

Toni Taipalus

**SKAALAUTUVAN VEKTORIGRAFIIKAN
SOVELLUSALUEET**

Tietojärjestelmätieteen
kandidaatintutkielma
27.2.2010

Jyväskylän yliopisto
Tietojenkäsittelytieteiden laitos
Jyväskylä

TIIVISTELMÄ

Taipalus, Toni Petteri

Skaalautuvan vektorigrafiikan sovellusalueet / Toni Taipalus

Jyväskylä: Jyväskylän yliopisto, 2010.

31 s.

Kandidaatintutkielma

Tässä tutkielmassa tarkastellaan skaalautuvan vektorigrafiikan standardin etuja ja heikkouksia sekä sen soveltumista erilaisiin käyttökohteisiin. Tarkoituksena on kartoittaa, mihin standardi sopii ja mihin se ei sovi. Arviointi pyritään tekemään ensisijaisesti kirjallisuuden pohjalta, mutta lisäksi on perehdytty standardin mukaisten dokumenttien rakenteeseen. Tarkoituksena on antaa käsitys standardin etujen ja heikkouksien vaikutukseen tilanteissa, joissa standardia harkitaan sovellettavaksi.

Tutkielmassa havaitaan, että standardin soveltuvuutta tiettyyn käyttökohteeseen on vaikea arvioida, jos tiedot sovellutuksen kohteesta ovat pintapuolisia. Ei siis voida yksiselitteisesti sanoa sen soveltuvan jollekin sovellusalueelle. Lisäksi huomattiin, että standardi toimii parhaiten muiden, erityisesti tekstipohjaisiin dokumentteihin perustuvien tekniikoiden kanssa.

AVAINSANAT: Skaalautuva, vektorigrafiikka, SVG, sovellusalue.

SISÄLTÖ

TIIVISTELMÄ	2
1 JOHDANTO	4
2 SKAALAUTUVA VEKTORIGRAFIikka.....	5
2.1 Käsitteistö.....	5
2.2 Edut	7
2.3 Heikkoudet	11
3 SOVELTUVUUS ERI KÄYTTÖKOHTEISIIN.....	15
3.1 Rakenteiset dokumentit	15
3.2 Opetuksen tuki.....	17
3.3 Maantieteelliset järjestelmät.....	20
3.4 Muut sovellutukset.....	22
4 YHTEENVETO	26
LÄHTEET.....	27
LIITE 1. SVG-dokumentti graafisessa- ja tekstimuodossa	31

1 JOHDANTO

Skaalautuvalla vektorigrafiikalla (*SVG*, Ferraiolo, Jun & Jackson 2009) tarkoitetaan tavanomaisemmasta bittikarttagrafiikasta (ts. rasterigrafiikka) poikkeavaa kuvaformaattia, joka mahdollistaa kuvan häviöttömän ja vapaan suurentamisen. Käytännössä tämä tarkoittaa sitä, että kuvia ja tekstiä voidaan suurentaa niiden muuttumatta epäselviksi. Skaalautuvuus toteutetaan korvaamalla pikselimatriisiin perustuva kuvanluonti vektoreihin perustuvilla muodoilla ja kuvioilla.

Laskentatehon kasvaessa ja mobiililaitteiden verkkoyhteyksien rajoittuneisuudesta johtuva tarve kaistankäytön minimointiin on tehnyt SVG:n entistä ajankohtaisemmaksi. Vektorigrafiikka on yleistynyt W3C:n (World Wide Web Consortium) standardoitua sen vuonna 2001. SVG-grafiikkakuvaus on XML (*Extensible Markup Language*, Bray ym. 2008) -pohjainen tekstidokumentti (esimerkki liitteessä 1), joka mahdollistaa sen käytön yhdessä muiden samaan tekniikkaan perustuvien teknisten toteutustapojen kanssa. Nykänen (2007, 3-4) listaakin SVG:n mahdollisiksi käyttökohteiksi web-julkaisujen ohella esitysgrafiikan, pelit, tieteelliset visualisoinnit ja graafiset käyttöliittymät.

Tämä tutkimus keskittyy W3C:n kaksikulotteiseen SVG (Scalable Vector Graphics) 1.1 -standardiin. Aluksi tutkimus käy läpi SVG:n edut ja heikkoudet, jonka jälkeen siirrytään arvioimaan niiden merkitystä erilaisissa käyttökohteissa kirjallisten lähteiden perusteella. Lopuksi arvioidaan SVG:n soveltuvuutta tarkasteltuihin kohteisiin, käyttökokemuksia, ja mitä menetelmiä SVG tulee mahdollisesti korvaamaan.

Tutkimuksessa huomataan, että SVG:n tehokkuus tietyssä käyttökohteessa on hyvin riippuvainen yksityiskohdista, eikä voida sanoa SVG:n olevan sopiva tai sopimaton johonkin sovellusalueeseen.

2 SKAALAUTUVA VEKTORIGRAFIikka

Ennen skaalautuvan vektorigrafiikan sovellusalueiden tutkimista on arvioitava sen heikkoja ja vahvoja puolia verrattuna yleisemmin käytettävään bittikarttagrafiikkaan. Tässä luvussa käydään läpi vektorigrafiikan käsitteistöä, etuja ja heikkouksia.

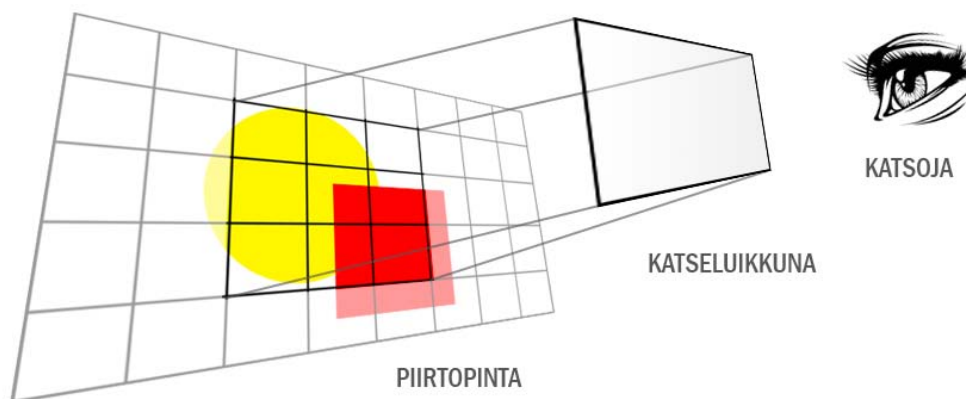
2.1 Käsitteistö

Bittikarttagrafiikkaan (ts. rasterigrafiikkaan) perustuvat kuvaformaattit käyttävät kuvatiedon tallentamiseen pikselimatriisia, jossa jokaisesta kuvapisteestä on tallennettu matriisiin väritieto. Menetelmä on standardi mm. digitaalisissa kameroissa. Vaughan-Nichols (2001) kuvaa yleisiä bittikarttaformaatteja kuten PNG (Portable Network Graphics) ja JPEG (Joint Photographic Experts Group) huonosti skaalautuviksi ja verrattain suurikokoisiksi jopa pakkausta käytettäessä. Suurin ero bittikarttagrafiikkaan verrattuna on, että vektorigrafiikka ei perustu absoluuttisiin arvoihin kuvapisteiden väritiedoista, vaan käyttää abstrakteja geometrisiä muotoja eli vektoreita kuvatiedoston esittämiseen. Vektorin sisältämä tieto koostuu sen alkupisteestä, suunnasta ja pituudesta. SVG-spesifikaatio kuvaa SVG:hen perustuvan kuvatiedoston luomisprosessia seuraavasti: kuvaa avattaessa ja sen kokoa muutettaessa (zoom) se piirretään kokonaan uudestaan. Tällöin jokainen vektori saa uuden kuvan kokoa vastaavan, skaalatun arvon.

Skaalautuvuudella tarkoitetaan SVG:n tapauksessa kahta asiaa. SVG-spesifikaatio ilmaisee skaalautuvuuden monimuotoisuuden seuraavasti. Ensinnäkin se on skaalautuvuutta grafiikan kannalta: elementtien dynaamisuutta koon muutoksiin. SVG:llä toteutetut elementit voidaan skaalata eri resoluutioihin, ja sama kuva voidaan näyttää aina laitteen (matkapuhelin,

tulostin, videotykki) täydellä resoluutiolla. Toiseksi se tarkoittaa teknologian skaalautuvuutta lukuisiin tiedostoihin, käyttäjiin ja sovelluksiin.

Otetaan esimerkiksi dokumentti, joka koostuu pelkästä suorasta viivasta, jonka alkupiste on dokumentin keskellä ja joka loppuu dokumentin ylälaitaan. Suoran piirtämiseen tarvittavat tiedot ovat siis muotoa: alkupiste koordinaateissa $(x/2, y/2)$, suunta 0 ja pituus $y/2$. Kun dokumentti sitten piirretään esimerkiksi resoluutioon 400x300, alkaa suora pisteestä (200, 150) ja sen pituus on 150 pikseliä. Vastaavaa menettelyä sovelletaan sen mukaan, minkä kokoinen dokumentista halutaan. Nykänen (2007, 67) vertaakin SVG:n piirtopintaa (canvas) äärettömän kokoiseen paperiin, johon dokumentin laatija kuvittaa näkemyksensä. Loppukäyttäjä tarkastelee paperia katseluikkunan (viewbox, liite 1, rivi 2) läpi, jonka katselukulman ja -etäisyyden dokumentin laatija on ikkunan attribuuteilla määrittänyt (kuvio 1).



Kuvio 1. Katseluikkuna ja piirtopinta. Katsoja näkee vain osan ympyrästä ja neliöstä.

Katseluikkunan määrittäykset eivät ole rajoitettuja tekstimuotoisessa SVG-dokumentissa annettuihin arvoihin, vaan ikkunaa voidaan lähentää, loitontaa ja liikuttaa (pan). Koska SVG-dokumentin elementit rakennetaan esitettäväksi (render) siinä järjestyksessä kuin ne tekstidokumentissa esitellään, määrittää katseluikkuna kaikkien elementtien koon. Liitteessä 1 tarkastellaan SVG-

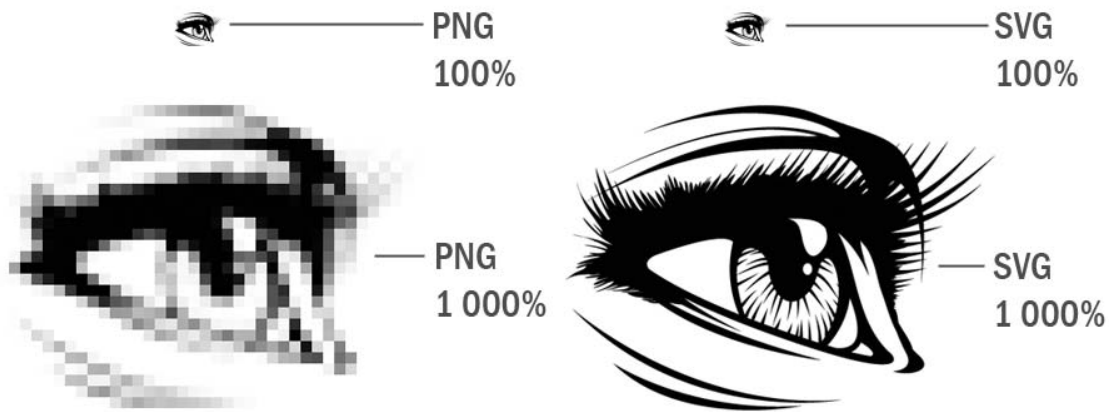
dokumenttia sekä sen loppukäyttäjälle esitettävässä muodossa, että tekstimuodossa. Tekstimuotoisessa osassa erotetaan dokumentin elementtien piirtojärjestys ja kuvassa sen vaikutus lopputulokseen. Ensin piirretään vaaleansininen tausta (rivit 16-17), sitten punainen ääriiviipiirros (rivit 18-20), suorakulmio tekstin ympärillä (rivit 21-22), teksti (rivit 23-26), bittikartta ja sen leikkaava maski (27-28) ja lopuksi bittikartalle kehykset (rivit 29-30).

2.2 Edut

SVG-standardilla on lukuisia etuja sekä muihin vektorigrafiikkaformaatteihin että bittikarttagrafiikkaan verrattuna: maksuttomuus, kuvien tarkka skaalautuvuus, monipuoliset graafiset ominaisuudet, pieni tiedostokoko, tekstipohjaisuus, yhteys XML-perheen välineisiin, semanttisuus, dynaamisuus, ja animoitavuus.

Koska SVG-standardi on W3C-konsortion kehittämä, on sen suunnitteluun osallistunut suuri joukko alan johtavia yrityksiä kuten Adobe, Corel, Apple, Macromedia, Microsoft, Sun ja Canon (SVG-spesifikaatio sekä Quint 2003). W3C:n perusteellisesta standardointiprosessimenettelystä (W3C, 2005) johtuen SVG:tä voidaan pitää luotettavana ja harkittuna kokonaisuutena. Konsortion toimintamallin mukaisesti SVG on myös maksuton standardi, joka antaa sille huomattavan edun kilpailijoihin kuten Adobe Flash (aiemmin Macromedia Flash) -formaattiin nähden.

Kuten sanottu, suurin ero vektorigrafiikalla bittikarttagrafiikkaan nähden on muotojen abstraktit arvot. Tämä ilmenee loppukäyttäjälle siten, että mahdollisuus kuvien suurentamiseen on lähes rajaton kuvan tarkkuuden kärsimättä. *Kuviossa 2* nähdään oikealla puolella SVG-dokumentin 10-kertainen suurennos, jossa ei huomata häviötä. Vasemmalla taas nähdään PNG-formaatissa tallennetun bittikartan vastaava suurennos, joka on pikselöitynyt.



Kuvio 2. Skaalautuvuus PNG- ja SVG -formaateissa.

Vaikka ajatus vektoreihin perustuvasta kuvatiedostosta saattaa aiheuttaa mielikuvan tarkasti rajoittuneista mahdollisuuksista kuvan piirrosta, Nykänen (2007), Quint (2003) sekä Mong & Brailsford (2003) luonnehtivat SVG:n graafisia ominaisuuksia monipuolisiksi. Standardi ei rajoitu pelkkiin vektoreihin ja muotoihin, vaan sisältää tehosteita kuten täyttekuvioita (pattern), läpinäkyvyyden (opacity), liukuvärejä (gradient), sumennuksia (blur) ja maskeja graafisten elementtien elävöittämiseen. SVG-spesifikaatio määrittelee SVG:lle myös alfakanavan eli läpinäkyvyyskartan (alpha channel). Alfakanava on edistynyt tehoste, joka simuloi kolmen luonnollisen värikanavan (RGB = Red, Green, Blue) lisäksi läpinäkyvyyskartan. Kartta toimii mustavalkoisena peitekuvana alkuperäisen kuvan päällä, ja jokaisen peitekuvan pikselin alla oleva alkuperäisen kuvan pikseli saa läpinäkyvyysarvon perustuen peitekuvan pikselin tummuuteen.

Bittikarttagrafiikkaan perustuvan kuvatiedoston koko määräytyy sen mittojen ja värimäärän mukaan (Nykänen, 2007, 124). Vektorigrafiikassa taas tiedoston koko määräytyy tekstipohjaisuudesta johtuen sen monimutkaisuudesta. Vektorigrafiikkaan perustuvaa tiedostoa voisi verrata tekstitiedostoon, jossa henkilö kuvailee sanoin, mitä kuva esittää: mitä enemmän kuvassa on asioita, sitä enemmän sen kuvailemiseen on käytettävä sanoja. Tästä johtuen

vektorigrafiikkaan perustuvat kuvat ovat erinomaisen skaalautuvuutensa lisäksi tiedostokooltaan pienempiä kuin bittikartat. SVG-spesifikaation mukaan SVG:n ensisijainen käyttökohde on verkossa sijaitsevien dokumenttien rikastaminen, ja verkon yli tapahtuvan tiedonsiirron kannalta pieni tiedostokoko onkin tavoiteltava lähtökohta. Tekstipohjaisuudesta johtuen dokumentin sisältämät tekstielementit ovat muokattavissa, toisin kuin rasterigrafiikassa. SVG-tiedostoformaatin (.svg) lisäksi W3C on kehittänyt standardin pakkaamiseen tarkoitetun .svgz-formaatin (SVG Zipped), joka SVG-spesifikaation mukaan mahdollistaa jopa 84% pakkaussuhteen. Haittapuoliksi mainitaan kuitenkin hitaampi tiedoston avaaminen sekä mahdollisesti puutteellinen tuki kolmannen osapuolen ohjelmistoissa.

W3C:n skaalautuvan vektorigrafiikkastandardin grafiikkakuvaus on XML-pohjainen tekstidokumentti. Tämä mahdollistaa standardin mukaisten tiedostojen luomisen yksinkertaisimmallakin tekstinkäsittelyohjelmalla, eikä näin ollen vaadi maksullisten kolmannen osapuolten ohjelmistojen hankkimista. Tämä seikka tuo SVG:lle jälleen yhden edun Flash-teknologiaan nähden. Kolmannen osapuolen ohjelmistojen käyttö voi kuitenkin nopeuttaa SVG-pohjaisten dokumenttien luomista huomattavasti. Quint (2003) arvioi SVG:n XML-pohjaisuutta erinomaiseksi ratkaisuksi web-integraation kannalta, sillä sen liittäminen osaksi tavallista HTML (*Hypertext Markup Language*, Raggett, le Hors & Jacobs 1999) -dokumenttia on suhteellisen vaivatonta. XML-pohjaisuuden ansiosta SVG-elementteihin voidaan myös lisätä graafisia tyylejä tyylisivujen (*CSS, Cascading Style Sheet*, Bos ym. 2009) avulla, ja XML:n puurakenne antaa kehittäjälle mahdollisuuden hierarkkisesti ryhmittää elementtejä dokumentin sisällä. Tekstipohjaisuuden ansiosta SVG ei nimestään huolimatta rajoitu vektorigrafiikkaan, vaan dokumenttiin voidaan myös liittää rasterigrafiikkaa joko dokumenttiin itseensä tai viittaamalla dokumentin ulkopuolisiin resursseihin kuten verkkosivuun tai tiedostoon. Tästä johtuen

SVG-dokumenteista voidaan tehdä dynaamisia linkityksen ja komentokielen avulla. XML-pohjaisuus myös takaa SVG:n neutraaliuden alustan ja laitteen suhteen.

Tekstipohjaisuudesta on myös hyötyä semanttisuuden kannalta. Ensinnäkin metatietojen lisääminen tekstipohjaiseen dokumenttiin ja niiden etsiminen on vaivatonta. Lisäksi dokumentille voidaan sen alustuksessa antaa kieli-attribuutti, joka Nykäsen (2007, 18) mukaan helpottaa sen löytymistä verkon hakukoneilla, joiden tulokset halutaan rajata kielen mukaan. Lisäksi dokumentti voi sisältää useita kieliversioita, ja pienimuotoisella komentokielen käytöllä dokumentista voidaan näyttää loppukäyttäjälle juuri hänelle sopiva kieliversio.

SVG-standardi mahdollistaa komentokielen käyttämisen. SVG-spesifikaation mukaan kehittäjä voi käyttää mitä tahansa ECMAScript-pohjaista komentokieltä, kuten JavaScript, JScript tai ActionScript. Komentokielen käyttäminen mahdollistaa SVG-dokumenttien dynaamisuuden (kuten esimerkiksi vaihtelun näyttökielen suhteen) ja interaktiivisuuden. Interaktiivisuus voi tarkoittaa esimerkiksi pienimuotoisia pelejä tai kartan koon muuttamista ja liikuttelua. Tämän lisäksi standardin mukaisia dokumentteja voidaan animoida W3C:n kehittämää SMIL-kieltä (*Synchronized Multimedia Integration Language*, Hoschka 1998) käyttäen. SMIL on tavanomaisesta kuvaperustaisesta (frame based) animaatiosta poiketen aikaperusteista (time based). Kuvaperustainen animaatio perustuu avainkuviin, jotka asetetaan animaation alku- ja loppupisteeseen. Animaation edetessä laitteisto laskee avainkuvien välissä tapahtuvien kuvien arvot ja piirtää ne loppukäyttäjälle. Kuvaperustainen animaatio on sidottu ennalta määrättyyn kuvataajuuteen (frame rate, frames per second), esimerkiksi televisiossa yleisen NTSC-standardin (National Television System Committee) mukaiseen 29,97 kuvaan sekunnissa. SVG:n aikaperustainen animaatio taas perustuu ennalta määrättyyn animaation

keston, jolloin animaatio piirretään suurimmalla kuvataajuudella, johon loppukäyttäjän laitteisto suorituskyvyltään riittää. Tämä mahdollistaa sekä animaatioiden näytävyyden lisääntymisen laitetehon kasvaessa että animaation skaalautuvuuden loppukäyttäjän laitteeseen.

Standardin eduksi voitaneen katsoa myös se, että dokumentti rakennetaan esitettäväksi (render) käyttäjän (client) luettavaksi (ts. kuvamuotoon tekstimuodosta) aina käyttäjän laitteella. Tämä siirtää rasituksen palvelimelta käyttäjälle, jolloin säästetään palvelimen verkkokaistan lisäksi muisti- ja prosessorikapasiteettia. Ongelma muodostuukin, jos käyttäjän laitteisto ei ole suorituskyvyltään riittävä SVG:n esittämiseen.

2.3 Heikkoudet

Skaalautuvalla vektorigrafiikalla vaikuttaisi olevan suuri joukko etuja yleisempään rasterigrafiikkaan nähden. Sen käytössä on myös haittapuolensa, jotka ovat suhteellisen nuoren iän lisäksi vaikeuttaneet sen yleistymistä. SVG vaatii esittämiseensä bittikarttagrafiikkaan verrattuna huomattavasti enemmän muistia ja prosessoritehoa, yleistymättömyytensä vuoksi se tarvitsee verkkoselaimeen asennettavan lisäsovelluksen toimiakseen (pois lukien Opera ja Safari), sen tekstinkäsittelyominaisuudet ovat verrattain tehottomat, sekä loppukäyttäjä voi nähdä dokumentin lähdekoodin. Näitä ongelmia tarkastellaan yksityiskohtaisemmin seuraavissa kappaleissa.

SVG:n suurin ongelma on sen vaatima laskentateho (Nykänen, 2007, 18). Morenon & de Oliveiran (2009) kokeellisten tulosten perusteella SVG vaatii selaimesta riippuen 6-31 prosenttiyksikköä enemmän prosessoritehoa käyttäjän päätteellä, kun vertailukohtana oli GIF (Graphic Interchange Format) -formaatti. 24-bittiseen bittikarttaan (BMP-24) verrattuna taas SVG tarvitsi käyttäjän päätteeltä jopa 25 prosenttiyksikköä vähemmän laskentatehoa. Palvelimella laskentatehon vaatimukset eivät eronneet merkittävästi GIF, PNG

ja JPG (Joint Photographic Group) -formaatteihin nähden. 24-bittiseen bittikarttaan verrattuna SVG tarvitsi laskentatehoa 1-9 prosenttiyksikköä vähemmän. SVG-standardin mukaisten dokumenttien voisi siis sanoa vaihtavan suuren tiedostokoon kasvaneeseen prosessorinkäyttöön, josta seuraa suurempi tarve tehokkaammalle laitteistolle, mutta vähäisempi tarve verkkokaistalle. Onkin hyvin tulkinnanvaraista, kumman resurssin säästämistä loppukäyttäjät enemmän arvostaa. Uudessa tietokoneessa on kirjoitushetkellä vähintään kaksisydinprosessori (dualcore), mutta mobiililaitteissa kuten matkapuhelimessa tai kämmentietokoneessa laskentateho on rajallisempaa. Toisaalta on myös tavallista, että mobiililaitteen käytössä oleva verkkokaista on rajoittuneempi.

Nykänen (2007, 13) arvioi SVG-standardin tekniikkaa hyvin tuetuksi huolimatta sen nuoresta iästä. Kirjoitushetkellä kuitenkin SVG:llä on täysi tuki vain Opera- ja Safari-verkkoselaimissa. Muiden verkkoselainten kuten Internet Explorerin, Mozilla Firefoxin ja Google Chromen tuki on olematon, jolloin ne eivät esitä SVG:tä lainkaan, tai vaillinaista, jolloin esimerkiksi staattiset dokumentit saatetaan esittää täydellisesti, mutta dynaamiset elementit ja animaatiot jäävät esittämättä. Selaimiin, joilla on puutteellinen tuki SVG:lle on asennettava Adoben maksuton SVG Viewer, joka mahdollistaa SVG:n näkymisen (Vaughan-Nichols 2001). Adobe Systems (2009) on kuitenkin lopettanut Viewerin tukemisen 1.1.2009 lukuisten markkinoilla olevien liitännäisten, kuten verkkoselainten natiivitukien vuoksi. Selaimista yhä yleisemmin oletuksena löytyvä tuki SVG:lle tekee erikseen asennettavan Viewerin turhaksi. Selaimen asennettavan liitännäisen lataaminen asettaa loppukäyttäjälle kynnyksen, jonka ylittäminen vaikeuttaa SVG:n yleistymistä. Vaughan-Nichols (2001) sekä Donney ym. (2008) huomauttavat, että huolimatta puutteellisesta selaintuesta SVG:tä tukevat ammattilaistason piirto-ohjelmat kuten Adobe Illustrator, Corel Draw ja Jasc WebDraw. Nykänen (2007) pitää

standardia yleisenä ja laajasti tuettuna, mutta Donney (2008) pitää selaintukea puutteellisena. Ottaen huomioon lähteiden julkaisuajankohdat, vaikuttaa siltä, että Nykäsen käsitys SVG:n yleistymisestä on positiivisesti väritynyt.

SVG-standardin tekstinkäsittelyominaisuudet jättävät toivomisen varaa (Nykänen 2007, 14), ja tämä ongelma on tuotu esille myös SVG-spesifikaatiossa. Nykänen näkee tekstinkäsittelyn ensisijaisena ongelmana sen sijoittamiseen liittyvän muotoilun puutteet: jokainen tekstirivi täytyy XML-syntaksin mukaisesti sijoittaa omaan elementtiinsä. Esimerkiksi tapauksessa, jossa tekstiä täytyy lisätä jo olemassa olevan tekstin sekaan, joudutaan teksti rivittämään manuaalisesti rivi kerrallaan uutta tekstisisältöä vastaavaksi. Ongelma korostuu, jos rivitys on ulkoasun kannalta erityisen tärkeää, esimerkiksi graafisessa suunnittelussa. SVG-spesifikaatio taas näkee tekstielementtien suurimpana puutteena kirjasintyyppien vihjeistykseen puuttumisen (font hinting, instructing). Microsoft Corporation (1997) määrittelee vihjeistykseen seuraavasti: vihjeistys on menetelmä, jonka perusteella kirjasintyyppin pikselit piirretään mahdollisimman tarkasti pienillä resoluutioilla. Ilman vihjeistystä SVG:n tekstielementit muuttuvat täten pienillä resoluutioilla vaikeaselkoisiksi tai jopa lukukelvottomiksi. SVG-spesifikaatio suosittelee muiden fonttitekniikoiden käyttämistä jos mahdollista, ja jos on todennäköistä että dokumenttia halutaan katsella pienellä resoluutiolla. Spesifikaatio suosittelee verkkokäyttöön esimerkiksi WebFont-formaattia, joka tukee vihjeistystä ja on suunniteltu erityisesti fonttitiedon esittämiseen mahdollisimman vähällä tilantarpeella.

XML-kielestä johtuva tekstipohjaisuus tuo standardille myös mahdollisen ongelman. Loppukäyttäjä saa SVG-dokumentin lähdekoodin vaivattomasti esiin millä tahansa verkkoselaimella. Kehittäjälle lähdekoodin - varsinkin monimutkaista komentokieltä sisältävän - antaminen saattaa tarkoittaa taloudellisen hyödyn menettämistä. Toinen vaihtoehto taas on dokumentin julkaisematta

jättäminen, mikä sekään ei vaikuta järkevältä vaihtoehdolta. Duignan, Biddle & Tempero (2003) arvioivat SVG:n varteenotettavimmaksi kilpailijaksi Adoben Flash-formaatin. Myös Flash perustuu vektorigrafiikkaan, mutta se tallennetaan binäärimuodossa, joka tekee siitä loppukäyttäjälle mahdotonta lukea. Formaatilekin puuttuu luonnollisesti tekstipohjaisuuden tuomat edut, mutta se on laajan tukensa ansiosta varteenotettava vaihtoehto, jos lähdekoodista ei haluta julkista.

Tässä luvussa esiteltiin SVG:n keskeisimmät käsitteet ja ominaisuudet ja käytiin läpi SVG-standardin etuja ja heikkouksia bittikarttagrafiikkaan nähden. Huomattiin myös, että SVG ei ole pelkästään kuvaformaatti, vaan monipuolinen graafinen laajennus XML-standardiin, animoitava, dynaaminen ja interaktiivisuuden mahdollistava kokonaisuus. Huolimatta standardin lukuisista eduista sillä on myös useita perustavanlaatuisia heikkouksia, jotka voivat rajoittaa sen soveltamista.

3 SOVELTUVUUS ERI KÄYTTÖKOHTEISIIN

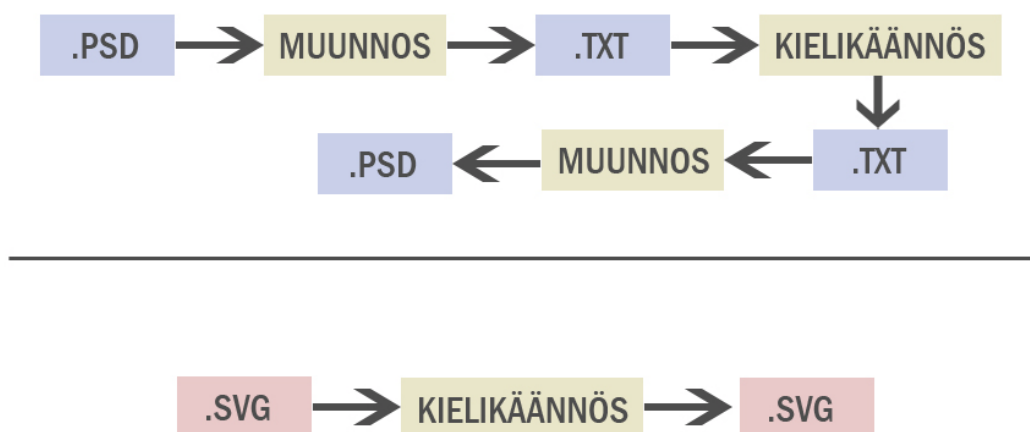
Skaalautuvan vektorigrafiikan käsitteistön sekä sen vahvuuksien ja heikkouksien kartoittamisen jälkeen siirrytään tarkastelemaan käyttökokemuksia ja standardin soveltuvuutta eri käyttökohteisiin. SVG:tä on koekäytetty useissa hyvin erilaisissa käytännön sovellutuksissa, joista merkittävimpiä tämä luku aluksi arvioi. Seuraavaksi tarkastellaan pienimuotoisempia käyttökohteita ja lopuksi arvioidaan SVG:n tulevaisuuden näkymiä.

3.1 Rakenteiset dokumentit

Mong & Brailsford (2003) arvioivat standardin ensisijaiseksi käyttökohteeksi verkkosivut, jotka perustuvat (X)HTML:ään ja CSS:ään, ja näiden dokumenttien rikastamisen (enrich) graafisella sisällöllä. XML-dokumenttiin voidaan liittää elementtikohtaista metatietoa. Jos kuvan luontiin käytetään SVG:tä, voidaan lisäksi tarkentaa dokumentin sisältämien elementtien sisältöä lisäämällä siihen esimerkiksi sijaintitieto. Esimerkiksi bittikarttamuotoon tallennetussa koulun luokkakuvassa voidaan XML-dokumenttiin tallentaa tiedot kuvassa olevista henkilöistä, mutta yksittäisen henkilön löytäminen suuresta joukosta saattaa olla työlästä. Kuvan luontiin käytettäessä SVG:tä jokainen oppilas on tekstimuodossa oma elementtinsä, ja elementit voidaan nimetä oppilaan mukaan.

Organisaatio, joka on keskittynyt graafisen sisällön tuottamiseen, käyttää useita graafisen alan ohjelmistoja, jotka eivät ole toistensa kanssa yhteensopivia (Donney ym. 2008). Eri ohjelmistojen käyttöön on tavallisesti syynä niiden erikoistuminen tiettyyn käyttötarkoitukseen, kuten vektori- tai rasterigrafiikkaan, valokuvankäsittelyyn, animointiin tai piirtämiseen. Eri ohjelmistot tuottavat erilaisia kuvaformaatteja, joiden tekstielementtejä tulisi voida muokata vaivattomasti jälkeinpäin esimerkiksi lokalisointia varten. Koska

tekstielementit sijaitsevat eri ohjelmistojen tukemissa kuvaformaateissa, jotka eivät usein ole yhteensopivia, vie niiden muokkaaminen suuresti aikaa. Nykyään käytössä olevaa menettelyä Donney ym. (2008) kuvaavat seuraavasti. Muokattavissa olevan kuvatiedoston (Photoshop PSD, CorelDraw CDR, Illustrator AI ym.) tallentamisen jälkeen juuri tiettyä tiedostoformaattia varten kehitetty muunnin (converter) erottelee tekstielementit XML- tai tekstitiedostoon, jossa kielenkääntäjä voi tekstin kääntää. Muokattu data muunnetaan sitten uudestaan kuvatiedostoon sopivaksi ja tallennetaan (kuvio 3). Prosessia kuvataan aikaa vieväksi ja tehottomaksi kahteen kertaan tapahtuvien kääntämisen ja tallentamisen vuoksi.



Kuio 3. Kielikäännösprosessin vaiheet tavanomaisella menettelyllä sekä SVG:llä.
 Kuva perustuu Donneyn ym. (2008) esityksiin.

Donney ym. (2008) esittävät vaihtoehdoksi tiedon tallentamista SVG-formaatissa. Kuten luvussa 2 mainittiin, useimmat ammattitason grafiikka-sovellukset tukevat SVG:tä tallennusmuotona ja mahdollistavat sen avaamisen. SVG:n käyttäminen poistaisi kokonaan raskaiden muuntimien käytön, ja eliminoisi tarpeen kääntäjien päivittämiseen aina, kun grafiikkasovelluksesta julkaistaan uusi versio. XML-pohjaisuuden tuoman semanttisuuden johdosta tiedon etsiminen niin dokumentin kuva- kuin tekstielementeistä helpottuisi bittikarttagrafiikkaan nähden. Tämän lisäksi SVG ei menetä muokattavuuttaan

tallennusvaiheessa, kuten esimerkiksi PSD:tä tallennettaessa rasteriperustaiseen formaattiin. SVG:n käyttö vaikuttaa siis varteenotettavalta vaihtoehdolta erityisesti lokalisointia vaativaa graafista sisältöä tuottavaan käyttöön. Kääntämisen helpottumisen lisäksi kuvatiedostosta tarvitsee tallentaa vain yksi monikielinen (multilanguage) tiedosto, eikä useaa bittikarttaversiota erikielisillä teksteillä. Tämä seikka vähentää entisestään standardin mukaisten tiedostojen vaatimaa tila- ja kaistavarausta.

SVG kuitenkin jättää toivomisen varaa tekstinkäsittelyssä, ja sen yleistymisen edellä mainituissa käyttötarkoituksissa edellyttää tämän puutteen korjaamista. Jos tekstielementtien luonti ja muokkaaminen on työlästä ja tehotonta, niitä sisältävää grafiikkaa kehittävät tuskin haluavat käyttää standardia työhönsä, vaikka se helpottaisi kielenkääntöprosessia. Samoin tekstin vihjeistämättömyydestä johtuvat luettavuusongelmat pienillä näytöillä rajoittavat skaalautuvuutta, johon koko standardi perustuu.

3.2 Opetuksen tuki

Nykyisin monet opetustapahtumat järjestetään verkon yli. Monimutkaisten asioiden hahmottaminen on kuitenkin opiskelijoille usein hankalaa, ja Bogaard, Vullo & Cascioli (2004) tutkivatkin SVG:n käyttöä opetuksen tukena pääasiassa sen animoitavuuden ja interaktiivisuuden tuoman havainnollistamisen kannalta. He kuvaavat SVG:llä ja komentokielellä tehtävää mallintamista helpoksi, dynaamiseksi ja helposti saavutettavaksi.

Bogaard, Vullo & Cascioli (2004) vertaavat SVG-standardia ja Adobe Flash -formaattia opetuksen kannalta. SVG:n eduksi lasketaan ensinnäkin standardin avoimuus. Standardin mukaista sisältöä tuottavat tahot ovat tuotteen myyjästä riippumattomia. Toiseksi on opiskelijoiden kannalta tärkeää, että dokumentin sisältämä tieto on helposti löydettävissä. Kolmanneksi dokumentin data voidaan erottaa itse esityksestä, jolloin dataa voidaan siis sijoittaa

dokumenttiin, mutta näyttää vain lähdekoodissa. Tämä tekee siitä skaalautuvaa laitetasolla, ja loppukäyttäjän näyttöresoluution perusteella voidaan päättää, kuinka paljon tietoa esitetään samalla kertaa. Viimeiseksi SVG on Flash-formaattiin nähden helposti kytkettävissä muihin verkon yli kommunikoiiviin standardeihin, kuten PHP:hen (PHP: Hypertext Preprocessor) tai SQL-tietokantaan (Structured Query Language) sekä XML-perustaisiin teknologioihin kuten CML:ään (Chemical Markup Language) kemiallisten yhdisteiden kuvaamiseksi tai MathML:ään matemaattisen kaavojen esittämiseksi. Lisäksi SMIL-animaation aikaperustaisuus helpottaa havainnollistamista, eikä dynaamisia elementtejä näin ollen tarvitse kehittää heikkotehoista laitteistoa käyttäviä opiskelijoita ajatellen.

Tavanomaisempaan rasterigrafiikkaan perustuvaan opetukseen nähden skaalautuvalla vektorigrafiikalla on useita etuja. Grafiikan laatu on parempaa ja kuvat skaalautuvia, sekä elementeistä voidaan tehdä interaktiivisia havainnollistamisen helpottamiseksi. Ketterlin, Mertensin & Vornbergerin (2007) interaktiivisen ja vektoriperusteisen opetusjärjestelmän käyttöön perustuva tutkimus tuo kuitenkin esille seuraavia perustavanlaatuisia ongelmia. Dokumentti, jossa yhdistyy videokuva ja SVG on huomattavan hidasta sekä ladata että rakentaa esitettäväksi (render). Ongelma korostuu, jos mallinnettava kokonaisuus edellyttää usean dokumentin näyttämisen samanaikaisesti. Samoin liikkuminen diojen (slide) välillä esimerkiksi Microsoft PowerPoint -esityksessä tapahtuu huomattavalla viiveellä. Myös tarve verkkoselaimen liitännäiselle nähdään kynnyksenä, joka tutkimuksen perusteella vaikuttaa negatiivisesti menetelmän vastaanottoon. Tutkimuksessa kokeiltiin myös Adobe Flash 9 -tekniikkaan perustuvaa Flex 2 -teknologiaa. Flex osoittautuu SVG-standardia huomattavasti tehokkaammaksi tässä käyttötarkoituksessa. Tutkimus osoittaa Flexin vaativan niinkään liitännäisen verkkoselaimen, mutta Flash-liitännäisen katsotaan löytyvän loppukäyttäjältä usein jo

ennestään. Käytännön kokeiluun osallistuneet myös raportoivat Flex-teknologialla toteutetun käyttöliittymän nopeudesta SVG:hen nähden.

Flash-teknologian parempi suorituskyky johtuu sen tallentamisesta binäärimuotoon (Duignan, Biddle & Tempero 2003). Ketterlin, Mertensin & Vornbergerin (2007) käyttökokemuksiin perustuvan tutkimuksen mukaan vaikuttaakin siltä, että loppukäyttäjät arvostavat jouhevuutta ja nopeaa vasteaikaa enemmän kuin XML:n tuomia etuja.

Teknologialtaan molemmat standardit ovat hyvin samanlaisia, mutta samalla niiden välillä on perustavanlaatuisia eroavaisuuksia, joiden vahvoja puolia on hankala jollei mahdoton yhdistää. Molemmat perustuvat vektorigrafiikkaan sisältäen sen tuomat edut kuten skaalautuvuuden ja pienen tiedostokoon. Samoin molemmat mahdollistavat animaatiot ja interaktiivisuuden. Teksti-pohjaisuus tekee SVG:stä helposti muokattavan, tiedonhausta helppoa ja erittäin yhteensopivaa muiden standardien kanssa. Samalla se kuitenkin heikentää suorituskykyä tehden dokumenttien rakentamisesta esitettäväksi Flash-teknologiaan verrattuna hidasta. Vaikka myös Flashiin voidaan liittää metatietoa, ei se korvaa XML-pohjaisen dokumentin mahdollisuutta vapaaseen lähdekoodin selaamiseen. Mitä monipuolisempia dokumentteja kehittäjä haluaa tehdä, sitä suuremmaksi kasvaa aika, joka niiden rakentamiseen loppukäyttäjää varten kestää. SVG:n tapauksessa ajat ovat Flashiin nähden suuremmat, ja rakentamisen hitaus korostuu mitä suurempia dokumentteja tarvitaan. Kuitenkin suurissa asiakokonaisuuksissa SVG:n yhteys XML-perheen välineisiin korostuu, jos kehittäjä haluaa dokumenttiinsa esimerkiksi aiemmin mainittuja matemaattisia kaavoja (MathML) tai kuvia kemiallisista yhdisteistä (CML). Flash-teknologiassa vastaavia välineitä ei ole. Tästä päätellen on erittäin tapauskohtaista, kumpi tekniikka on tehokkaampaa opetuskäytössä ja erityisesti mallintamisessa. Probertsin ym. (2001) mukaan tekniikoiden samankaltaisuudet johtuvat osittain samoista kehittäjistä (Adobe ja sen

edustajat) ja osittain taas SVG-standardin kehityksen pohjautuminen vanhempiin vektorigrafiikkastandardeihin kuten EPS:ään (Encapsulated PostScript).

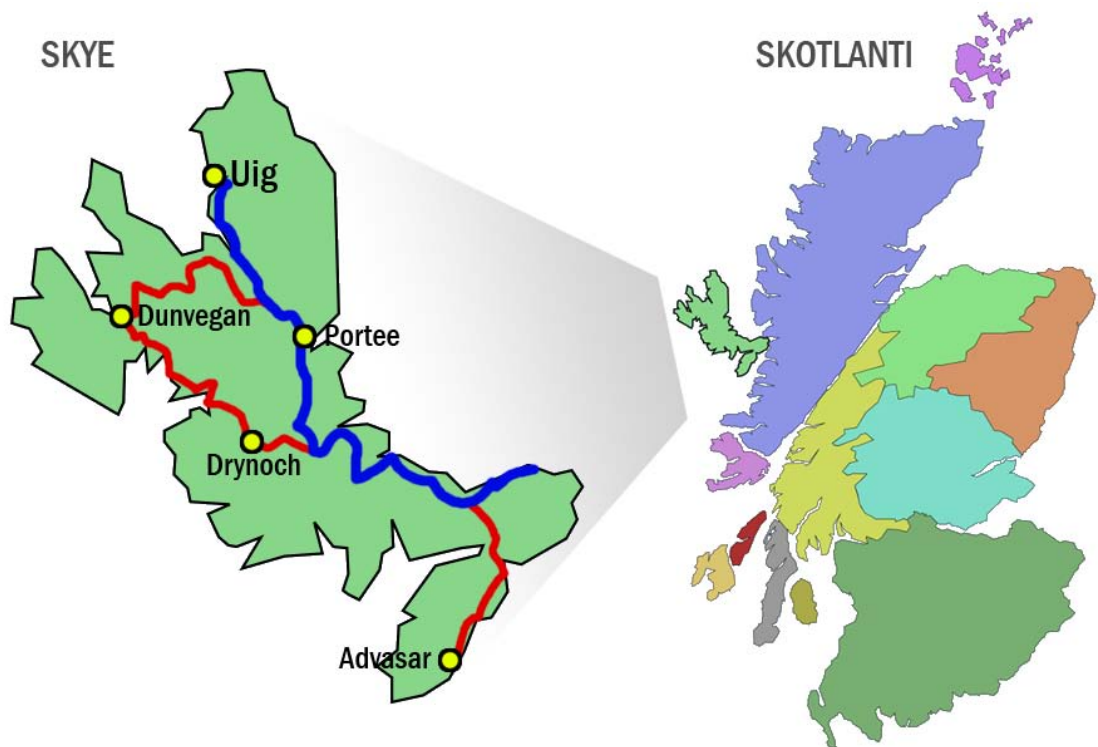
Toinen SVG:n kilpailija verkkoluentojen tukimateriaalina on virtPresenter-ohjelmisto (Mertens, Ketterl & Vornberger 2007). VirtPresenter muuntaa PowerPoint-esitykset verkkokäyttöön sopiviksi esityksiksi, joihin tallennetaan myös luennoitsijan puhe. Opiskelijat voivat virtPresenterin käyttöliittymän kautta nähdä kunkin kalvon sisällön ja hypätä tarpeettomaksi luokittelemansa aineiston yli tai käydä uudelleen läpi vaikeaksi kokemansa asiat. PowerPoint-esitykset eivät kuitenkaan ole grafiikaltaan skaalautuvia, ja vaativat SVG:tä enemmän tallennustilaa ja verkkokaistaa.

3.3 Maantieteelliset järjestelmät

Skaalautuva vektorigrafiikka vaikuttaa sopivan ominaisuuksiensa perusteella hyvin graafiseen osoite- tai tiekarttapalveluun: kartat ovat muodoiltaan suhteellisen yksinkertaisia, vektorigraafinen kartta on erinomaisesti tarkennettavissa ja tekstipohjaisuus takaa tiedon helpon löytymisen. Lisäksi, kuten alakohdassa 3.1 tuli esille, erikielisten versioiden lokalisointi helpottuu. Christel & Huang (2001) kuvailevatkin SVG-standardia erinomaiseksi juuri karttasovelluksiin, koska ranta- ja rajaviivat pysyvät tarkkoina vaikka käyttäjä tarkentaisikin karttaa erityisen paljon.

Huang, Li & Gartner (2008) näkevät teknologian kehittymisen johtavan yhä enemmän siirtymiseen verkossa toimiviin maantieteellisiin tietojärjestelmiin (Web-based Geographic Information System, WebGIS). Man ym. (2009) katsovat kuitenkin jo olemassa olevien tekniikoiden olevan raskaita, ja tarjoavan usein liiankin monimutkaisia työkaluja loppukäyttäjälle. Ratkaisuksi he perustelevat SVG:tä sen tekstipohjaisuuden tuoman helpon muokattavuuden vuoksi. Eritasoisia ominaisuuksia vaativille loppukäyttäjille

voidaan räätälöidä sopiva versio karttapalvelusta. Lisäksi he esittävät, että SVG:n elementteihin kuuluvat sijaintitiedot voidaan pienimuotoisella komentokielen käytöllä muuttaa jo olemassa oleviin karttajärjestelmiin sopiviksi koordinaateiksi. Molemmat tutkimukset keskittyvät SVG:n etuihin monimutkaisissa tietojärjestelmissä, jotka on liitetty tietokantaan. SVG:n tekstipohjaisuus tekee tiedon vaihtamisesta tietokannan kanssa huomattavasti kitkattomampaa kuin muiden olemassa olevien GIS-järjestelmien. Yrityskäytössä SVG:llä näyttää siis olevan hyvät mahdollisuudet vakiintua yleiskäyttöisenä standardina.



Kuvio 4. Esimerkki, kuinka karttajärjestelmä voisi toimia. Käyttäjä tarkentaa valintansa Skyen saareen, josta hän saa tarkempaa tietoa. Tarkentamista jatketaan tarpeen mukaan.

Navigaattoreiden karttasovellukset ovat yhä useammin kolmiulotteisia, ja näkymä päivittyy reaaliajassa. Vaikka karttojen muodot voidaan pitää yksinkertaisina ja siten nopeasti rakennettavissa (kuvio 4), tuo animaatio silti huomattavaa lisäkuormitusta prosessorille, joka navigaattorissa on teholtaan rajallinen.

Kolmiulotteisuudesta Duignan, Biddle & Tempero (2003) kirjoittavat seuraavasti. SVG on standardina lähtökohtaisesti kaksiulotteinen, mutta lisäulottuvuuksia voidaan mallintaa komentokielellä. Se ei kuitenkaan ole suositeltavaa, sillä kolmannen ulottuvuuden simulointi pelkkien X- ja Y-koordinaattiakseleiden arvojen perusteella vaatii suuria määriä laskentatehoa. Tämä yhdistettynä liikkuvan kuvan vaatimaan laskentatehoon tekee SVG:n käytöstä reaaliaikaista kuvaa mallintavissa järjestelmissä tehotonta.

Standardiin perustuvat sovellutukset ovat kuitenkin omiaan juuri navigaattorin kaltaisissa laitteissa, joissa dokumentin lähdekoodi tarkastelu on loppukäyttäjälle hankalaa johtuen tiedostoselaimen (explorer) puuttumisesta. Samoin SVG:n vähäinen tilankäyttö ei kuormita laitteen rajallista kapasiteettia ja SVG:stä on olemassa ominaisuuksiltaan karsittuja versioita juuri mobiililaitteita varten. Ongelmaksi nousee vain fonttien vihjeistämättömyys, joka tekee pienillä näytöillä teksteistä vaikeasti luettavaa. Tekstiin ja sen käsittelyyn liittyvät ongelmat näyttävätkin nousevan standardin suurimmaksi ongelmaksi.

3.4 Muut sovellutukset

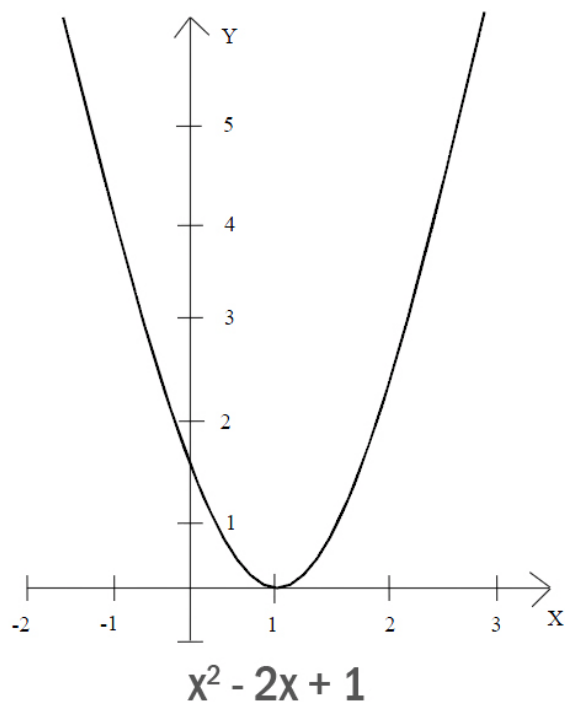
Skaalautuvan vektorigrafiikan standardia on sen nuoresta iästä huolimatta kokeiltu monissa ympäristöissä vaihtelevin tuloksin. Seuraavassa tarkastellaan SVG:n soveltuvuutta lääketieteellisten kuvien esittämiseen ja merkkäämiseen, monimutkaisen merkistön ryhmittelyyn ja matemaattisten käyrien esittämiseen. Lääketieteellisessä käytössä tarve suuriresoluutioisille ja ennen kaikkea tarkoille kuville on suuri. Vektorigrafiikan käytöllä kuvista on mahdollista saada huomattavasti paremmin tarkkuutensa säilyttäviä kuin bittikartta-grafiikkaan perustuvista kuvista ja interaktiivisuutta voidaan hyödyntää samalla periaatteella kuin karttasovelluksissa. Pienimuotoisella komentokielen käytöllä anatomisia kuvia voidaan tarkentaa aina solutasolle asti kuvion 4 periaatetta mukaillen. Mohammed & Fiaidhi (2005) korostavat erityisesti XML-

perustaisuuden tuovan SVG:n helposti yhdistettäväksi lääketieteellisessä käytössä käytettäviä tietojärjestelmiä. Tutkimus kuitenkin muistuttaa, että röntgenkuvat perustuvat valokuvatekniikkaan, joka tallentaa bittikarttagrafiikkaa. Bittikarttagrafiikkaa vektoroimalla ei kuvaan saada lisää informaatiota, vaikka suurennoksen tulos onkin enemmän silmää miellyttävää. SVG-formaattiin kääntämällä kuvat saadaan kuitenkin mahtumaan pienempään tilaan, ja näin säästetään sekä tallennustilaa että verkkokaistaa, samalla saadaan käyttöön tekstipohjaisuuden tuomat edut. Wang, Rabsch & Liu (2007) katsovat SVG:n olevan myös erinomainen vaihtoehto lääketieteellisten kuvien merkkaukseen niin tekstillä kuin grafiikan korostuksella rajauksin ja tarkennuksin. Lääketieteellisessä käytössä yleistymiseen SVG:llä ei näyttäisi olevan juuri esteitä.

Tämän kohdan tarkastelu perustuu pääosin Quintin & Apelin (2005) esitykseen. Japanin kieltä opiskellessa opiskelijan täytyy tuntea kahden aakkosjärjestelmän (*hiragana* ja *katakata*) lisäksi tuhansia merkkejä sisältämä *kanji*-merkistö. Kanji-merkit ovat aakkosjärjestelmän merkkeihin verrattuna monimutkaisia (esimerkiksi sana *kirkas* ”明”). Vaikka tekstiä voidaan tuottaa pelkillä kahdella aakkosjärjestelmällä, kanjia sisältämää tekstiä on mahdoton lukea sitä tuntematta. Kanji-merkkejä voidaan kategorisoida niiden löytämisen helpottamiseksi. Kanji jaetaan ryhmiin perustuen siihen, kuinka monta viivaa käyttäen se piirretään, ja miten viivat lopetetaan. Vektoroimalla kanji-merkistö, ja käyttämällä tähän erityisesti SVG-standardia, saadaan tietokanta, josta opiskelija voi etsiä tiettyä kanji-merkkiä joko tekstuaalisesti sen tuntomerkkien perusteella tai SVG:hen tallennettuja muotoja vertaamalla. SVG sopii tehtävään muita vektorigrafiikkastandardeja paremmin jälleen tekstipohjaisuutensa ansiosta. Jokaista kanji-merkin sisältämää viivaa varten luodaan dokumenttiin uusi elementti, ja elementtien lukumäärän perusteella voidaan alustavasti rajata haluttua joukkoa. SVG näyttää soveltuvan hyvin edellä mainittuun, joskin

pienimuotoiseen käyttötarkoitukseen, eivätkä sen huonot puolet pääse korostumaan.

MathML (Mathematical Markup Language) on matemaattinen kuvauskieli ja XML-sovellus, joka on tarkoitettu matemaattisten kaavojen esittämiseen. Bruhn & Burton (2004) tarkastelevat tutkimuksessaan SVG:n yhdistämistä MathML:n kanssa muun muassa käyrien piirtämistä varten seuraavin tuloksin. Tutkimuksessa luodaan MathML:llä kirjoitetun toisen asteen polynomin pohjalta käyrä koordinaatistoon (kuvio 5). Käyrän luominen SVG:llä valmiin MathML-merkkauksen pohjalta onnistuu jopa ilman komentokielen käyttöä.



Kuvio 5. Toisen asteen polynomi ja sen tuottama käyrä. Bruhn & Burton (2004).

Tutkimuksen esimerkin dokumenttia voisi kuitenkin huomattavasti pienentää komentokieltä ja tyylisivua käyttämällä. Tutkimuksen perusteella SVG:tä voi pitää erinomaisena laajenuksena MathML:lle. Koska MathML luo pelkästään fonttitiedostoon perustuvia merkkejä, on se SVG:n tavoin skaalautuvaa, ja tekniikoiden yhdistäminen luo täysin skaalautuvia dokumentteja. Lisäksi molemmat ovat maksuttomia ja tarvitsevan tekstipohjaisuutensa ansiosta vain

vähän tallennustilaa. Matemaattisten kuvioiden jäädessä suhteellisen yksinkertaisiksi ei ole syytä olettaa, että standardin suuri tarve laskentateholle tulee ongelmalliseksi, eikä käyrän piirtämiseen tarvita kuin yksi elementti. Kolmiulotteisessa koordinaatistossa SVG:n puutteet kuitenkin tulevat vastaan, eikä sillä voida ilman komentokielen käyttöä esittämään esimerkiksi parabo-
loideja.

4 YHTEENVETO

Tässä tutkielmassa tarkasteltiin World Wide Web Consortiumin Scalable Vector Graphics -standardia erityisesti sen sovellusalueita silmällä pitäen. Aluksi arvioitiin kirjallisiin lähteisiin perustuen standardin etuja ja heikkouksia sekä bittikarttagrafiikkaan että varteenotettavimpaan kilpailijaan, Adoben Flash-formaattiin nähden. Lopuksi tarkasteltiin erilaisia käyttökohteita, joissa SVG-standardista on saatu käyttökokemuksia.

SVG-standardin keskeisimmän puutteen havaittiin olevan suuri tarve laskentateholle johtuen standardin perustumisesta abstrakteihin muotoihin. Muut merkittävät ongelmat liittyivät fonttien vihjeistämättömyyteen ja tekstinkäsittelyn puutteellisiin ominaisuuksiin. Erityisesti tekstiin ja fontteihin liittyvät puutteet vaikuttivat olevan suurin este monissa käyttökohteissa.

Standardin havaittiin puutteistaan huolimatta soveltuvan hyvin sen ensisijaiseen käyttökohteeseen eli rakenteellisten dokumenttien rikastamiseen graafisella sisällöllä. Muissa käyttökohteissa kuten opetuksen tukena ja karttasovelluksissa standardin soveltuvuuden havaittiin olevan suuresti riippuvainen siitä, miten asia halutaan loppukäyttäjälle esittää. Tästä johtuen standardin ei voida yksiselitteisesti sanoa olevan soveltuva tai soveltumaton mihinkään käyttökohteeseen. Standardin huomattiin myös pääsevän parhaalla mahdollisella tavalla oikeuksiinsa, kun sitä käytetään muiden, erityisesti XML:ään pohjautuvien tekniikoiden kanssa. Tämän tutkielman ulkopuolelle jäivät SVG:n ominaisuuksiltaan karsitut versiot kuten SVG Basic ja SVG Tiny sekä Flashia lukuun ottamatta muiden vektorigrafiikkastandardien vertaaminen SVG:hen.

LÄHTEET

- Adobe Systems. 2009. Adobe SVG Viewer [viitattu 27.11.2009]. Saatavilla [www-muodossa <URL: http://www.adobe.com/svg/viewer/install/>](http://www.adobe.com/svg/viewer/install/).
- Bogaard, D.S., Vullo, R.P. & Cascioli, C. D. 2004. SVG for educational simulations. Proceedings of the 5th Conference on Information Technology Education. Salt Lake City, UT, USA, October 28-30. ACM, 43-49.
- Bos, B., Çelik, T., Hickson, I. & Lie, H.W. (toim.) 2009. W3C: CSS 2.1 Recommendation [viitattu 18.2.2010]. Saatavilla [www-muodossa <URL: http://www.w3.org/TR/CSS2/>](http://www.w3.org/TR/CSS2/).
- Bray, T., Paoli, J., Sperberg-McQueen, C.M., Maler, E. & Yergeau, F. (toim.) 2008. W3C: XML 1.0 (Fifth Edition) Recommendation [viitattu 18.2.2010]. Saatavilla [www-muodossa <URL: http://www.w3.org/TR/REC-xml/>](http://www.w3.org/TR/REC-xml/).
- Bruhn, R. & Burton, P. 2004. Displaying mathematics in a web browser using MathML and SVG. Proceedings of the 2nd Annual Conference on Mid-south College Computing, Little Rock, Arkansas. ACM, 97-106.
- Christel, M.G. & Huang, C. 2001. SVG for navigating digital news video. Proceedings of the 9th ACM International Conference on Multimedia. Ottawa, Canada, September 30 - October 05. ACM, 483-485.
- Donney, J., Murphy, S., Sacre, C., Scholz, A. & Walters, D. 2008. Globalization of graphics: delineating a research into using the scalable vector graphics file format to improve the translation of graphics. Proceedings of the 26th Annual ACM international Conference on Design of Communication. Lisbon, Portugal, September 22-24. ACM, 87-92.

- Duignan, M., Biddle, R. & Tempero, E. 2003. Evaluating scalable vector graphics for use in software visualisation. Teoksessa T. Pattison & B. Thomas (toim.) Proceedings of the Asia-Pacific Symposium on Information Visualisation - Volume 24. Adelaide, Australia. ACM, 127-136.
- Ferraiolo, O., Jun, F. & Jackson, D. (toim.) 2009. W3C: SVG 1.1 Recommendation [viitattu 1.12.2009]. Saatavilla [www-muodossa <URL:http://www.w3.org/TR/SVG11/>](http://www.w3.org/TR/SVG11/).
- Huang, H., Li, Y. & Gartner, G. 2008. An SVG-Based Method to Support Spatial Analysis in the Web Environment. International Conference on Computer Science and Software Engineering, December 12-14. IEEE Computer Society, 623-626.
- Hoschka, P. (toim.) 1998. W3C: SMIL 1.0 Recommendation [viitattu 18.2.2010]. Saatavilla [www-muodossa <URL: http://www.w3.org/TR/REC-smil/>](http://www.w3.org/TR/REC-smil/).
- Ketterl, M.; Mertens, R. & Vornberger, O. 2007. Vector Graphics for Web Lectures: Comparing Adobe Flash 9 and SVG. 9th IEEE International Symposium, December 10-12. IEEE Computer Society, 389-395.
- Mertens, R., Ketterl, M. & Vornberger, O. 2007. The virtPresenter lecture recording system: Automated production of web lecture with interactive content overviews. International Journal of Interactive Technology and Smart Education 4(1), 55-66.
- Microsoft Corporation. 1997. Microsoft typography: What is Hinting? [viitattu 28.12.2009]. Saatavilla [www-muodossa <URL: http://www.microsoft.com/typography/TrueTypeHintingWhat.mspx>](http://www.microsoft.com/typography/TrueTypeHintingWhat.mspx).
- Mohammed, S.M.A. & Fiaidhi, J.A.W. 2005. Developing secure transcoding intermediary for SVG medical images within peer-to-peer ubiquitous environment. Proceedings of the 3rd Annual Communication Networks

- and Services Research Conference, May 16-18. IEEE Computer Society, 151-156.
- Mong, J.C. & Brailsford, D.F. 2003. Using SVG as the rendering model for structured and graphically complex web material. Proceedings of the 2003 ACM Symposium on Document Engineering. Grenoble, France, November 20-22. ACM, 88-91.
- Moreno, E.D. & de Oliveira, J.I.F. 2009. Architectural impact of the SVG-based graphical components in web applications. *Computer Standards & Interfaces* 31(6), 1150-1157.
- Nykänen, O. 2007. SVG: Skaalautuva vektorigrafiikka. Jyväskylä: Docendo.
- Proberts, S., Mong, J., Evans, D. & Brailsford, D. 2001. Vector Graphics: From PostScript and Flash to SVG. ACM Symposium on Document Engineering. Atlanta, Georgia, USA, November 9-10. ACM, 135-143.
- Quint, A. 2003. Scalable vector graphics. *IEEE Multimedia* 10(3), 99-102.
- Quint, J. & Apel, U. 2005. Does learning how to read Japanese have to be so difficult: and can the web help? Special Interest Tracks and Posters of the 14th International Conference on World Wide Web. ACM, 1152-1153.
- Raggett, D., le Hors, A. & Jacobs, I. (toim.) 1999. W3C: HTML 4.01 Recommendation [viitattu 18.2.2010]. Saatavilla [www-muodossa <URL: http://www.w3.org/TR/html401/>](http://www.w3.org/TR/html401/).
- Vaughan-Nichols, S.J. 2001. Will vector graphics finally make it on the web? *Computer* 34(12), 22-24.
- Wang, F., Rabsch, C. & Liu, P. 2007. Using SVG to Model and Query Annotations and their History. IEEE International Conference on Bioinformatics and Biomedicine, November 2-4. IEEE Computer Society, 412-422.

Man, Y., Xiuhong, C., Chunling, Y. & Jianwei, N. 2009. A Practical and Light Integrated WebGIS Based On SVG. WRI International Conference, Volume 3, January 6-8. IEEE Computer Society, 142-146.

W3C. 2005. Technical Report Development Process [viitattu 12.12.2009].

Saatavilla www-muodossa <URL: <http://www.w3.org/2005/10/Process-20051014/tr>>.



SVG:n elementtejä: tekstiä, kaaria, muotoja, bittikartta, maski ja liukuväri. Alla koodiosa.

```

01<?xml version="1.0" standalone="no"?>
02<svg viewBox="0 0 400 200"
03xmlns="http://www.w3.org/2000/svg" version="1.1"
04xmlns:xlink="http://www.w3.org/1999/xlink"
05xml:space="preserve" width="100%" height="100%">
06<title>TJTA302-Liite1</title>
07<defs>
08  <linearGradient id = "liukuväri" x = "0%" y = "100%">
09      <stop stop-color = "#c2c2c2" offset = "0%" />
10      <stop stop-color = "white" offset = "100%" />
11  </linearGradient>
12<clipPath id="maski">
13  <circle cx="100" cy="100" r="80" />
14</clipPath>
15</defs>
16<rect id="tausta" x="0" y="0" width="100%" height="100%"
17      fill="#a3c1ff" stroke="black" stroke-width=".03"/>
18<path id="lintu" d="M908.723,531.412c-25.879-4.461-71.326,6.369-
-----leikattu tilan säästämiseksi-----
19  C951.67,550.934,946.459,537.92,908.723,531.412z" fill="red"
20transform="translate(-25,-90) scale(0.5,0.5)"/>
21<rect x="170" y="70" width="160" height="60"
22      fill="url(#liukuväri)" stroke="black" stroke-width="2"/>
23<text font-size = "20" fill = "black" stroke = "none">
24  <tspan x = "190" y = "95"> NULLA DIES</tspan>
25  <tspan x = "190" y = "120"> SINE LINEA</tspan>
26 </text>
27  <image x="-110" y="20" width="400" height="200"
28      xlink:href="bittikartta.JPG" clip-path="url(#maski)"/>
29<circle id="kehys" cx="100" cy="100" r="80" fill="none"
30  stroke="black" stroke-width="10" />
31</svg>

```