin ja luomiseen sekä formaatista toiseen muuttamiseen. Ohjelmisto on laajennettavissa erilaisten liitännäisten ja skriptien avulla. GIMP-ohjelmistosta on omat Windows- ja Mac-versiot. SVG-tuki on saatavilla liitännäisen avulla, versiosta 2.0 lähtien. (About GIMP, 2007.)



## 4.2.2 SVG-pohjaiset ohjelmistot

### InkScape

Inkscape on avoimen lähdekoodin grafiikkaohjelmisto, jonka ominaisuudet ovat samankaltaisia kuin Illustrator, Freehand, CorelDraw tai Xara X -ohjelmistoissa, mutta käyttäen tiedostoformaattina SVG:tä. Tuettuja SVG-ominaisuuksia ovat muodot, polut, teksti, markkerit, kloonit, alpha-kanava, muunnokset, liukuvärit, täyttökuviot ja ryhmittäminen. Inkscape tukee myös Creative Commons meta-dataa, noodien muokkausta, tasoja, monimutkaisia polkuoperaatioita, bittikarttakuvan jäljentämistä, tekstiä polulla, liukuvaa tekstiä sekä suoraa XML-editoimista. Import-toiminnolla voi tuoda JPEG-, PNG-, TIFF- ja muita kuvaformaatteja, ja export-toiminnolla tuottaa PNG- ja muita vektoripohjaisia formaatteja. Inkscape pyrkii olemaan täysin yhteensopiva XML-, SVG- ja CSS-standardien kanssa. SVG-tallennusvaihtoehdot ovat Plain SVG, Plain SVGZ, Inkscape SVG ja Inkscape SVGZ. (About Inkscape 2007.)

### Sodipodi

Sodipodi on ilmainen vektoripohjainen piirto-ohjelma, kuten CorelDraw tai Adobe Illustrator. Natiivina tiedostoformaattina se käyttää SVG:tä. Sen moderni näyttömoottori mahdollistaa antialiasoidun käyttöliittymän, alpha-läpinäkyvyydet, vektorifontit ja niin edelleen. Sodipodi on koodattu C-kielellä, käyttäen Gtk+ työkaluja sekä joitakin Gnome- tai KDE-kirjastoja. Ohjelmisto toimii suurimmalla osasta Unix- ja Windows-käyttöjärjestelmien versioista. (Sodipodi: About 2007.)

### Batik

Batik on Java-pohjainen työkalusto ohjelmistoille tai sovelluksille, joissa halutaan käyttää kuvia SVG-formaatissa eri tarkoituksissa, kuten näyttö, generointi ja manipulointi. Tarkoituksena on, että suunnittelijoilla on käytettävissä joukko perusmoduuleja, joita voi käyttää yhdessä tai erikseen tukemaan tiettyjä SVG-ratkaisuja. Näitä moduuleja ovat esimerkiksi SVG Parser, SVG Generator ja SVG DOM. Toinen tavoite on tehdä Batikista hyvin laajennettava ja sallia esimerkiksi kustomoitujen SVG-elementtien käsittely. Perusmoduulijoukon lisäksi käytettävissä on täysin toimiva SVG-selain, joka varmentaa eri moduulien yhteensopivuuden. (Batik SVG Toolkit 2007.)

## Sketsa SVG Editor

Sketsa on SVG-pohjainen vektorinpiirto-sovellus. Sen ominaisuuksiin kuuluu monia työkaluja sisällöntuotannon optimointiin, mm. ominaisuus-paletti, DOM-editori, lähdekoodi-editori, resurssi-editori, SVG:n muototyökalut, muunnostyökalut sekä lisäksi piirrostyökalut. SVG:tä käytetään natiivina tiedostomuotona. (Sketsa SVG Editor 5.0 2007.)

### **4.2.3 Konversiotyökalut**

Konversiotyökalut ovat sovelluksia ja liitännäisiä, jotka muuttavat tiedostot jostain grafiikkaformaattista SVG-formaattiin ja toisin päin. Tarjolla on monia milteistä tahansa formaatista konvertoimiseen; ladattavia sovelluksia, java-pohjaisia online-palveluja sekä liitännäisiä erilaisiin ohjelmistoihin mistä halutaan SVG-muodossa grafiikkaa ulos.

## **4.3 Mobiili SVG**

### **4.3.1 SVG:n soveltaminen mobiililaitteissa**

Matkapuhelinteollisuus valitsi vuonna 2001 SVG:n mobiililaitteiden grafiikka-alustan pohjaksi. Moni alan johtavista yhtiöistä liittyi ponnisteluihin tuottaen SVG Tiny ja Basic profiilit, yhteisesti kutsuttuna SVG Mobile, jotka kohdistuvat resurssi-rajoitteisille laitteille kuten matkapuhelimet ja PDA:t. 3GPP-organisaatio otti SVG Mobile spesifikaation vaadittavaksi grafiikkaformaattiksi uuden sukupolven puhelimille ja multimediaviestintään (MMS).

SVG-tuettuja puhelimia viedään jo maailmanlaajuisesti. Tällä hetkellä markkinoilla on 169 matkapuhelinmallia, joissa on SVG Tiny 1.1 tuki sekä vastaavasti 28 mallia, joissa on JSR 226. Näistä suurin osa on Nokian ja Sony Ericssonin puhelimia. (Shipping and Announced SVG Phones, 2007.)

Tänä päivänä Mobile SVG mahdollistaa mm. seuraavanlaisia palveluita ja ominaisuuksia:

- operaattoripalvelut (Vodafone ja KDDi/AU)
- parannellut J2ME-sovellukset (Motorola i870)
- selaus (eg. Sony Ericsson World)
- käyttöliittymäsuunnittelu (Nokia S60 Edition 3.0)
- mobiililaitteen personointi (suurin osa Sony Ericssonin, Benq/Siemensin ja Motorolan puhelimista)

(Ikivo: Mobile SVG, 2007.)

### 4.3.2 Mobiili-viewerit ja toolkitit

Mobile SVG Viewerit ovat SVG-katseluohjelmia mobiililaitteille. Näitä katseluohjelmia ja mobiilityökaluja, joilla tuotetaan sovelluksia ja sisältöä mobiililaitteille ja muita palveluja tarjoavat mm. BitFlash, Ikivo, Tinyline, Opera ja Sun Java. (Viewer Implementations, 2007; Mobile SVG Viewers, 2007.)

#### Bitflash

Bitflash SVG Tiny player, kaupallinen

SVG Tiny on oivallinen teknologia interaktiivisten sovellusten luomiseen sen tehokkaan mikroDOM (uDOM) tuen ansiosta. BitFlash tarjoaa uDOM:iin suoran yhteyden C-ohjelmointirajapinnan, ECMAScriptin tai Java-ohjelmointipintojen avulla. (Products: SVG Tiny Player, 2006.)

#### Ikivo

Ikivo SVG Player (svg 1.1) ja SVG Multimedia Player (svg 1.2), kaupallinen

Ikivo SVG Player on kompakti, tehokas ja joustava mobiili svg-ohjelmistotuote, joka on suunniteltu tuomaan täydelliset SVG-kyvyt mille tahansa mobiililaitteelle ja mobiilisovellukselle. Sen avulla on mahdollista lähettää, vastaanottaa, katsella ja käsitellä mobiili-SVG:tä, ja se mahdollistaa monia SVG-sovellustapahtumia kuten selaus, multimediamiestintä, dynaaminen käyttöliittymä ja dokumenttien katselu. (Ikivo: SVG Player, 2007.)

#### TinyLine

TinyLine SVG Tiny viewer ja SDK toolkit, freeware

TinyLine SVG työkalusto koostuu SVG Tiny sovelluksista ja ohjelmistosuunnittelutyökaluista J2ME laitteille. Sovellukset ovat itse asiassa erilaisia SVG Tiny J2ME-viewereitä. Kaksi pääominaisuutta erottaa Tinyline SVG viewerin muista; ensinnäkin se on puhdas J2ME CLDC 1.0 toteutus, joten se asennetaan kuten java-pelit. Toiseksi, sitä voidaan käyttää erilaisissa laitteissa aina matkapuhelimesta kannettavaan tietokoneeseen. (TinyLine SVG, 2006.)

#### Opera Mobile

Opera Mobile browser, kaupallinen

Operan mobiiliselain Symbian S60 ja Windows mobiililaitteille. Sen ominaisuuksiin kuuluu kaksi tapaa jolla selain asettuu näytölle, paranneltu käyttöliittymä, pikänäppäinkomennot sekä muut selaamista helpottavat toiminnot. (Opera for Symbian S60 Phones, 2007.)

## Sun Java

Sun Java WTK (Wireless Toolkit), ilmainen

Sun Javan WTK (Wireless Toolkit) on työkalusto langattomien sovellusten kehittämiseen, jotka perustuvat J2ME:n CLDC (Connected Limited Device Configuration) ja MIDP (Mobile Information Device Profile) säännöksiin. Nämä sovellukset toimivat matkapuhelimissa, kuluttajien PDA-laitteissa sekä muissa mobiililaitteissa. Työkalusto sisältää emulointiympäristöjä, suoritustehon optimointi ja säätö ominaisuuksia, tarvittavia dokumentaatiota ja esimerkkejä, joita kehittäjät tarvitsevat tuodakseen tehokkaita ja onnistuneita mobiilisovelluksia. (Sun Java Wireless Toolkit for CLDC, 2007.)

Matkapuhelinvalmistajien omat toolkit- ja SDK-paketit tarjoavat nimensä mukaan työkalut ja muun tarvittavan dokumentaation kunkin valmistajan puhelimille tarkoitettujen ohjelmistojen luomiseen. Laajin ja helpoiten löydettävissä oleva valikoima on Nokian ja Sony Ericssonin sivustoilla. Toki myös muilla matkapuhelinvalmistajilla, kuten Samsungilla ja Motorolalla, on omat SDK-työkalut ja dokumentaatiot, mutta ne ovat joko vaikeasti löydettävissä tai vaativat rekisteröitymistä.

## Sony Ericsson

Valikoimassa on erilaisia Java ME-työkaluja, plugin-lisäyksiä muihin työkaluihin, teknisiä dokumentteja sekä muita ohjeistuksia sovelluskehittäjille. (Java ME Platform Docs & Tools, 2007.)

## Nokia

Työkaluvalikoima on jaoteltu kahteen osaan: Series 40 ja S60 puhelinmallien tukemiin tekniikoihin. Sovelluskehittäjille on Java, C++ ja Carbide sekä Python pohjaisia työkaluja, ja sisällöntuottajille on erilaisia Web, Flash Lite ja muita media-työkaluja. (Nokia Tools and SDKs mobile applications development, 2007.)

#### 4.4 Selaintuki ja viewerit

PC:lle tarkoitettut viewerit ovat lähinnä selaimia, joissa on SVG-tuki joko natiivina tai liitännäisen avulla. Muutama itsenäinen SVG-katseluohjelma on myös olemassa, mutta suosituimpia tähän tarkoitukseen ovat selaimet.

##### Adobe SVG Viewer 3

Selaimen asennettava plug-in, jolla voidaan tarkastella SVG-kuvia ja web-sisältöä jossa on SVG:tä. Tukee SVG Full 1.1 -profiilin mukaisia ominaisuuksia joitakin poikkeuksia lukuun ottamatta. (Adobe SVG Viewer, 2007.)

##### Mozilla Firefox SVG Support

Mozillan SVG-toteutus on natiivi SVG-tuki Firefox-selaimessa. Firefox 2 tukee suurta osaa SVG 1.1:n elementeistä; ei-tuetut ominaisuudet ovat pääosin fontti-, suodatin- ja animaatio-elementtien ominaisuuksia.

(Mozilla SVG Project, 2006; Mozilla SVG Status, 2006; Mozilla Developer Center: SVG in Firefox, 2007.)

##### Opera 9

Opera tukee pääosaa SVG 1.1 Basic ja SVG 1.1 Tiny profiilien elementeistä. Suurin osa elementeistä on tuettu, jotkin tietyin rajoituksin. (Web Specifications Supported in Opera 9, 2007.)

##### Batik Squiggle SVG Browser

Squiggle on Batikin java-pohjainen, itsenäinen SVG-katseluohjelma, joka toimii yhteen Batikin Toolkit:in muiden ohjelmien kanssa. Tukee suurinta osaa SVG 1.1 spesifikaation elementeistä. Squiggle toimii kuten muutkin katseluohjelmat, tosin siinä on perustoimintojen lisäksi mahdollisuus pyörittää sisältöä sekä valita tekstiä kuvasta. (Batik SVG Toolkit, 2007.)

##### eSVG 2.5 Viewer ja Mobile Viewer

eSVG (embedded SVG) katseluohjelmat pohjautuvat SVG 1.1 spesifikaatioon, mobiili-profiilin Tiny ja Basic profiilikuvauksiin sekä SVG DOM 2-rajapinta spesifikaatioon. Tällä hetkellä katseluohjelmat tukevat SVG 1.1 Tiny ja Basic-profiilien elementtejä. eSVG Viewer on tehty PC-alustoille, ja eSVG Mobile Viewer versiot toimivat Symbian, Windows Mobile 5, Windows CE, Pocket PC, .NET, JAVA ja eCos alustoilla. (eSVG: Features, 2007.)

## 5 CASE: SVG-kuvan toteutus

### 5.1 Case:n kuvaus

Tässä casessa tutkitaan hieman erilaisia SVG:n luomistapoja sekä SVG-kuvien katselua ja käyttöä tietokoneella sekä mobiililaitteessa. Esimerkit ovat yksinkertaisia ja koostuvat lähinnä peruselementeistä. Tutkiminen suoritetaan kokeilemalla ja vertailemalla muutamaa erilaista tapaa toteuttaa SVG-kuvia sekä testaamalla lopputuloksia eri kohteissa. Lopuksi arvioidaan tutkinnan tuloksia.

Toteutustekniikoiden (Notepad vs. vektorigrafiikkaohjelmistot) vertailussa käytetään kuva-aiheena perusmuotoja erilaisilla viivan ja täytön tehosteilla. Näin mahdolliset eroavaisuudet nähdään helposti ja kuvat syntyvät nopeasti.

Katseluohjelmia ja skaalautuvuutta testatessa käytetään monimutkaisempaa kuvaa, joka on vektorigrafiikkaa SVG-muodossa. Näin nähdään miten skaalaaminen ja erilaiset näytöt vaikuttavat kuvan tarkkuuteen.

## 5.2 Toteutusvaiheiden määrittely

### Testi A

1. Testikuvan piirtäminen vektoriohjelmistoilla (kuvan elementtien hahmottamisen helpottamiseksi).
2. Tallentaminen SVG-tiedostoksi.
3. Kuvassa käytettyjen elementtien listaaminen.
4. Kuvan luominen Notepadilla koodaten.
5. Testaaminen selaimissa (Mozilla Firefox ja Internet Explorer, jossa Adoben SVG Viewer).
6. Vertailu ja tulosten kirjaaminen.

### Testi B

1. Testikuvan piirtäminen vektoriohjelmistolla ja tallentaminen SVG-tiedostoksi.
2. SVG-tiedoston mahdollinen muokkaaminen, koodin siivoaminen ja validointi.
3. SVG-kuvan upottaminen html-sivuun, erilliseen SVG-kuvaan linkittäminen html-sivulta.
4. SVGT-version tallentaminen kuvasta.
5. Testaaminen selaimilla sekä mobiililaitteessa.
6. Vertailu ja tulosten kirjaaminen.

Toteutuksissa käytettävät ohjelmistot: Adobe Illustrator CS2 (perinteinen vektorinpiirto-ohjelmisto), Inkscape (svg-pohjainen ohjelmisto), Notepad (käsinkoodaten).

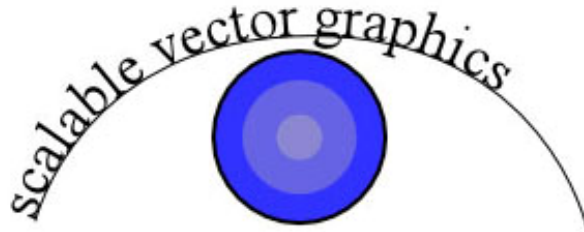
Testaamiseen käytettävät selaimet: Internet Explorer, Mozilla Firefox ja Opera, ja mobiililaitte: Sony Ericsson Z610i.

## 5.3 Toteutus

### 5.3.1 Testi A: Toteutustapojen vertailu

#### Toteutus

Kuva toteutetaan peruselementeillä. Kuvassa on ympyrä, jossa on täyttö ja reunaviiva sekä kaksi läpinäkyvää soikiota. Ympyrän päälle piirretään kaari, jota pitkin teksti kulkee.

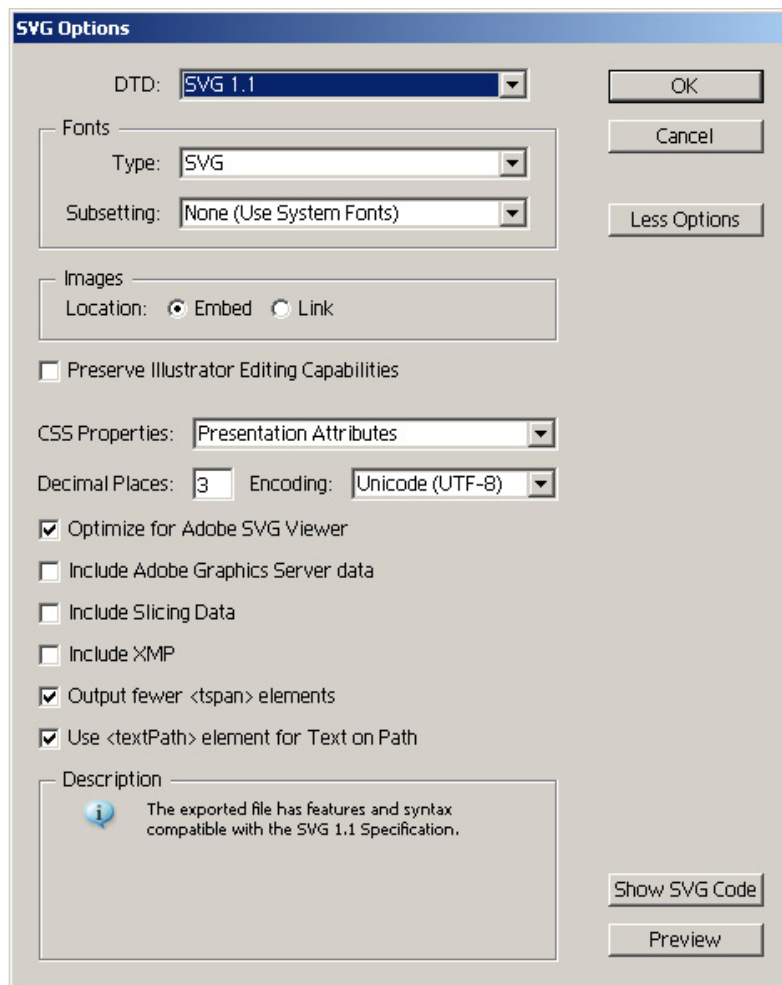


**Kuva 8 Testikuva**

Illustrator:

Piirretään ympyrä, jonka halkaisija on 30mm. Viivan paksuudeksi asetetaan 2pt ja väriksi musta. Ympyrän täytöksi valitaan sininen (#003CFF). Piirretään sinisen ympyrän päälle kaksi ympyrää lisää, joiden väriksi laitetaan valkoinen ja läpinäkyvyydeksi 30 % ja 50 %. Ympyrät linjataan samankeskisiksi. Yläpuolelle piirretään kaari, jonka päätepisteet ovat samalla tasolla kuin ympyrän alareuna. Kaarelle laitetaan kulkemaan teksti ”scalable vector graphics”. Kuva tallennetaan SVG-muotoon seuraavilla asetuksilla:

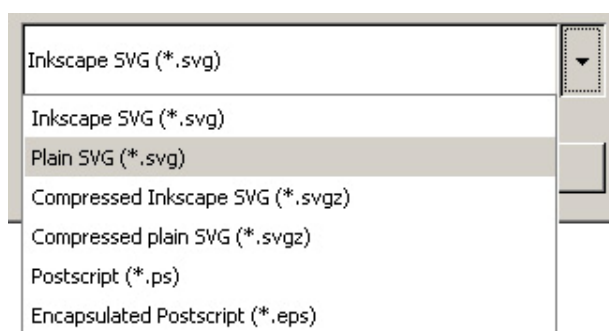




Kuva 9 Tallentaminen Illustratorista SVG 1.1.-muotoon

Inkscape:

Kuva piirretään samoin kuin edellä ja tallennetaan ”Plain SVG”-muotoon.



Kuva 10 Tallentaminen Inkscapesta Plain SVG-muotoon

Notepad:

Kirjoitetaan koodi (Ossi Nykäsen kirjan) esimerkkien mukaisesti ja vastaamaan edellä piirrettyjen kuvien arvoja ja tallennetaan SVG-muotoon.

## Testaus ja tulokset

Toisin kuin voisi olettaa, Illustratorilla luodun sinisen ympyrän koodi on lyhyempi ja selkeämpi kuin Inkscapeella luodun koodi. Ensin mainitussa ympyrät ovat circle-elementtejä, kun taas toisessa ne ovat monimutkaisesti määriteltyjä path-elementtejä. Dokumenttimäärittelyt on esitetty hieman eri tavoin.

Illustratorissa sininen ympyrä:

```
<circle fill="#3333FF" stroke="#000000" stroke-width="2"
stroke-miterlimit="3.8637" cx="151.963" cy="73.749"
r="41.642"/>
```

Inkscapeessa sininen ympyrä:

```
<path d="M 545.48238,378.59042 C 545.48238,444.97955
491.66336,498.79857 425.27423,498.79857 C 358.8851,498.79857
305.06608,444.97955 305.06608,378.59042 C 305.06608,312.20129
358.8851,258.38227 425.27423,258.38227 C 491.66336,258.38227
545.48238,312.20129 545.48238,378.59042 L 545.48238,378.59042
z " transform="matrix(0.438499,0,0,-
0.438499,11.55008,429.281)"
style="opacity:1;fill:blue;fill-opacity:1;stroke:black;stroke-
width:5.70126724;stroke-miterlimit:4;stroke-
dasharray:none;stroke-opacity:1" id="path1874" />
```

Firefox-selaimessa avattuna vektorigrafiikkaohjelmistoilla luodut kuvat asettuvat eri tavalla; Illustratorilla luotu asettuu vasempaan yläkulmaan, Inkscapeella luotu tietyn verran irti vasemmasta ja yläreunasta. Explorerilla avattuna kuvat asettuvat samoin, mutta Adobe SVG Viewer -liitännäisen ansiosta niitä pääsee tarkastelemaan lähemmin joko valitsemalla toiminto valikosta, joka tulee näkyviin painettaessa hiiren oikealla napilla kuvan päältä, tai näppäinkomennoilla. Zoomaaminen tapahtuu pitämällä Ctrl-nappia pohjassa ja shift-napin kanssa takaisin, liikuttaminen pitämällä Alt-nappia pohjassa ja alkutilanteeseen päästään valitsemalla ”Original View” valikosta. Operassa avattuna kuvat asettuvat myös kuten edellä, ja selaimen oma SVG-tuki sallii tarkastelun kuten Explorerissa.

Notepadilla tehty SVG-kuva antaa Firefoxissa vain virheilmoituksen, mutta aukeaa Explorerissa ja Operassa yläreunaan ja irti vasemmasta reunasta. Virheilmoitus johtuneee tiedoston määrittelyiden puutteellisuudesta tai Firefoxin SVG-tuessa olevista puutteista.

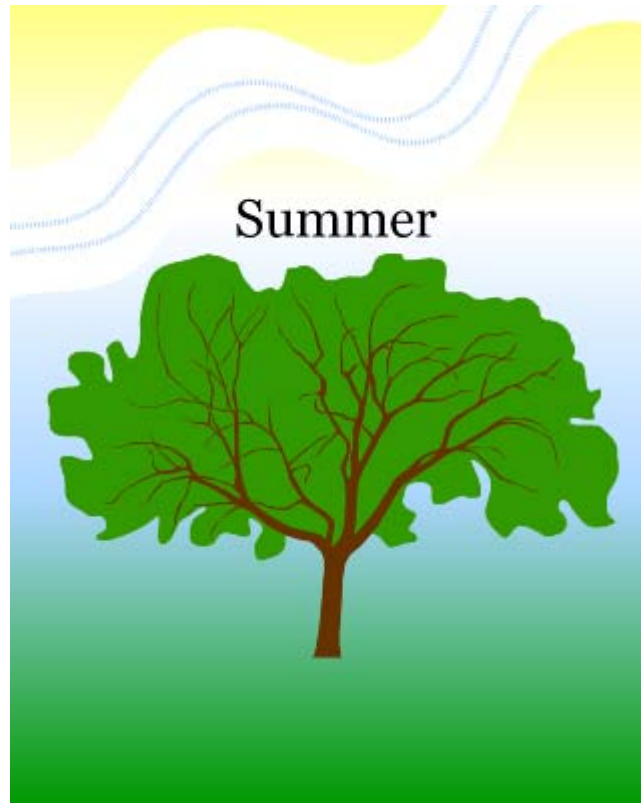
Eri versioiden kuvissa ei ole muutoin huomattavia eroja, paitsi Notepadilla koodatun kuvan kaari ei ole näkyvä, koska se on määritelty defs-elementissä. Kaikissa teksti on valittavissa kuvasta, paitsi Firefoxilla avattuna.

Huomattavaa on, että Internet Explorer ilmoittaa joka kerta ActiveX control-komponentista, joka täytyy sallia voidakseen tarkastella SVG-tiedostoja.

### 5.3.2 Testi B: Katseluohjelmien ja SVG-versioiden testaus

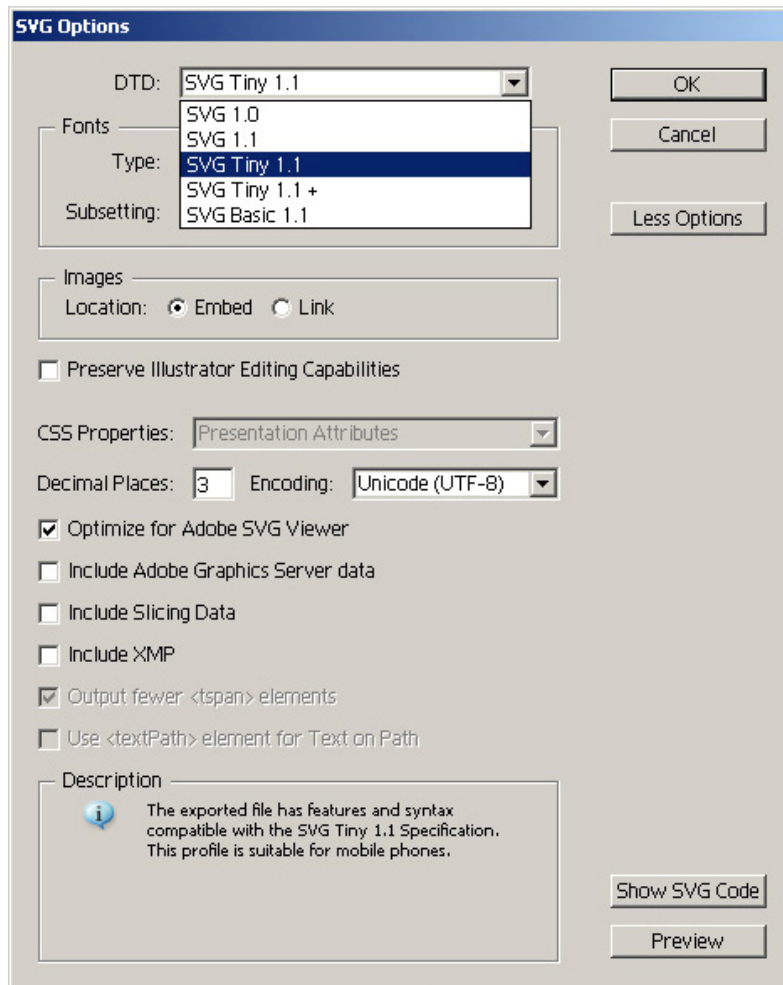
#### Toteutus

Piirretään vektorigrafiikkaohjelmistolla kuva, jossa on monipuolisempia muotoja kuin edellä sekä graafisia elementtejä ja tehosteita (liukuväri-täyttö, katkoviivaa, tekstiä) samassa kuvassa.



Kuva 11 Vektorigrafiikkaa testikuvaksi

Kuvasta tallennetaan SVG 1.1 versio selaimelle Adobe SVG Viewerille optimoituuna ja SVGT 1.1 versio mobiililaitteella tarkastelemista varten.



Kuva 12 Tallentaminen SVG Tiny 1.1.-muotoon

Lisäksi luodaan yksinkertainen html-sivu, jossa on upotettuna SVG-kuva `img-`, `object-` ja `embed-`elementeillä sekä linkki, jossa viitataan SVG-kuvaan.

### Testaus ja tulokset

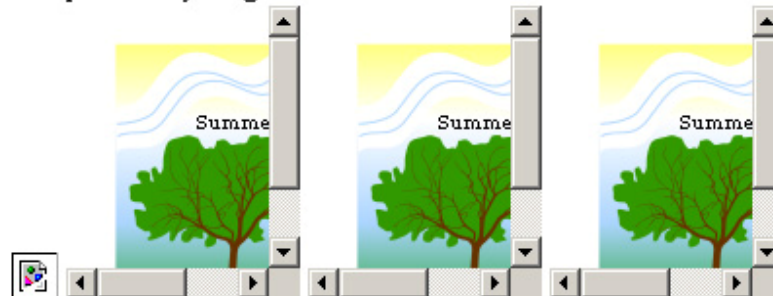
Kuvat avautuvat joka selaimessa ja näkyvät perustilassa oikein. Suurennettuna SVGT-versiossa kuvan ”ulkopuolelle” jatkuva kaistale ja katkoviiva eivät skaalaudu, koska SVG Tiny profiilissa ei ole Clipping-elementtiä tuettuna, joten ne näyttävät pikselisiltä.



Kuva 13 Pikselöitynyt kaari

HTML-sivussa kaikki paitsi `img`-elementti näyttävät `svg`-kuvan. Firefox-selaimella kuvat ovat eräänlaisessa ikkunassa, jossa on vierityspalkit, eivätkä siten näy kokonaan. Ongelma korjaantuu `width` ja `height`-attribuuttien arvoja kasvattamalla.

Alla pitäisi näkyä `svg`-kuva:



Jos ei näy, tässä [SVG-kuva](#) linkkinä.

Kuva 14 HTML-sivu, jossa `SVG`-kuvia upotettuna

Mobiililaitteessa, tässä tapauksessa Sony Ericssonin puhelimessa, `SVG`-kuvan tarkastelu tapahtuu lataamalla kuva puhelimen kuvakansioon ja avaamalla se; oletuksena kuva aukeaa puhelimen selaimessa. Kyseisessä puhelinmallissa on tuki `SVG`-muodossa olevalle grafiikalle, joten tiedoston avaamisessa ei esiinny ongelmia.

## 5.4 Lopputulosten arviointia ja yhteenveto

Testaamisessa käytetyt kuvat olivat suhteellisen yksinkertaisia, joten lopputulokset olivat ennalta arvattavia. Vastaan tulleet ongelmat kuvien luonnissa olivat joko niin pieniä, että ne tulivat selvitettyksi jo tekovaiheessa tai ne eivät haitanneet lopputulosta. Koodin kirjoittaminen aiheutti eniten päänvaivaa, ja esimerkkien kaltaisia kuvia tuskin kannattaa käsin kirjoitellakaan.

Tarkoituksena oli lähinnä kokeilla, miten SVG:n luominen tapahtuu käytännössä ja miten eri selaimet sitä näyttävät. Lopputulosten ja prosessin aikana tehtyjen havaintojen perusteella voi sanoa, että skaalautuvan vektorigrafiikan luominen on helppoa, jos käytettävissä on vektorigrafiikkaohjelmisto, josta voi tallentaa suoraan SVG-formaatissa. Muista ohjelmistoista tallentaminen tapahtuu joko exporttaamalla tiedoston toiseen ohjelmistoon tai jonkin konversiotyökalun avulla.

SVG-kuvan luominen koodaamalla on vaihtoehto silloin, kun ei ole vektorigrafiikkaohjelmistoa käytettävissä, tarvittava kuva koostuu peruselementeistä ja esittää jotakin kaaviota ja teknistä kuviota. Staattisten SVG-kuvien luomisessa ei välttämättä tarvitse kirjoittaa riviäkään koodia, tosin koodin ymmärtäminen helpottaa esimerkiksi tilanteissa, joissa kuvaa halutaan muuttaa vain vähän ja nopeasti, eikä tarvittavaa ohjelmistoa ole käytettävissä.

## 6 YHTEENVETO

Tämän opinnäytetyön tarkoitus on vastata kysymyksiin mitä SVG on, miten sitä luodaan, mihin se soveltuu ja missä sitä käytetään. Aiheessa on runsaasti tutkimista, monia näkökantoja ja kokonaisuuksia, joita voi lähteä tutkimaan syvemmin kuin tässä työssä. Vertailukohteiden (suomenkielisten tutkielmien) puutteen ja aiheen uutuuden vuoksi, aihetta on käsitelty yleisemmältä tasolta ja laajasti, jotta aiemmin esitettyihin kysymyksiin voidaan vastata. Tästä syystä myöskään itse formaattia ei juuri ole lähdetty vertailemaan jonkin muun vastaavan formaatin kanssa.

Käsittelyssä tuli esille monia seikkoja, miksi SVG on niinkin laajalti tuettu kuin se nykypäivänä on. Standardin avoimuus, graafisuus ja samalla teknisyyt sekä W3C:n ohella monien suurien yhtiöiden tuki, edesauttavat sen kehittämistä ja helpottavat erilaisten SVG-sovellusten toteuttamista. Suurena etuna on myös ohjelmistoriippumattomuus – SVG:tä voi luoda vaikka Notepadilla – sekä laaja tuki selaimissa, joilla SVG-tiedostojen tarkastelu pääasiassa tapahtuu erinäisten liitännäisten ja katseluohjelmien ohella.

SVG on monipuolinen standardi, jonka hyödyntäminen varmasti lisääntyy erityisesti mobiililaitteissa sitä myötä kuin tuki laajentuu sille ja SVG Tiny 1.2 profiili julkaistaan standardiksi. Sovellukset (kuten karttapalvelut ja tieteelliset kaaviot), joissa tiedon haettavuus ja esitettävyyt ovat oleellisia, käyttävät jo SVG:tä hyödyksi. Kun ottaa huomioon formaatin avoimuuden ja sen kanssa yhdisteltävissä olevien formaattien ominaisuudet ja mahdollisuudet, vain mielikuvitus on rajana, mitä kaikkea SVG:llä voidaan tehdä.

## LÄHTEET

### Kuvat

Kuvat 1 ja 2: Nykänen O. 2007, SVG – Skaalautuva vektorigrafiikka, Jyväskylä: Docendo.

Kuvat 3 ja 6: Web-Based SVG Map System: Design and Implementation, 2007. GIS Development [viitattu 5.11.2007]. Saatavilla:

[http://www.gisdevelopment.net/technology/gis/techgis\\_001a.htm](http://www.gisdevelopment.net/technology/gis/techgis_001a.htm)

Kuva 4: Clinical Study to SVG, 2004. SVGOpen.org [viitattu 5.11.2007]. Saatavilla:

<http://www.svgopen.org/2004/papers/ClinicalStudyToSVG/>

Kuva 5: SVG Scenarios using Microsoft Office Visio 2003, 2003. SVGOpen.org [viitattu 5.11.2007]. Saatavilla:

[http://www.svgopen.org/2003/papers/SVG\\_Scenarios\\_using\\_Microsoft\\_Office\\_Visio\\_2003/index.html](http://www.svgopen.org/2003/papers/SVG_Scenarios_using_Microsoft_Office_Visio_2003/index.html)

Kuva 7: BeeTV; The beeweb approach, 2007. Beeweb Technologies [viitattu 5.11.2007].

Saatavilla: <http://www.beeweb.com/mwt/index.php/products/beetv/>

### Kirjallisuus

Caplin, S. ja Banks, A. 2003, The Complete Guide to Digital Illustration, Cambridge: Ilex Press Ltd.

Nykänen, O. 2007, SVG - Skaalautuva vektorigrafiikka, Jyväskylä: Docendo

Tuikka, T. ja Kanala S. 2001, XML Ohjelmoinnin perusteet, Helsinki: Edita Publishing Oy

### Internet

About GIMP, 2007 [verkkójulkaisu]. GIMP [viitattu 27.2.2007]. Saatavissa:

<http://www.gimp.org/about/introduction.html>

About Inkscape, 2007 [verkkójulkaisu]. Inkscape [viitattu 27.2.2007]. Saatavissa:

<http://www.inkscape.org/>

About SVG, 2004 [verkkójulkaisu]. W3C [viitattu 29.10.2006]. Saatavissa:

<http://www.w3.org/Graphics/SVG/About.html>

Adobe Livedocs: About bitmap and vector graphics, 2007 [verkkójulkaisu]. Adobe [viitattu 27.2.2007]. Saatavissa: [http://livedocs.adobe.com/fireworks/8/fwhelp/g\\_s\\_02\\_b4.htm](http://livedocs.adobe.com/fireworks/8/fwhelp/g_s_02_b4.htm)



Adobe SVG Viewer, 2007 [verkkojulkaisu]. Adobe [viitattu 30.3.2007]. Saatavissa: <http://www.adobe.com/svg/viewer/install/> ;  
<http://www.adobe.com/svg/indepth/pdfs/CurrentSupport.pdf>

Adobe SVG Zone, 2007 [verkkojulkaisu]. Adobe [viitattu 30.3.2007]. Saatavissa: <http://www.adobe.com/svg/illustrator/illustrator.html>

Animation SVG 1.1, 2003 [verkkojulkaisu]. W3C [viitattu 19.10.2007]. Saatavissa: <http://www.w3.org/TR/SVG11/animate.html>

Applicability of CGM versus SVG for technical graphics, 2006 [verkkojulkaisu]. CGMO-open.org [viitattu 8.11.2007]. Saatavissa: <http://www.cgmopen.org/technical/cgm-svg-20030508-2.htm>

Batik SVG Toolkit, 2007 [verkkojulkaisu]. Apache [viitattu 27.2.2007]. Saatavissa: <http://xmlgraphics.apache.org/batik/>

Bring Scalable Vector Graphics to life with built-in animation elements, 2003 [verkkojulkaisu]. IBM [viitattu 7.10.2007]. Saatavissa: <http://www.ibm.com/developerworks/xml/library/x-tipsvganim/>

Cascading Style Sheets, 2007 [verkkojulkaisu]. W3C [viitattu 31.10.2007]. Saatavissa: <http://www.w3.org/Style/CSS/>

Chastain, S. 2007. Graphics Software – Vector [verkkojulkaisu]. About.com [viitattu 26.4.2007]. Saatavissa: <http://graphicssoft.about.com/od/glossary/1/blvector.htm>

Color SVG 1.1, 2003 [verkkojulkaisu]. W3C [viitattu 19.10.2007]. Saatavissa: <http://www.w3.org/TR/SVG/color.html>

Computer Graphics Metafile, 2007 [verkkojulkaisu]. Wikipedia [viitattu 26.10.2007]. Saatavissa: [http://en.wikipedia.org/wiki/Computer\\_Graphics\\_Metafile](http://en.wikipedia.org/wiki/Computer_Graphics_Metafile)

Concepts SVG 1.1, 2003 [verkkojulkaisu]. W3C [viitattu 19.10.2007]. Saatavissa: <http://www.w3.org/TR/SVG11/concepts.html>

CorelDRAW Graphics Suite X3, 2007 [verkkojulkaisu]. Corel [viitattu 27.2.2007]. Saatavissa: [http://www.corel.com/servlet/Satellite?c=Product\\_C1&cid=1150981051301&lc=en&pageName=uk%2FLayout](http://www.corel.com/servlet/Satellite?c=Product_C1&cid=1150981051301&lc=en&pageName=uk%2FLayout)

Document Object Model, 2005 [verkkojulkaisu]. SVG Wiki [viitattu 12.10.2007]. Saatavissa: [http://wiki.svg.org/Document\\_Object\\_Model](http://wiki.svg.org/Document_Object_Model)

eSVG: Features, 2007 [verkkojulkaisu]. eSVG [viitattu 5.11.2007]. Saatavissa: [http://esvg.ultimodule.com/bin/esvg/templates/default.asp?resolutionfile=templatespath|default.asp&area\\_3=pages/features](http://esvg.ultimodule.com/bin/esvg/templates/default.asp?resolutionfile=templatespath|default.asp&area_3=pages/features)

eSVG: What is it, 2007 [verkkojulkaisu]. eSVG [viitattu 5.11.2007]. Saatavissa: <http://esvg.ultimodule.com/bin/esvg/templates/splash.asp?NC=4366X>

Extensible Markup Language, 2007 [verkkojulkaisu]. Wikipedia [viitattu 7.11.2007]. Saatavissa: <http://en.wikipedia.org/wiki/XML>

Extensible Markup Language (XML), 2007 [verkkojulkaisu]. W3C [viitattu 8.5.2007]. Saatavissa: <http://www.w3.org/XML/>

Extensible Markup Language (XML) 1.0 (Fourth Edition). 2006 [verkkojulkaisu], W3C [viitattu 16.8.2006]. Saatavissa: <http://www.w3.org/TR/REC-xml/>

The Extensible Stylesheet Language Family (XSL), 2007 [verkkojulkaisu]. W3C [viitattu 18.10.2007]. Saatavissa: <http://www.w3.org/Style/XSL/>

Freehand MX at a glance, 2007 [verkkojulkaisu]. Adobe [viitattu 30.3.2007]. Saatavissa: <http://www.adobe.com/products/freehand/productinfo/overview/>

GeoMedia WebMap, 2007 [verkkojulkaisu]. InterGraph [viitattu 5.11.2007]. Saatavissa: [http://www.intergraph.com/gmwm/Key\\_Features.asp](http://www.intergraph.com/gmwm/Key_Features.asp)

Google Maps SVG Support, 2006 [verkkojulkaisu]. Ajaxian [viitattu 7.11.2007]. Saatavissa: <http://ajaxian.com/archives/google-maps-svg-support>

Ikivo: Mobile SVG, 2007 [verkkojulkaisu]. Ikivo [viitattu 12.3.2007]. Saatavissa: <http://www.ikivo.com/03mobileSVG.html>

Ikivo: SVG Player, 2007 [verkkojulkaisu]. Ikivo [viitattu 12.3.2007]. Saatavissa: [http://www.ikivo.com/pdf/Ikivo\\_SVG\\_Player.pdf](http://www.ikivo.com/pdf/Ikivo_SVG_Player.pdf)

Interactivity SVG 1.1, 2003 [verkkojulkaisu]. W3C [viitattu 19.10.2007]. Saatavissa: <http://www.w3.org/TR/SVG11/interact.html>

Introduction to SVG, 2003 [verkkojulkaisu]. InformIT [viitattu 25.9.2007]. Saatavissa: <http://www.informit.com/articles/article.aspx?p=99036&seqNum=2>

Java ME Platform Docs & Tools, 2007 [verkkojulkaisu]. Sony Ericsson [viitattu 21.8.2007]. Saatavissa: [http://developer.sonyericsson.com/site/global/docstools/java/p\\_java.jsp?cc=gb&lc=en&ver=4000&template=ps1&zone=ps&lm=ps1](http://developer.sonyericsson.com/site/global/docstools/java/p_java.jsp?cc=gb&lc=en&ver=4000&template=ps1&zone=ps&lm=ps1)

Linking SVG 1.1, 2003 [verkkojulkaisu]. W3C [viitattu 19.10.2007]. Saatavissa: <http://www.w3.org/TR/SVG11/linking.html>

Mobile SVG Profiles: SVG Tiny and SVG Basic, 2003 [verkkojulkaisu]. W3C [viitattu 27.9.2007]. Saatavissa: <http://www.w3.org/TR/SVGMobile/>

Mobile SVG Viewers, 2007 [verkkojulkaisu]. SVGi [viitattu 27.2.2007]. Saatavissa: <http://www.svgi.org/directory.html?type=2>

Mozilla Developer Center: SVG in Firefox, 2007 [verkkojulkaisu]. Mozilla [viitattu 19.9.2007]. Saatavissa: [http://developer.mozilla.org/en/docs/SVG\\_in\\_Firefox](http://developer.mozilla.org/en/docs/SVG_in_Firefox)

Mozilla SVG Project, 2006 [verkkojulkaisu]. Mozilla [viitattu 19.9.2007]. Saatavissa: <http://www.mozilla.org/projects/svg/>

Mozilla SVG Status, 2006 [verkkojulkaisu]. Mozilla [viitattu 20.9.2007]. Saatavilla: <http://www.mozilla.org/projects/svg/status.html>

Multimedia, 2007 [verkkojulkaisu]. SVG Wiki [viitattu 12.10.2007]. Saatavissa: <http://wiki.svg.org/Multimedia>

Multiple pages, 2004 [verkkojulkaisu]. W3C [viitattu 5.11.2007]. Saatavissa: <http://www.w3.org/TR/2004/WD-SVG12-20041027/multipage.html>

Nokia Tools and SDKs mobile applications development, 2007 [verkkojulkaisu]. Nokia [viitattu 5.11.2007]. Saatavilla: [http://www.forum.nokia.com/main/resources/tools\\_and\\_sdks/index.html](http://www.forum.nokia.com/main/resources/tools_and_sdks/index.html)

Opera for Symbian S60 Phones, 2007 [verkkojulkaisu]. Opera [viitattu 20.9.2007]. Saatavilla: <http://www.opera.com/products/mobile/products/s60/>

Products: SVG Tiny Player, 2006 [verkkojulkaisu]. Bitflash [viitattu 29.2.2007]. Saatavissa: [http://www.bitflash.com/prod\\_playerSVGT.html](http://www.bitflash.com/prod_playerSVGT.html)

Scalable Vector Graphics, 2007 [verkkojulkaisu]. Wikipedia [viitattu 26.2.2007]. Saatavissa: <http://en.wikipedia.org/wiki/SVG>

Scalable Vector Graphics (SVG) 1.1 Specification, 2003 [verkkojulkaisu]. W3C [viitattu 25.9.2007]. Saatavissa: <http://www.w3.org/TR/SVG11/>

Scalable Vector Graphics (SVG) Full 1.2 Specification, 2005 [verkkojulkaisu]. W3C [viitattu 13.4.2007]. Saatavissa: <http://www.w3.org/TR/SVG12/>

Scripting SVG 1.1, 2003 [verkkojulkaisu]. W3C [viitattu 5.11.2007]. Saatavissa: <http://www.w3.org/TR/SVG11/script.html>

Shipping and Announced SVG Phones, 2007 [verkkojulkaisu]. SVG.org [viitattu 8.2.2007]. Saatavissa: [http://svg.org/special/svg\\_phones](http://svg.org/special/svg_phones)

Sketsa SVG Editor 5.0, 2007 [verkkojulkaisu]. Kiyut [viitattu 27.2.2007]. Saatavissa: <http://www.kiyut.com/products/sketsa/index.html>

SMIL Animation, 2001 [verkkojulkaisu]. W3C [viitattu 4.11.2007]. Saatavissa: <http://www.w3.org/TR/smil-animation/>

Sodipodi: About, 2007 [verkkojulkaisu]. Sodipodi [viitattu 27.2.2007]. Saatavissa: <http://www.sodipodi.com/index.php3>

Sun Java Wireless Toolkit for CLDC, 2007 [verkkojulkaisu]. Sun Java [viitattu 4.10.2007]. Saatavissa: <http://java.sun.com/products/sjwtoolkit/>

SVG Document Object Model (DOM) SVG 1.1, 2003 [verkkojulkaisu]. W3C [viitattu 14.10.2007]. Saatavissa: <http://www.w3.org/TR/SVG11/svgdom.html>

SVG Phones, 2007 [verkkojulkaisu]. SVG Wiki [viitattu 27.9.2007]. Saatavissa: [http://wiki.svg.org/SVG\\_Phones](http://wiki.svg.org/SVG_Phones)

SVG Print 1.2, 2007 [verkkojulkaisu]. W3C [viitattu 1.5.2007]. Saatavissa: <http://www.w3.org/TR/SVGPrint12/>

Synchronized Multimedia Integration Language (SMIL) 1.0 Specification, 1998 [verkkojulkaisu]. W3C [viitattu 5.11.2007]. Saatavissa: <http://www.w3.org/TR/REC-smil/>

TinyLine SVG, 2006 [verkkojulkaisu]. Tinyline [viitattu 27.2.2007]. Saatavissa: <http://www.tinyline.com/svgt/index.html>

Tutorial – printing, 2007 [verkkojulkaisu]. Adobe [viitattu 30.3.2007]. Saatavissa: <http://www.adobe.com/svg/basics/printing.html>

Using GIS data intelligence on the web with Scalable Vector Graphics (SVG), 2005 [verkkojulkaisu]. SVGOpen.org [viitattu 27.9.2007]. Saatavissa: <http://svgopen.org/2005/papers/PublishingYourGISDataIntelligenceOnTheWebUsingSVG/index.html>

Using SVG to Create Compelling User Interfaces for Web Services, 2005 [verkkojulkaisu]. Idealliance [viitattu 4.10.2007]. Saatavissa: [http://www.idealliance.org/papers/dx\\_xml03/papers/02-04-05/02-04-05.html](http://www.idealliance.org/papers/dx_xml03/papers/02-04-05/02-04-05.html)

Vector Graphics, 2007 [verkkojulkaisu]. Wikipedia [viitattu 6.11.2006]. Saatavissa: [http://en.wikipedia.org/wiki/Vector\\_graphics](http://en.wikipedia.org/wiki/Vector_graphics)

Viewer Implementations, 2007 [verkkojulkaisu]. SVG Wiki [viitattu 6.11.2007]. Saatavissa: [http://wiki.svg.org/Viewer\\_Implementations](http://wiki.svg.org/Viewer_Implementations)

Web Specifications Supported in Opera 9, 2007 [verkkojulkaisu]. Opera [viitattu 5.11.2007]. Saatavilla: <http://www.opera.com/docs/specs/opera9/#graphics> ; <http://www.opera.com/docs/specs/svg/>

World Wide Web Consortium julkaisee SVG Tiny 1.2 -spesifikaation alustavana suositusehdotuksena, 2006 [verkkojulkaisu]. W3C [viitattu 8.10.2007]. Saatavissa: <http://www.w3c.tut.fi/press/2006/0810-svgtiny/index.html>

XML 10 kohdan tiivistelmänä, 2003 [verkkojulkaisu]. W3C [viitattu 2.10.2007]. Saatavissa: <http://www.w3c.tut.fi/translations/xml/xmlin10pts/>

## **LIITTEET**

LIITE 1: SVG mobiiliprofiilien moduulit -taulukko

LIITE 2: CD-ROM

Sisältö: Sähköiset lähteet, Opinnäytetyö pdf-muodossa, Tiivistelmä ja englanninkielinen abstrakti rtf-muodossa, SVG mobiiliprofiilien moduulit -Excel-taulukko, Adoben SVG Viewer -asennustiedosto, Casen esimerkit svg-muodossa.

Liite1: SVG:n mobiiprofiilien moduulit	SVG Full 1.1	SVG Basic 1.1	SVG Tiny 1.1
<b>Structure</b>			
Core Attribute Module	x (Full)	x	x
Structure Module	x (Full)	x	x (Basic)
Container Attribute Module	x (Full)		
Viewport Attribute Module	x (Full)	x	
<b>Style</b>			
Style Module	x (Full)	x	
<b>Painting</b>			
Paint Attribute Module	x (Full)	x	x (Basic)
Opacity Attribute Module	x (Full)	x	
Graphics Attribute Module	x (Full)	x	x (Basic)
<b>Hyperlinking</b>			
Hyperlinking			x
Hyperlinking Module	x (Full)	x	
XLink Attribute Module	x (Full)	x	x
External Resources Attribute Module	x (Full)	x	
<b>Conditional Processing</b>			
Conditional Processing Module	x (Full)	x	x
<b>Shapes</b>			
Shape Module	x (Full)	x	x
<b>Images</b>			
Image Module	x (Full)	x	x
<b>Text</b>			
Text Module	x (Full)	x	x (Basic)
<b>Color Profile</b>			
Color Profile Module	x (Full)	x	
<b>Gradients</b>			
Gradient Module	x (Full)	x	
<b>Patterns</b>			
Pattern Module	x (Full)	x	
<b>Clipping</b>			
Basic Clip Module	x (Full)	x	
<b>Masking</b>			
Mask Module	x (Full)	x	
<b>Markers</b>			
Marker Module	x (Full)		
<b>Fonts</b>			
Font Module	x (Full)	x	x
<b>Interactivity</b>			
Document Events Attribute Module	x (Full)	x	
Graphical Element Events Attribute Module	x (Full)	x	
Animation Events Attribute Module	x (Full)	x	
<b>Scripting</b>			
Scripting Module	x (Full)	x	
<b>Views</b>			
View Module	x (Full)	x	
<b>Filters</b>			
Basic Filter Module	x (Full)	x	
<b>Animation</b>			
Animation Module	x (Full)	x	x
<b>Extensibility</b>			
Extensibility Module	x (Full)	x	x
<b>Document Model</b>			
Basic Document Model Module		x	