

Erkki Mäkinen (toim.)

**Pieniä tietojenkäsittely-
tieteellisiä tutkimuksia**

Kevät 2007



TIETOJENKÄSITTELYTIETEIDEN LAITOS
TAMPEREEN YLIOPISTO

D-2007-5

TAMPERE 2007

TAMPEREEN YLIOPISTO
TIETOJENKÄSITTELYTIETEIDEN LAITOS
JULKAISUSARJA D – VERKKOJULKAISUT
D-2007-5, KESÄKUU 2007

Erkki Mäkinen (toim.)

**Pieniä tietojenkäsittely-
tieteellisiä tutkimuksia**

Kevät 2007

TIETOJENKÄSITTELYTIETEIDEN LAITOS
33014 TAMPEREEN YLIOPISTO

ISBN 978-951-44-6974-9
ISSN 1795-4274

Sisällysluettelo

MVC-sovelluskehyksistä Java EE -ympäristössä.....	1
<i>Mike Arvola</i>	
Yrityksen tietoturvallisuuden hallintasuunnitelma.....	20
<i>Mika Ekman</i>	
Watson-Crick automaattit.....	30
<i>Sirkka Eloranta</i>	
Käytettävyys ja visuaalinen suunnittelu.....	43
<i>Tuomo Hakaoja</i>	
Yliopisto-opiskelijoiden opiskelutaidon kehittäminen verkko-oppimisympäristöjen avulla.....	57
<i>Johanna Hiden</i>	
Ei-visuaaliset vuorovaikutuskeinot ja multimodaalisuus mobiililaitteissa.....	73
<i>Eveliina Houni</i>	
IP-liikenteen luokittelu.....	85
<i>Mikko Lammi</i>	
Teknologian turhauttavuus.....	102
<i>Santtu Mansikkamaa</i>	
Kuvadatan pakkausalgoritmeista.....	109
<i>Iikka Mattila</i>	
Mobile Social Media.....	118
<i>Sari Miettinen</i>	
Koliulotteisen geometrian topologisesta esittämisestä monikulmioverkoilla..	132
<i>Matias Muhonen</i>	
Yhteisölliset verkkopalvelut.....	149
<i>Jarno Ojala</i>	
Roskapostin torjunta.....	159
<i>Harri Pirttinen</i>	
Roskapostin suodatusmenetelmät.....	173
<i>Harri Sundström</i>	
Laser, LED, neste vai vaha? – Väritulostamisen vaihtoehdot tämän päivän toimistoissa.....	187
<i>Jussi Tarvainen</i>	
Avoin lähdekoodi ja lisenssit.....	204
<i>Riikka Valtanen</i>	
Dokumentinhallintaohjelmat pienen organisaation näkökulmasta.....	222
<i>Aleksi Vuorenmaa</i>	

MVC-sovelluskehysistä Java EE -ympäristössä

Mike Arvela

Tiivistelmä.

Java EE -ympäristöön on tarjolla paitsi aiempaa enemmän myös entistä laadukkaampia kehitystyökaluja ja tekniikoita. Moni saatavilla olevista sovelluskehysistä noudattaa MVC-paradigmaa, mikä mahdollistaa käyttöliittymälogiikan erottamisen sovelluslogiikasta. Monissa tapauksissa erilaisia toteutusvälineitä tai niiden osia voidaan käyttää yhdessä halutun päämäärän saavuttamiseksi. Tässä tutkielmassa esitellään MVC-arkkitehtuurin pääpiirteet ja keskeisiä sovelluskehysistä, jotka mahdollistavat sen toteuttamisen Java EE -ympäristössä. Samalla pyritään tuomaan esille niiden vahvuuksia, heikkouksia ja erityispiirteitä ja pohditaan tulevaisuuden kehityssuuntia.

Avainsanat ja -sanonnat: J2EE, Java EE, MVC, Model-View-Controller, Mallinäköohjain, Ohjelmistoarkkitehtuuri, Suunnittelumallit, WWW

CR-luokat: D.1.5, D.2.11, D.2.13, K6.3

1. Johdanto

Yhä useammat sovellukset rakennetaan suoraan WWW-ympäristöön, jossa eräs keskeisimpiä haasteita on käyttöliittymälogiikan erottaminen sovelluslogiikasta. Ohjelmistojen koko myös kasvaa nopeasti, jolloin ylläpidettävyyden ja skaalautuvuuden merkitys korostuu. Yleinen tapa vastata näihin haasteisiin on rakentaa sovellus MVC-suunnittelumallin mukaisesti.

Käytännössä jokaiselle WWW-ympäristössä käytettävälle ohjelmointikielelle on kehitetty lukuisia erilaisia MVC-mallia noudattavia kehysratkaisuja. Keskenään kilpailevia toteutuksia saattaa kielestä riippuen olla jopa kymmeniä.

Java-kieli lukuisine oheisteknologioineen on saavuttanut suurta suosiota monissa yritystason WWW-ratkaisuissa. Skaalautuvuutensa ja saavutettavuutensa ansiosta se on kerännyt laajan käyttäjäkunnan ja siitä on tullut varteentotettava vaihtoehto myös pienten ja keskisuurten yritysratkaisujen toteuttamiseen. Vaikka Java EE on hyvin suosittu teknologia-alusta, siitä on tehty verrattain vähän tieteellistä tutkimusta, kuten toteaa myös Westkämper [2004].

Java EE -sovelluskehityksessä tilanteeseen tuo lisäväriä se, että eri sovelluskehysten osia tai komponentteja voidaan käyttää rinnakkain. Näin työvälineet ja teknologiat on mahdollista valita tarpeiden mukaan. Koska sovelluskehityksessä ei tunnetusti kuitenkaan ole yhtä ainoa oikeaa tapaa asioiden tekemiseen, on kehittäjän tunnettava käytettävissä olevien tekniikoiden vahvuudet ja heikkoudet saadakseen niistä mahdollisimman paljon hyötyä.

Tämän tutkielman tavoitteena on antaa yleiskuva keskeisistä Java EE -teknologioista ja MVC-arkkitehtuurin merkityksestä WWW-sovelluskehityksessä. Aluksi käydään läpi MVC-arkkitehtuurin perusajatukset, minkä jälkeen esitellään keskeiset Java EE -ydynteknologiat. Tämän jälkeen vertaillaan yleisesti käytettyjä sovelluskehityksiä ja tuodaan esille niiden erityispiirteitä, vahvuuksia ja heikkouksia. Lopuksi pohditaan tulevaisuuden kehityssuuntia.

2. MVC (Model-View-Controller)

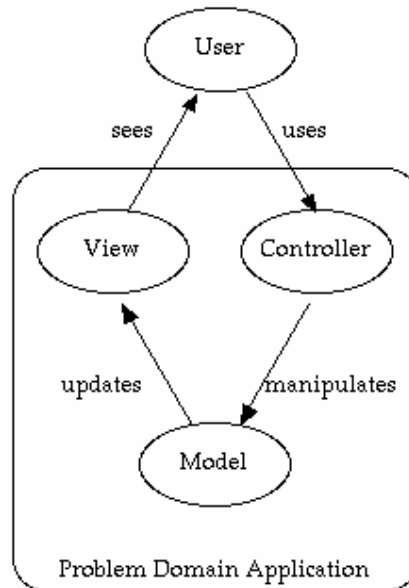
MVC (model-view-controller, malli-näkymä-ohjain) on suunnittelumalli, joka on luotu sovellusalueeseen sidotun ohjelmistologiikan erottamiseen käyttöliittymälogiikasta. Se on hyvin yleinen graafisten käyttöliittymien ohjelmoinnissa, jossa sen voidaankin katsoa saavuttaneen de facto -standardin aseman.

2.1. Historia

MVC-käsite juontaa juurensa 1970-luvulle, jolloin olioajattelu oli vielä nuori käsite ohjelmistokehityksessä. Norjalainen Trygve Reenskaug luonnosteli MVC-arkkitehtuurin perusajatuksen vuonna 1973 vieraillessaan Xerox PARC -tutkimuskeskuksessa. Hänen tavoitteenaan oli mallintaa telakan hallinnollinen tietojärjestelmä siten, että suurten ja monimutkaistenkin tietomäärien käsittely olisi mahdollista [Rupp, 2003]. Reenskaug kirjoitti ensimmäisen malliin pohjautuvan käytännön toteutuksen vuonna 1978. Hänen mukaansa mallin keskeisin tavoite oli käyttäjän mentaalimallin mahdollisimman saumaton yhdistäminen tietokoneelle luotuun digitaalisen malliin sovellusalueesta [Reenskaug]. Myös Microsoft [MSDN] esittää MVC-mallin ratkaisuna tietovaraston ja käyttöliittymän sujuvaan yhteenliittämiseen.

2.2. Rakenne

MVC-mallissa sovellus jaetaan kolmeen osa-alueeseen: *Malli* (model) sisältää sovellusaluekohtaisen tiedon ja sovelluslogiikan tämän tallentamiseen, ylläpitoon ja käsittelyyn. *Näkymä* (view) määrittää käyttöliittymän ulkoasun ja mallin tietojen esitystavan käyttöliittymässä. *Ohjain* (controller) vastaanottaa käyttäjältä tulevat käskyt ja muokkaa mallia ja näkymää niiden perusteella [Wikipedia, 2006a].



Kuva 1. MVC-mallin rakenne [Baray, 1999]

MVC-mallista on lukuisia erilaisia toteutuksia, jotka poikkeavat toisistaan yksityiskohdiltaan. Yksinkertaistettuna malli voidaan kuitenkin ajatella syklinä, jossa *ohjain* esittää järjestelmään saapuvaa syötevirtaa, *malli* sen käsittelyä ja *näkymä* järjestelmän tulostetta [WACT, 2005]. Tätä sykliä on havainnollistettu kuvassa 1. Mallin pohjalta on myös luotu laajennuksia, joiden avulla se on saatu soveltumaan erilaisiin käyttötilanteisiin (kts. kohta 2.3).

2.3. MVC-mallin vahvuuksia ja puutteita

Koskimiehen ja Mikkosen [2005] mukaan MVC-malli on luonnollinen valinta graafisen käyttöliittymän toteutusmekanismiksi, sillä se mahdollistaa mallin uudelleenkäytön lähes kaikissa tilanteissa. Hänen mukaansa lisäksi ”lähes kaikki tällaiset ympäristöt tavalla tai toisella ohjaavat käyttäjää MVC-mallin mukaiseen arkkitehtuuriin”. Malli pyritään tavallisesti toteuttamaan siten, että se on riippumaton siihen liittyvistä näkymistä ja ohjaimista. Tällöin yhteen malliin voidaan liittää useita näkymiä ja ohjaimia [Näsi, 2005].

Java Blueprints -dokumentaatiossa [Sun Blueprints, 2002b] mainitaan muutamia MVC-suunnittelumallin käyttöönotosta seuraavia etuja. Erottamalla sovelluksen osa-alueet (tiedon säilyvyys, esitys, kontrolli) toisistaan MVC-malli vähentää kahden koodin määrää ja tekee sovelluksesta helpommin muokattavan. Samalla se auttaa sovelluskehittäjiä paneutumaan omien erikoistaitojensa mukaisiin työtehtäviin ohjelmoitaessa selkeästi määriteltyjä rajapintoja. Uusien tietolähteiden käyttöönotto helpottuu, kun voidaan ohjelmoida yksinkertainen *muunnin* (adapter) tiedon siirtämiseksi tietolähteeltä näkymärajapin-

nalle. Vastaavasti näkymärajaan voidaan luoda erilaisia asiakasohjelmato-
teutuksia. Myös luokkien väliset vastualueet selkeytyvät, mikä helpottaa oh-
jelmistovirheiden löytämistä ja korjaamista.

MVC-mallin edustaman arkkitehtuurityylin ongelmana on Koskimiehen ja
Mikkosen [2005] mukaan se, että näkymä- ja ohjainluokat ovat kiinteästi toisiin-
sa liittyviä, mikä hankaloittaa niiden uudelleenkäyttöä muissa yhteyksissä. Nä-
si [2005] tuo lisäksi esille näkymien ja ohjainten riippuvuuden mallista. Myös
useiden samanaikaisten syötevirtojen hallitseminen (esim. multimodaalisissa
käyttöliittymissä) on ongelmallista, sillä malli ei ota kantaa siihen, miten eri
syötteet yhdistetään yhdeksi komennoksi. MVC sopiikin parhaiten sovelluk-
siin, joissa on kerrallaan käytössä vain yksi vaihtuva syötekanava, kuten taval-
linen WWW-sivu.

3. Java Platform Enterprise Edition 5 (Java EE)

Sun Microsystems mieltää Java-kielen jakautuvaksi kolmeen osaan käyttökoh-
teidensa mukaan. Näistä Java Platform Enterprise Edition 5 (aiemmalta nimel-
tään Java 2 Enterprise Edition, J2EE), on Java-kielen WWW-ympäristöön tarkoi-
tettu versio [Sun, 2006]. Tässä luvussa esitellään keskeiset Java EE -teknologiat
Servlet API ja JSP (JavaServer Pages). Luvussa 4 esitellään lisäksi standardoidut
kehysratkaisut EJB (Enterprise JavaBeans) sekä JSF (JavaServer Faces).

Wikipedia [2006c] mainitsee Java EE-kehityksen etuina hyvin matalat käyt-
töönottokustannukset. Tätä väitettä tukee osaltaan saatavilla olevien ilmaisten
kehitystyökalujen huomattava määrä. Esimerkiksi IBM:n alun perin kehittämä
Eclipse-kehitysympäristö on nykyään paitsi avointa lähdekoodia myös eräs ke-
hittyneimmistä alustariippumattomista kehitystyökaluista.

Java EE -sovelluksia ajetaan *sovelluspalvelimella* (application server), joita on
olemassa lukuisia erilaisia. Monien yritysten tarjoamien kaupallisten sovellus-
palvelinten lisäksi tarjolla on myös ilmaisvaihtoehtoja.

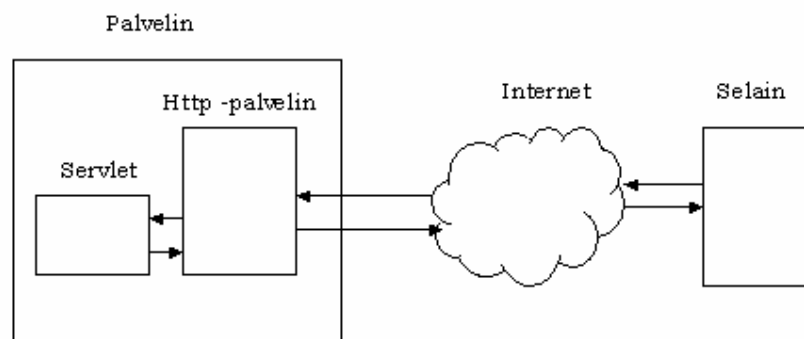
Java EE -sovelluskehityksen yhteydessä puhutaan yleisesti Model 2 -
arkkitehtuurista. Tällöin erityinen ohjain*palvelma* (servlet) välittää pyynnön jol-
lekin JSP-näkymälle, joka palauttaa valmiin näkymän selaimelle. J2EE Blue-
Prints -dokumentaatioissa suositellaan MVC-mallia avuksi Model 2 -tyyppisen
arkkitehtuurin toteuttamiseen [Sun, 2002b].

Tässä luvussa esiteltyjä tekniikoita kehitetään yhteisöllisellä Java Com-
munity Process -menetelmällä, joka tuottaa numeroituja JSR-spesifikaatiodoku-
mentteja (Java Specification Request). Kohtien sisältö pohjautuu enimmäkseen
Sunin vastaavien teknologioiden referenssidokumentaatioon ([Sun, 2006]). Täs-
sä esiteltyt tekniikat muodostavat WWW-pohjaisen Javan rungon, jonka ympä-

rille muu toiminnallisuus ja sovelluskehukset on lähes poikkeuksetta rakennettu.

3.1. Java Servlet API

Palvelma (servlet) on Java-kielinen ohjelmakomponentti (JSR 154, API versio 2.5), jolla voidaan toteuttaa WWW-ympäristöön dynaamista toiminnallisuutta [Sun, 2006]. Katajisto [2000] toteaa, että palvelmat toimivat Javaa-tukevilla WWW-palvelimilla vastaavasti kuin *sovelmat* (applet) Javaa tukevilla WWW-selaimilla. Niitä voidaan ajatella CGI-rajapinnan (Common Gateway Interface) toteutuksena Java-ympäristössä eli mekanismina, jolla toteutetaan pyyntö/vastaus -tyyppisiä sovelluksia (ks. kuva 2).



Kuva 2. Selaimen ja palvelman välinen liikenne [Katajisto, 2000]

Katajisto viittaa Callawayhin (1999), jonka mukaan palvelmien edut CGI:hin verrattuna tulevat esiin suorituskyvyssä, siirrettävyydessä ja turvallisuudessa. Toisin kuin CGI:ssä, jossa luodaan uusi prosessi jokaista asiakasohjelman pyyntöä kohti, useimmat palvelmat ajetaan samassa prosessiavaruudessa kuin palvelin ja ne ladataan kertaalleen palvelimen käynnistyessä. Katajiston mukaan tästä on etua erityisesti tietokantasovelluksissa, joissa sama tietokantayhteys voidaan jakaa useamman pyynnön kanssa.

Palvelma on käytännössä Java-luokka, joka toteuttaa Servlet API:ssa määritellyn GenericServlet- tai HttpServlet-rajapinnan. Se vastaanottaa objektin, joka sisältää WWW-selainohjelmalta tulleen sivupyynnön (HttpServletRequest) ja luo sen perusteella vastausobjektin (HttpServletResponse), joka määrittelee takaisin selaimelle lähetettävät tiedot [Sun, 2006].

3.2. JavaServer Pages (JSP)

JavaServer Pages (JSR 245, versio 2.1) on teknologia, joka yksinkertaistaa WWW-sivujen dynaamista generointia. JSP-sivu käyttää sisällön muodostamiseen Java-kielellä toteutettuja XML-tageja ("avainsanoja"), jotka kätkevät sovel-

luslogiikan sisäänsä. Kun JSP-sivu on evaluoitu kokonaisuudessaan, muodostettu sisältö palautetaan. JSP-sivut käännetään palvelmiksi ja ne saattavat myös sisältää viittauksia muihin komponentteihin, kuten EJB-*papuihin* (bean, kts. kohta 4.1).

JSP-tekniikan tavoitteena on yksinkertaistaa sivunluontiprosessia erottamalla esitystapa varsinaisesta sisällöstä. Selaimelle lähetettävä sivu on usein sekoitus staattista sivupohjaa ja dynaamisesti generoitua sisältöä, jolloin on helpompi työskennellä JSP-sivuilla kuin yrittää saada aikaan sama toiminnallisuus pelkillä palvelmilla. [Sun, 2006]

JSP-sivuilla voidaan hyödyntää esimerkiksi JSTL-tagikirjastoa (JavaServer Pages Standard Tag Library, JSR 52), johon on koottu tageja tavallisille toiminnolle kuten ehtolauseille, XML-tiedon käsittelylle, tietokantatoiminnolle, silmukoille sekä internationalisoinnille.

4. Erilaisia sovelluskehityksiä

Tämä luku esittelee sovelluskehityksiä sekä tekniikoita, joita voidaan käyttää MVC-mallia hyödyntävässä Java EE -sovelluskehityksessä. Tarkasteltavat ratkaisut on tietoisesti rajattu muutamaa keskeiseen tekniikkaan, joiden tiedetään olevan laajalti käytössä ja jotka ovat saaneet osakseen paljon näkyvyyttä niin Internetin foorumeilla kuin niistä kirjoitettujen kirjojenkin muodossa.

Java EE -kehityksen matalien lähtökustannusten puolesta puhuu se seikka, että suuri osa saatavilla olevista tekniikoista on ilmaiseksi ladattavia avoimen lähdekoodin projekteja. Pelkästään Java-ympäristöön on pikaisella haulla löydettävissä yli 40 WWW-kehystä. Vaikka näistä kaikki eivät ilmoittaisikaan noudattavansa MVC-paradigmaa, on silti selvää, että tässä tutkielmassa esitellyt kehitykset edustavat vain jäävuoren huippua.

Tässä luvussa tarkastellaan seuraavia tekniikoita: Enterprise JavaBeans, JavaServer Faces, Spring Framework ja Apache Struts.

4.1. Enterprise JavaBeans (EJB)

EJB on standardoitu sovelluskehitys komponenttipohjaisten, hajautettujen yritystason sovellusten kehittämiseen. EJB kehitettiin helpottamaan niitä tehtäviä, jotka vievät tyypillisesti eniten aikaa sovelluskehityksessä [Richardson, 2006].

EJB-määrittelyn kehitti alun perin IBM vuonna 1997, minkä jälkeen Sun otti sen jatkokehitykseen Java Community Processin alle. EJB:n versiot 2.0 ja 2.1 määritellään JSR-dokumenteissa 19 ja 153, ja versio 3.0 JSR:ssä 220 [Wikipedia, 2006d].

4.1.1. Historia

Richardson [2006] käy läpi EJB:n syntyvaiheita ja suunnitteluperiaatteita. Ennen EJB:tä kehittäjien tuli toteuttaa itse sovelluksen *läpileikkaavat osa-alueet* (Cross-cutting Concerns) kuten transaktioiden hallinta, autorisointi ja persistenssi (tiedon säilyminen ohjelman suorituskertojen välillä). Tämä oli työlästä ja luodut ratkaisut usein myös sekoittuivat muun sovelluslogiikan kanssa. EJB:n avulla nämä sovelluksen osa-alueet voitiin nyt kirjoittaa kapseloituina *paperina* eli Enterprise JavaBean -komponentteina. Tämä mahdollisti niiden hallinnan keskitetyksi ja erillään muusta sovelluksesta. EJB-kehys huolehtii niin komponenttien käyttöoikeuksista kuin myös tiedon persistoinnista tietokantaan. EJB:n versioiden 1.0 ja 2.0 on sanottu onnistuneen osittain aspektien erottamisessa sovelluslogiikasta, mutta tästä seurasi uusia ongelmia: komponentit olivat vahvasti sidottuja EJB-kehykseen ja sen tarjoamiin rajapintoihin. Lisäksi sovelluslogiikka ei ollut siirrettävissä kehyksen eri versioiden välillä.

EJB otettiin käyttöön monissa suurissa yrityksissä sen ilmestyttyä. Tästä seurasi pian ongelmia, joiden seurauksena EJB:n maine kärsi. Suurimpana yksittäisenä ongelmana pidettiin sen tarjoamien rajapintojen monimutkaisuutta. Monet sovelluskehittäjät kokivat tämän ongelmalliseksi, etenkin kun ongelmat, joita alun perin pyrittiin ratkaisemaan, olivat itsessään monimutkaisia. EJB-sovelluskehitykseen luotiin helpottavia kehitystyökaluja, mutta nämä eivät poistaneet alkuperäistä ongelmaa. Tämän katsotaan vaikuttaneen siihen, että tyytymättömät ohjelmoijat ryhtyivät kehittämään ”kevyitä” (lightweight) avoimen lähdekoodin vaihtoehtoja, kuten Hibernaten tiedon persistointiin ja Spring Frameworkin (esitelty luvussa 0). [Wikipedia, 2006d]

4.1.2. EJB 3.0

Tällä hetkellä Java-yritysmaailmassa vallitsee konsensus, jonka mukaan sovelluslogiikkaa tarjoavat komponentit kannattaa toteuttaa POJOina (*Plain Old Java Object*), joiden ei vaadita toteuttavan erityisiä rajapintoja. Käytännössä tämä tarkoittaa sitä, että komponentilla tulee olla parametrin oletusmuodostin, ja sen ominaisuudet ovat käsiteltävissä *getter*- ja *setter*-metodeilla. Uusi EJB 3.0 -määrittely pohjautuu POJO-ajattelulle keskeisen tavoitteen ollessa se, etteivät komponentit aiempien määrittelyn tavoin olisi riippuvaisia kehyksen rajapinnoista.

Monet kehittäjät näkevät yksinkertaistetun 3.0-määrittelyn tervetulleena kehityssuuntana. Richardson [2006] ja Zadrozny [n.d.] nostavat esille mm. seuraavia asioita:

- Java-annotaatioiden (ohjelmakoodin yhteyteen kirjoitettu metatieto) ja oletusasetusten käyttäminen vähentää huomattavasti tarvetta monimutkaisille XML-konfiguraatioille
- Komponenttien keskinäiset riippuvuudet yksinkertaistuvat, kun suositetaan riippuvuuksien injektointia (kts. alakohta 4.3.2) ja käytetään vähemmän koodiartefakteja
- Uudistettu persistointimalli pohjautuu POJO-ajatteluun
- Aiemmasta poiketen testaaminen on mahdollista myös sovelluskontekstin ulkopuolella.

Keith [2006] mainitsee lisäksi, että EJB3 mahdollistaa oikean luonnollisen perinnän, mikä ei aiemmin ollut mahdollista tiukoista rajoituksista ja konventioista johtuen. Myös olioiden tallentaminen relaatiotietokantaan on nyt mahdollista, kun EJB:n persisointirajapinta pohjautuu Oraclen kehuttuun TopLink-arkkitehtuuriin.

4.2. JavaServer Faces (JSF)

JavaServer Faces (JSR 252, versio 1.2) on WWW-sovelluskehys, joka pyrkii yksinkertaistamaan käyttöliittymien kehittämistä Java EE -sovelluksissa. Oletuksena JSF käyttää näyttötekniikkana JSP:tä, mutta tarkoitukseen voidaan hyödyntää myös muita teknologioita.

JSF koostuu palvelmien käyttöön tarkoitetuista Java-luokista sekä räätälöidyistä JSP-tagikirjastoista [Stelting, 2003], jotka mahdollistavat JSF-toiminnallisuuden liittämisen JSP-sivupohjiin. Se sisältää kokoelman rajapintoja moniin eri tarkoituksiin: käyttöliittymäkomponenttien esittämiseen, tapahtumakäsittelyyn, syötteiden tarkistamiseen, navigaation määrittelyyn, internalisointiin ja saavutettavuuteen. Listauksessa 2 annetaan esimerkki painikekomponentin luomisesta JSF:llä.

Sun [2006] mainitsee JSF-tekniikan vahvuuksiksi helppokäyttöisyyden, standardisoinnin ja laiteriippumattomuuden.

JSF:ssa sovelluslogiikka on erotettu esityslogiikasta, minkä sanotaan mahdollistavan työtehtävien jakamisen ja johtavan lyhyempiin kehityssykleihin (joskin huomattakoon, että sama voidaan jollain tasolla yleistää koskemaan kaikkia MVC-mallin mukaisia toteutuksia). Samaten JSF-tekniikan todetaan mahdollistavan kaikkien käyttöliittymäaspektien sujuvan hallinnan: se tarjoaa tapahtumapohjaisen mallin (Event Model) palvelinpuolelle ja mahdollistaa käyttöliittymän tilan hallitsemisen sivupyynnöiden välillä. JSF-kehys on suunniteltu siten, että se mahdollistaa integroinnin muiden tekniikoiden kanssa: esimerkiksi ulkoisia JavaBean-komponentteja voidaan konfiguroida käyttöön JSF-elementeiksi. [Stelting, 2003]

Toisaalta Stelling mainitsee JSF:n heikkouksiksi käyttöönoton edellyttämän konfiguroinnin, sivupyynnöiden prosessoinnin kompleksisuuden sekä sen kustannustehottomuuden.

JSF-sivupyynnön elinkaari on esitetty kuvassa 3.

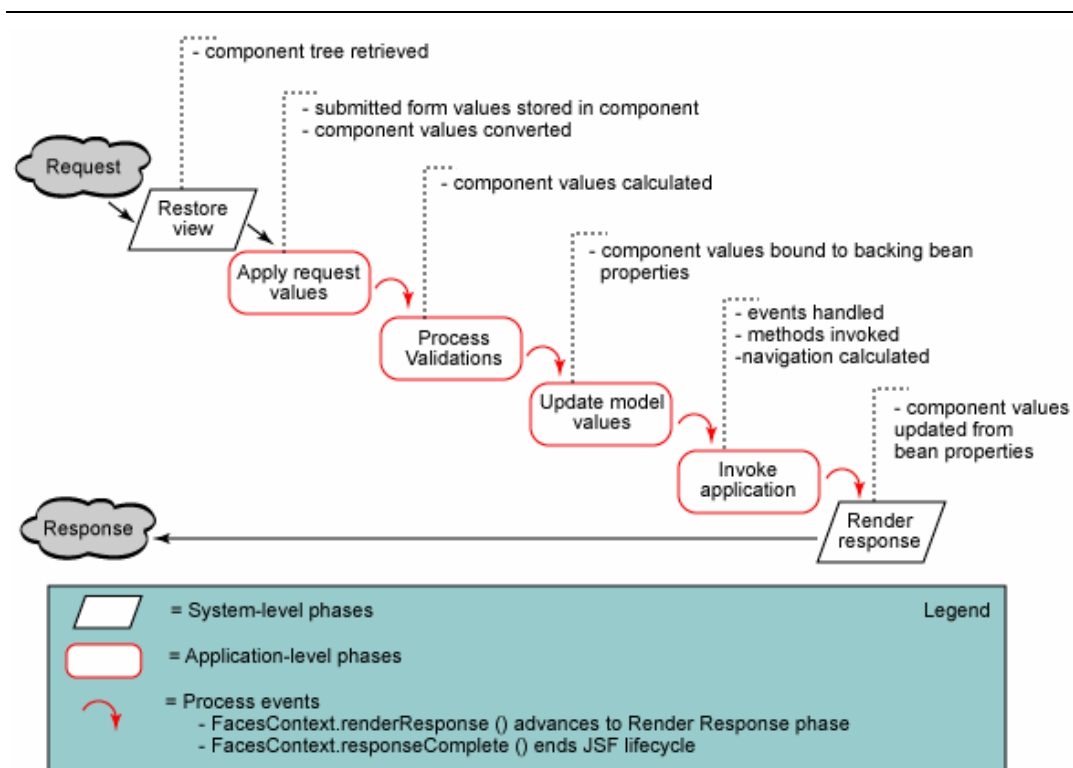
```
private static UIComponent buildButton(String displayValue,
                                       String bindingValue,
                                       String idPrefix,
                                       String id)
{
    HtmlCommandExButton button =
        app.createComponent(HtmlCommandExButton.COMPONENT_TYPE);

    button.setId(idPrefix + displayValue + "_" + id);
    button.setTitle(displayValue);
    button.setValue(displayValue);
    button.setStyleClass("commandExButton");

    MethodBinding mbButton = app.createMethodBinding(bindingValue, null);
    button.setAction(mbButton);

    return button;
}
```

Listaus 1. Esimerkki JSF-painikekomponentin luomisesta ¹



Kuva 3. JSF-sivupyynnön elinkaari ²

¹ Peter Kasson, Dynamic JSF Development, http://www.jsfcentral.com/articles/dynamic_jsf.html, viitattu 18.12.2006.

4.3. Spring Framework (Spring Web MVC)

4.3.1. Esittely

Spring Framework³ on Interface21-nimisen yhtiön kehittämä avoimen lähdekoodin sovelluskehys. Sen kehitystyön voidaan katsoa saaneen alkunsa siitä, kun Rod Johnson kokosi vuonna 2000 kirjaansa *Expert One-on-one J2EE Design and Development* ajatuksiaan siitä, kuinka J2EE-kehitystä voitaisiin helpottaa ja yhdenmukaistaa. Lukijoiden toivomuksesta kirjassa esitetty koodi julkaistiin avoimen lähdekoodin lisenssin alaisena, ja ryhmä vapaaehtoisia loi Spring-projektin tavoitteenaan kehittää edelleen Johnsonin koodia [Wikipedia, 2006b].

Kehyksen ensimmäinen versio 1.0 julkaistiin yli vuoden kestäneen kehitystyön seurauksena maaliskuussa 2004. Vuonna 2005 sen kehittäjäjoukko perusti Interface21:n mm. tarjotakseen kaupallista tukea yrityksille. Paljon odotettu Spring 2.0 valmistui marraskuussa 2006 [Spring, 2007].

Ensimmäisen versionsa julkaisun jälkeen Spring saavutti suurta suosiota kehittäjien keskuudessa. Se teki lyhyessä ajassa tunnetuksi useita monille kehittäjille ennestään tuntemattomia tekniikoita. Tästä on seurannut myös kritiikkiä ja keskustelua, sillä joissakin piireissä Springin suunnitteluperiaatteet nähtiin siirtymänä kauemmas perinteisistä ohjelmointimalleista [Wikipedia, 2006b].

Spring Framework (tarkemmin Spring Web MVC) on tämän hetken suosituimpia tekniikoita Java EE -pohjaisissa MVC-kehyksissä. Kehyksestä on lisäksi olemassa myös .NET-arkkitehtuuriin perustuva versio Spring.NET.

4.3.2. Suunnitteluperiaatteet

Springin tekijöiden kantavana ajatuksena on ollut rakentaa helppokäyttöinen kehys, joka kannustaisi olioperustaiseen ajatteluun asettamatta liian suurta painoarvoa käytetyille teknologiaratkaisuille. Springin tarjoamat rajapinnat on pyritty rakentamaan siten, että sovelluslogiikka pystyttäisiin pitämään täysin erillään niistä. Lisäksi integraatio muihin suosittuihin teknologioihin, kuten Hibernaten tarjoamaan ORM-toiminnallisuuteen, on pyritty tekemään helpoksi. [Spring, 2007]

Eräs keskeisimpiä Springissä hyödynnettyjä suunnittelumalleja on IoC, *Inversion of Control* (tarkemmin *Dependency Injection*), riippuvuuksien antaminen "ulkoa päin". Tavoitteena on minimoida komponenttien väliset riippuvuussuhteet käyttäen plugin-arkkitehtuuria. Riippuvuudet määritellään Springille XML-konfiguraatitiedostossa, jolloin luodut oliot eivät ole suorassa riippu-

² Richard Hightower, The JSF application lifecycle, <http://www-128.ibm.com/developerworks/library/j-jsf2/>, viitattu 1.5.2007.

³ Spring Framework, <http://www.springframework.org/>

vuussuhteessa keskenään. Springin komponenttitehdas BeanFactory huolehtii olioiden luonnista, palauttaen kehittäjälle viitteet käyttövalmiisiin komponentteihin. Komponentteja voidaan tällöin korvata toisilla ajonaikaisesti, ja esimerkiksi erilaisiin asennuksiin voidaan reagoida joustavasti konfiguraation tasolla. Tästä saavutetaan selvää hyötyä esimerkiksi testattaessa, jolloin sovelluksen osia voidaan vaivattomasti korvata esimerkiksi *mock*-toteutuksilla.

Ulkoiset riippuvuudet on Springissä mahdollistettu käyttämällä sovelluksen konfigurointiin *papuja* (engl. *JavaBean*, usein viitattu myös nimellä *POJO*, kts. alakohta 4.1.2). Tämä lähestymistapa mahdollistaa rajapintojen monipuolisen käytön.

```
<bean id="courseContainer" class="fi.uta.cs.course.CourseContainer">
  <property name="courses">
    <list>
      <ref bean="course1" />
      <ref bean="course2" />
    </list>
  </property>
</bean>

<bean id="LecturerLL" class="fi.uta.cs.course.Lecturer">
  <property name="name">Lasse Luennoitsija</property>
  <property name="email"><value>lasse@cs.uta.fi</value></property>
</bean>

<bean id="course1" class="fi.uta.cs.course.BasicCourse">
  <property name="name" value="Olio-ohjelmoinnin perusteet" />
  <property name="lecturer"><ref bean="LecturerLL" /></property>
</bean>

<bean id="course2" class="fi.uta.cs.course.BasicCourse">
  <property name="name" value="Tutkimuskurssi" />
  <property name="lecturer"><ref bean="LecturerLL" /></property>
</bean>
```

Esimerkki papujen XML-konfiguroinnista Springissä

4.3.3. Rakenne ja ominaisuudet

Springin kehitystyö on koko sen olemassaolon ajan ollut hyvin aktiivista ja nykyään se pitääkin sisällään hyvin kattavan listan ominaisuuksia ja komponentteja eri käyttötarkoituksiin. Kehys on rakennettu modulaarisesti seitsemään osaan siten, että monia sen osia voidaan käyttää myös toisistaan irrallaan ja myös muissa ympäristöissä kuin WWW:ssä. Tämän tutkielman aihepiirin kannalta olennaisimpia ovat Springin Web- ja MVC-moduulit. Usein kehiksen tähän osa-alueeseen viitataan nimellä *Spring Web MVC*.

Springin moduulit vastuualueineen on lueteltu taulukossa 1. Yksittäisen sivupyynnön kulku Springin rakennehierarkiassa on kuvattu liitteessä 1.

Moduuli	Vastuualue
Core	IoC, ulkoisten riippuvuuden hallinta (BeanFactory)
Context	Sovelluskonteksti, internationalisointi, tapahtumanhallinta, resurssien lataaminen
DAO	JDBC-abstraktiokerros, transaktioiden hallinta kaikille POJOille
ORM	Rajapinnat suosituille ORM-toteutuksille kuten JPA, JDO, Hibernate ja iBatis
AOP	Läpileikkaavien aspektien hallinta (<i>interceptors, pointcuts</i>)
Web	WWW-ympäristöön liittyvä toiminnallisuus, integraatio esim. Strutsin kanssa
MVC	MVC-mallin toteutus, joka erottaa näkymistä sekä sovel-luslogiikan että WWW-lomakkeet

Taulukko 1. Springin moduulit vastuualueineen [Spring, 2007]

Merkittävimpinä Web MVC:n ominaisuuksista voidaan mainita seuraavat:

- Laajennettavuus, mukautuvuus: kunkin osa-alueen hoitaa tarkoitukseen varta vasten suunniteltu olio, joka voidaan tarvittaessa korvata erilaisella toteutuksella
- Olioiden helppo sitominen (binding) WWW-lomakkeisiin ja niiden oikeellisuuden tarkistaminen tarkoitukseen luoduilla *validaattoreilla* (ei edellytä Spring-spesifisyyttä)
- Näkymien joustava räätälöinti ja integraatio muiden näyttötekniologioiden kanssa
- Mukautettava ja kattava tuki lokalisoinnille (mm. erilaisia *sieppareita* (interceptor) *ympäristömäärittelyksen* (locale) vaihtamiseen automaattisesti) ja teemoille
- Tagikirjastot helpottamaan JSP-sivupohjien ja lomakkeiden luontia.

4.4. Apache Struts

4.4.1. Esittely

Apache Struts⁴ on Java Servlet API -määrittelyyn pohjautuva avoimen lähdekoodin kehys, jonka loi alun perin Craig McClanahan. Se lahjoitettiin Apache Foundationin kehitettäväksi toukokuussa 2000, jolloin se liitettiin osaksi avoi-

⁴ Apache Struts, <http://struts.apache.org/>

men lähdekoodin Java-työkaluja kehittävää Jakarta Projectia. Vuonna 2005 siitä tuli itsenäinen Apache-projekti. [Wikipedia, 2007a]

Struts on eräs viime vuosien suosituimmista, ellei kaikkein suosituin, WWW-sovelluskehys. Strutsista on olemassa kaksi merkittävää versiota, Struts 1 ja alkuvuodesta 2007 julkaistu Struts 2. Strutsin uusi versio on projektille merkittävä virstanpylväs, sillä kahden kehittäjäryhmän ryhdyttyä tekemään yhteistyötä se on seuraaja Strutsin lisäksi myös WebWork-kehykselle.

4.4.2. Suunnitteluperiaatteet

Strutsin alkuperäisenä ydinajatuksena on yksinkertaisesti MVC-arkkitehtuurin toteuttaminen näyttö- ja sovelluslogiikan erottamiseksi toisistaan. Strutsin uusi 2.0-versio pyrkii tekemään kehyksestä aiempaa modulaarisemman ja helpommin konfiguroitavan. Se perustuu plugin-arkkitehturiin, joka mahdollistaa kehyksen integroinnin muihin teknologioihin. Esimerkiksi Springin IoC-containeria käytetään papujen konfigurointiin, minkä seurauksena Struts on nyt Spring-yhteensopiva.

Strutsin uudella versiolla on paljon yhteistä EJB3-määrittelyn peruseriaatteiden kanssa. Sen ydinosat perustuvat luokkien sijasta rajapintoihin, mikä vähentää komponenttien keskinäisiä riippuvuuksia. Lisäksi Struts 2 helpottaa sovelluksen konfiguroimista hyödyntämällä asetuksissa hyviksi koettuja oletusarvoja. Myös POJO-ajattelu on rantautunut kehykseen, mikä näkyy esimerkiksi siinä, ettei kehyksen toimintojen (Action) enää edellytetä toteuttavan tiettyjä rajapintoja.

4.4.3. Rakenne ja ominaisuudet

Struts 2 korvaa aiemmin käytetyn ActionServlet-nimisen ohjainosan (controller) FilterDispatcher-komponentilla, joka noudattaa Java Servlet API 2.3:ssa määriteltyä suodatinmekanismia (rajapinta javax.servlet.Filter). Suodattimia voidaan liittää yhteen ketjuksi, jolloin kukin suodatin pystyy käsittelemään pyynnön haluamallaan tavalla, välittäen sen sitten ketjussa seuraavalle suodattimelle. Suodatinmekanismi on hyvin monikäyttöinen, sillä sitä voidaan hyödyntää esimerkiksi autentikointiin, lokittamiseen, tiedonpakkaukseen ja -salaukseen, välimuistiin tallentamiseen, kuvien konvertointiin, resurssien hallintaan ja MIME-tyyppien ketjuttamiseen.

Eri toiminnoista seuraavat näkymät voidaan määritellä Strutsissa tulos-tyypeinä (Result Type), mikä mahdollistaa erilaisten teknologioiden käytön riippuen kulloinkin suoritettavan toiminnon lopputuloksesta. Kehyksestä löytyy valmis tuki näyttöteknologioille kuten JSP, FreeMarker, Velocity, PDF ja JasperReports. Koska kukin tulosolio sisältää myös tiedon toiminnon lopputuloksesta (*success, none, error, input, login*), on yksikkötestien kirjoittaminen erilai-

sille toiminnoille kätevää. Listauksessa 2 on esimerkki useiden näkymien konfiguroinnista samalle toiminnolle siten, että käytettävä näkymä määräytyy toiminnon lopputuloksen perusteella.

```
<action name="Hello">
  <result>/hello/Result.jsp</result>
  <result name="error">/hello/Error.jsp</result>
  <result name="input">/hello/Input.jsp</result>
</action>
```

Listaus 2. Yksi toiminto, monta näkymää [Struts, 2007]

Käyttöönoton helppoutteen on kiinnitetty erityistä huomiota. Uuden version myötä Strutsiin voidaan asentaa plugineja JAR-paketteina ilman erillistä konfiguraatiota. Monet muutokset astuvat voimaan ilman containerin uudelleenkäynnistystä. Ajan trendin mukaisesti Struts tarjoaa myös sisäänrakennetun AJAX-tuen. [Struts, 2007]

Yksittäisen sivupyynnön elinkaari Strutsin arkkitehtuurissa on kuvattu liitteessä 2.

5. Yhteenveto

Modernien avoimen lähdekoodin kehysten kiistaton vahvuus on se, että niiden eri osia voidaan käyttötarkoituksen mukaan yhdistellä halutun päämäärän saavuttamiseksi. Sen sijaan, että ne pyrkisivät itsessään tarjoamaan kaikenkattavia ratkaisuja spesifeihin tarpeisiin ne tarjoavat hyvän perusarkkitehtuurin, josta voidaan korvata ja johon voidaan lisätä toiminnallisuutta muista kehyksistä tai teknologioista. Tämä havainnollistuu hyvin esimerkiksi Strutsin ja Springin kohdalla, joiden dokumentaatiosta löytyy jo lähtökohtaisesti ohjeita esim. erilaisten näyttötekniologioiden käyttöön. Lisäksi esimerkiksi Strutsin uusi versio on jo valmiiksi yhteensopiva Spring-papujen kanssa.

Monissa tapauksissa lähdekoodin avoimuus ja joustava lisensointi ovat tarjonneet aktiiviselle kehittäjäyhteisölle mahdollisuuden kehittää uudelleenkäytettäviä ja modulaarisia komponentteja, joita voidaan hyödyntää useissa erilaisissa projekteissa. Tämä on helpottanut kehysarkkitehtuurien kehitystyötä, kun se on vähenemässä määrin ollut riippuvainen laadukkaiden teknologiaratkaisujen saatavuudesta ja on voitu paremmin keskittyä suunnittelutyöhön. Monista käsitteistä ja tekniikoista tuntuukin muodostuneen eräänlaisia standardeja, joita suositaan monessa eri kehyksessä.

Eräs Java-kehysten suurimmista haasteista on perinteisesti ollut helppouden ja yksinkertaisuuden säilyttäminen kehyksen koon ja oheisteknologioiden määrän kasvaessa. On havaittavissa, että myös Springin ja Strutsin tapauksessa kehys on laajentunut niin suureksi, että kokonaisuuden hahmottaminen voi olla hankalaa. Tästä voi seurata ongelmia erityisesti silloin, kun kehys ei tarjoa

valmista toteutusta tiettyyn ongelmaan ja ratkaisu edellyttää syvällisempää perehtymistä sen mekanismeihin ja periaatteisiin.

EJB 3.0:lle tuntuu asetetun melko paljon odotuksia, eikä välttämättä syyttä. Se pyrkii ennakkoluulottomasti korjaamaan aiemmissa versioissaan tehtyjä virheitä ja lainaa hyväksi koettuja ideoita OS-yhteisön puolelta. Toisaalta uusien yrittäjien, etunenässä Spring Framework, käyttäjäkunta ja näkyvyys tuntuvat kasvaneen nopeata tahtia, eivätkä monet aiemmin EJB-leiristä näihin ”kevyisiin vaihtoehtoihin” siirtyneet kehittäjät välttämättä enää koe tarvetta siirtyä takaisin. Myös JSF tuntuu jääneen verrattain vähälle huomiolle.

Springin lisääntyvästä suosiosta kertoo Internet-näkyvyyden lisäksi myös se, että sille on julkaistu enenevässä määrin erilaisia integraatiokirjastoja kolmansien osapuolten sovelluksiin. Lisäksi Interface21 järjestää säännöllisesti Spring-aiheista koulutusta ympäri maailmaa [Spring, 2007].

Strutsin tuore 2.0-versio on vielä verrattain uusi, minkä takia on vaikea arvioida sen saavuttamaa suosiota kehittäjien keskuudessa. Sillä tuntuu kuitenkin nykyisellään olevan paljon yhteistä Springin kanssa. Strutsin kehittäjäkuntaa on kuvattu hyvin yhteisölliseksi, ja tätä väitettä tukevat myös erot näiden kahden kehyksen dokumentaatiossa: siinä, missä Springin dokumentaatio on huolellisesti kasattua ja kattavaa, Struts luottaa wiki-pohjaisessa dokumentaatiossaan yhteisön voimaan.

Eräät analyttikot ovat ennustaneet Java EE -ratkaisujen joutuvan ennen pitkää monimutkaisuutensa vuoksi väistymään kevyempien vaihtoehtojen tieltä pyrittäessä kohti palvelulähtöistä arkkitehtuuria (SOA, Service Oriented Architecture) [Seeley, 2006]. Eräs potentiaalisista lähitulevaisuuden tekniikoista saattaisikin olla esimerkiksi Grails, joka pyrkii yhdistämään Java-tekniikan voiman ja skaalautuvuuden Ruby On Railsin vaivattomuuteen.

Viiteluettelo

[Baray, 1999] Cristobal Baray, The Model-View-Controller (MVC) Design Pattern, <http://cristobal.baray.com/indiana/projects/mvc.html>. Checked 8.12.2006.

[Johnson, 2005] Rod Johnson, J2EE development frameworks. *Computer* **38**, Issue 1, (Jan 2005), 107-110.

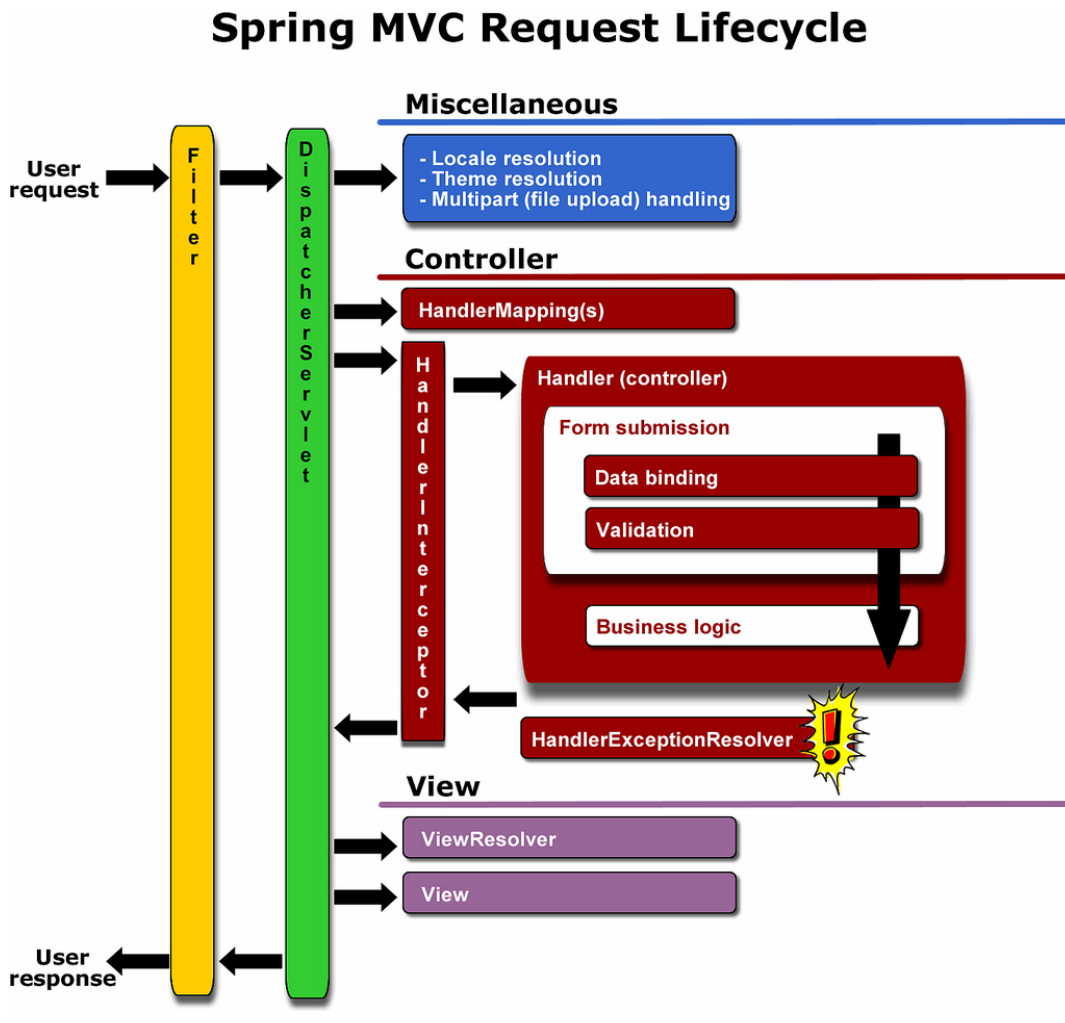
[Katajisto, 2000] Teemu Katajisto, Java-Servletit. LuK-tutkielma, Jyväskylän yliopisto, 14.12.2000. <http://www.mit.jyu.fi/opiskelu/opinnayte/LuK/Java-servletit/>, viitattu 17.12.2006.

[Keith 2006] Mike Keith, Preparing for EJB 3.0. http://www.oracle.com/technology/tech/java/newsletter/articles/toplink/preparing_for_ejb_3.html. Checked 21.4.2007.

- [Koskimies ja Mikkonen, 2005] Kai Koskimies ja Tommi Mikkonen, Ohjelmistoarkkitehtuurit. Talentum, 2005.
- [MSDN] Microsoft Developer Network, Model-View-Controller, <http://msdn2.microsoft.com/en-us/library/ms978748.aspx>. Checked 8.12.2006.
- [Näsi, 2005] Juuso Näsi, Sovelluslogiikan eriyttäminen käyttöliittymästä. *Käyttöliittymien ohjelmistoarkkitehtuurit -seminaari*, Tampereen yliopisto, 2005. <http://www.cs.uta.fi/~rr/kohar/>, viitattu 8.12.2006.
- [Reenskaug] Trygve M. H. Reenskaug, History of MVC, <http://heim.ifi.uio.no/~trygver/themes/mvc/mvc-index.html>. Checked 8.12.2006.
- [Richardson, 2006] Chris Richardson, Untangling enterprise Java. ACM Press, June 2006.
- [Rupp, 2003] N. Alex Rupp, Beyond MVC: A New Look at the Servlet Infrastructure, <http://today.java.net/pub/a/today/2003/12/11/mvc.html?page=1>. Checked 8.12.2006.
- [Seeley, 2006] Rich Seeley, Analysts see Java EE dying in an SOA world. http://searchwebservices.techtarget.com/originalContent/0,289142,sid26_gci1198211,00.html. Checked 17.12.2006.
- [Spring, 2007] Interface21, The Spring Framework, <http://www.springframework.org/>. Checked 1.5.2007.
- [Stelting, 2003] Stephen A. Stelting, About JavaServer Faces (a.k.a. Making Faces). Colorado Software Summit 2003 Paper, <http://www.softwaresummit.com/2003/speakers/SteltingJavaServerFaces.pdf>. Checked 18.12.2006.
- [Struts, 2007] Apache Software Foundation, Struts, <http://struts.apache.org/>. Checked 19.2.2007.
- [Sun Blueprints, 2002a] Sun Microsystems, Inc., Design Patterns: Model-View-Controller, <http://java.sun.com/blueprints/patterns/MVC-detailed.html>. Checked 13.11.2006.
- [Sun Blueprints, 2002b] Sun Microsystems, Inc., Designing Enterprise Applications with the J2EE Platform, Second Edition, http://java.sun.com/blueprints/guidelines/designing_enterprise_applications_2elweb-tier/web-tier5.html. Checked 11.12.2006.
- [Sun, 2006] Sun Microsystems, Inc., Java Enterprise Edition, <http://java.sun.com/javaeel/index.jsp>. Checked 11.12.2006.
- [Sun, 2007] Sun Microsystems, Inc., The Essentials Of Filters, <http://java.sun.com/products/serolet/Filters.html>. Checked 25.4.2007.
- [WACT, 2005] Web Application Component Toolkit, Model View Controller, http://www.phpwact.org/pattern/model_view_controller. Checked 13.11.2006.
- [Westkämper, 2004] Timo Westkämper, Architectural models of J2EE Web tier frameworks. University of Tampere, Dept. of Computer Sciences, Master's

- Thesis, May 2004, http://www.cs.uta.fi/research/theses/masters/Westkamper_Timo.pdf. Checked 25.4.2007.
- [Wikipedia, 2006a] Wikipedia, Model-view-controller, <http://en.wikipedia.org/wiki/Model-view-controller>. Checked 13.11.2006.
- [Wikipedia, 2006b] Wikipedia, Spring Framework (Java), http://en.wikipedia.org/wiki/Spring_Framework_%28Java%29. Checked 12.12.2006.
- [Wikipedia, 2006c] Wikipedia, Java Platform, Enterprise Edition, http://en.wikipedia.org/wiki/Java_Platform%2C_Enterprise_Edition. Checked 18.12.2006.
- [Wikipedia, 2006d] Wikipedia, Enterprise JavaBean, http://en.wikipedia.org/wiki/Enterprise_JavaBean. Checked 18.12.2006.
- [Wikipedia, 2007a] Wikipedia, Apache Struts, http://en.wikipedia.org/wiki/Apache_Struts. Checked 19.2.2007.
- [Zadrozny, n.d.] Peter Zadrozny, Commentary on the State of J2EE, <http://www.oracle.com/technology/pub/articles/masterj2ee/commentary.html>. Checked 21.4.2007.

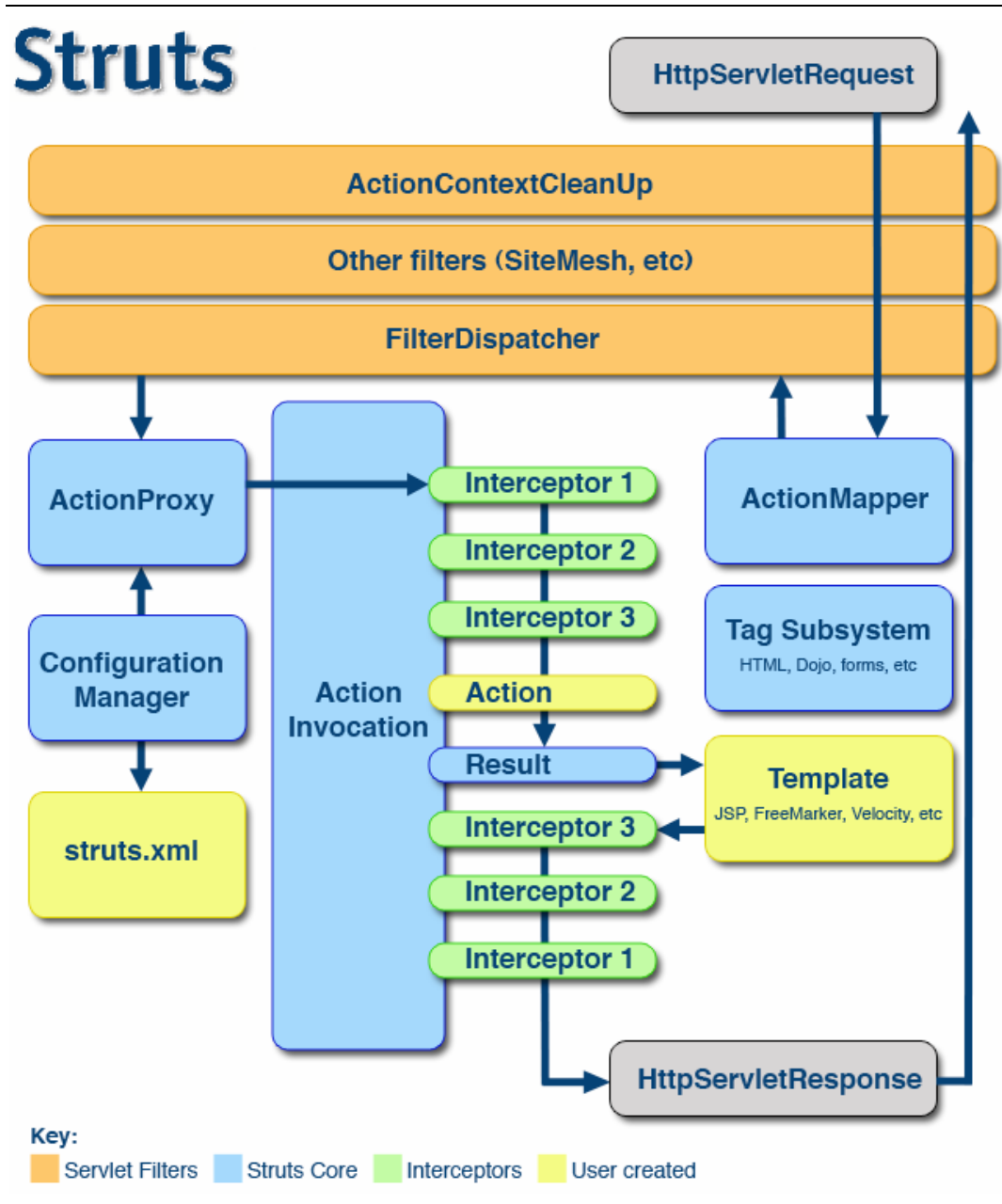
Spring MVC – Sivupyynnön elinkaari



Spring MVC Request Lifecycle ⁵

⁵ Jon Chase, Spring MVC Request Lifecycle, http://www.flickr.com/photo_zoom.gne?id=89101625&size=l, viitattu 1.5.2007.

Struts 2.0 – Sivupyynnön elinkaari



Struts 2 Architecture [Struts, 2007]

Yrityksen tietoturvallisuuden hallintasuunnitelma

Mika Ekman

Tiivistelmä.

Tutkielmassa tarkastelen yritysten nykyisiä tietoturvallisuus tilannetta sekä tutkin tietoturvallisuuden parantamiseen kehitettyjen standardien merkitystä tietoturvallisuuden parantamiseksi, sekä niiden käyttämistä yrityksen suunnittelun apuvälineenä tietoturvallisuuden hallintasuunnitelman laatimisessa. Lisäksi tutustun myös suomen elinkeinoelämän keskusliiton laatimaan ohjeistukseen yritysturvallisuuden kehittämiseksi. Tämän kautta tulee myös tietoa siitä miten EK:n jäsenyrityksissä toimitaan yritysturvallisuuden parantamiseksi.

Avainsanat ja -sanonnat: Tietoturvallisuus, BS 7799, ISO/IEC 17799

CR-luokat: K6.5

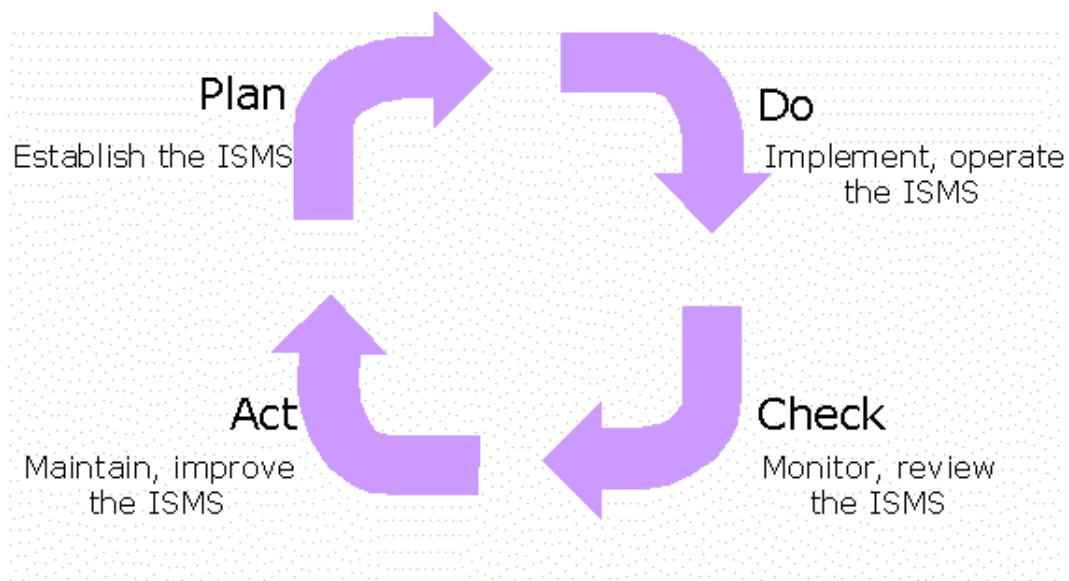
1. Johdanto

Yrityksien tietoturvallisuus on tullut viime vuosina entistä tärkeämmäksi osaksi yritysten toimintaa. Tähän on vaikuttanut erityisesti tietoliikenteen mahdollistuminen sekä yritysten sähköiset palvelut ja työkalut, joita yrityksissä käytetään. Yksi yritysten tärkeimmistä pääomista on tieto. Ohjelmisto-alan yrityksillä yrityksen tärkein pääoma voi olla kiinni pelkästään tehtyjen ohjelmistojen dokumentoinneissa sekä lähdekoodeissa. Tärkeän tiedon tuhoutuminen tai joutuminen kilpailijoille saattaa vaikuttaa yrityksen kilpailukykyyn, taloudelliseen kannattavuuteen tai joissakin tapauksissa pelkkä imagollinen tappio on yritykselle iso takaisku. Tietoa voi esiintyä monessa eri muodossa, osa tiedosta voi olla tallennettuna tietokoneelle tai ulkoiselle tai ulkoiselle palvelimelle tai se voi olla kirjoitettuna tai tulostettuna paperille. Riippumatta siitä missä muodossa tieto on tallennettuna, tulee yrityksen kannalta tärkeän tiedon turvallisuuteen kiinnittää erityistä huomiota. Tiedon turvaaminen on nykyään hyvin haastava tehtävä ja ilman tarkkaa suunnittelua ja suunnitelman toteuttamista yritys ei voi uskotella, että tärkeä tieto olisi turvassa.

Tässä tutkielmassa käyn läpi nykyistä tilannetta yritysten tietoturvallisuudessa sekä tutkin asiaa tietoturvallisuuden tason määrittävien standardien avulla. Erityisesti tutkimuksen kohteena on isobritannialaisen standardi-instituutin kehittämä BS 7799 -standardi sekä kansainvälisen ISO-standardointiorganisaation (International Organization for Standardization) siitä kehittämän ISO/IEC 17799 -standardi. Näiden standardien tarkoituksena on opastaa yrityksiä parantamaan tietoturvaa, sekä varautumaan mahdollisiin tietoturvauhkiin.

2. Tietoturvallisuuden hallintasuunnitelma

Tietoturvallisuuden hallintasuunnitelmalla tarkoitetaan suunnitelmaa, jossa listataan yrityksen tietoturvaa uhkaavat tekijät sekä niiden merkitykset yrityksen liiketoiminnalle. Tämän lisäksi suunnitelmassa määritellään suunnitelma, jolla uhkia pystytään estämään. Lähtökohtana tietoturvallisuuden hallintasuunnitelman (Information Security Management System, ISMS) laatimiselle tulee yrityksen ylimmän johdon laatia raamit sille, miten tärkeää eri tietojen turvallisuuden takaaminen yritykselle on. Tässä vaiheessa täytyy ottaa myös huomioon tietoturvan yleisten tavoitteiden ja vaatimusten lisäksi lainsäädännölliset sekä mahdollisten asiakassopimusten asettamat vaatimukset. Suunnitelman tulisi sisältää kaikki tietoturvan kannalta olennaiset asiat, mutta se ei saa olla liian yksityiskohtainen, jotta se on helposti omaksuttava. Tarkemmat yksityiskohdat eri asioiden vaatimien toimien toimenpiteiden listaamiseksi voidaan laatia erillisiin dokumentteihin, jotta suunnitelman toimintaa seuraavan henkilön on mahdollista havaita mahdolliset puutteet helposti. Suunnitelman noudattaminen on tärkeää ja suunnitelman noudattaminen vaatii myös sitä, että suunnitelman noudattamista seurataan säännöllisesti. Jokaisessa suuressa yrityksessä olisi syytä olla myös erityinen tietoturva-asiantuntija, jota tarpeen vaatiessa voidaan käyttää. Pienemmissä yrityksissä tällaisen palvelun ostaminen asiaan erikoistuneelta yritykseltä lienee kannattavin ratkaisu ellei tietoturvallisuus ole yrityksen muutoin keskeinen ala.



Kuva 1. Eteneminen kohti hallintasuunnitelmaa.

Suunnitelman laatiminen lähtee tarkasta suunnittelusta. Tässä vaiheessa lasketaan myös, se miten kannattavaa suunnitelman laatiminen ja toteuttami-

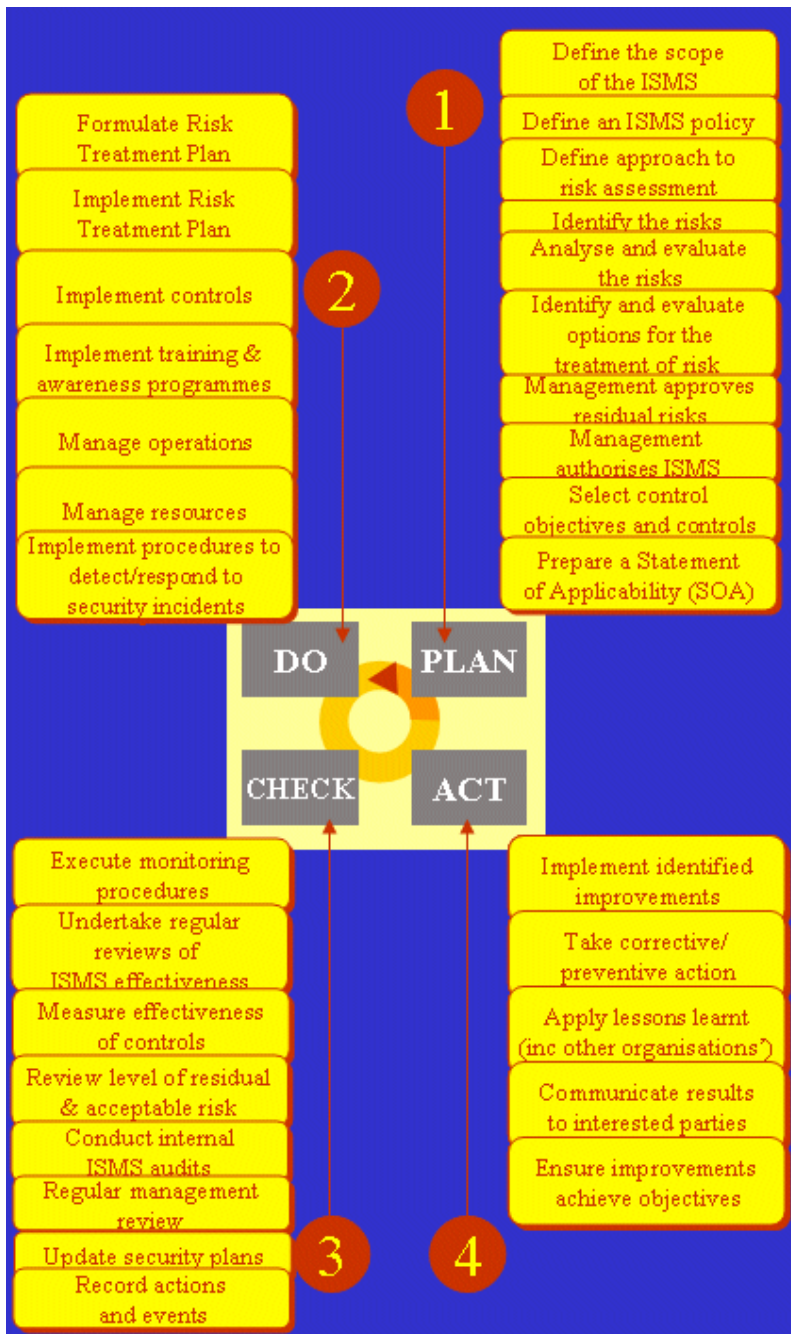
nen yritykselle on. Toimenpiteitä valittaessa tulisi kuitenkin pitää mielessä, ettei sellaisia toimenpiteitä kannata toteuttaa joiden toteuttaminen tulee kalliimmaksi kuin uhan toteutuessa siitä aiheutuneen vahingon vaikutus on. [Johansson, 2003]

Suunnitelman toisessa vaiheessa laadittu suunnitelma toteutetaan. ISO17799-standardissa on kaksitoista eri vaihetta, jotka on koottu alkuperäisen BS 7799 -standardin 127 eri vaiheesta osiin, jotta eri osa-alueiden jakaminen henkilöille helpottuisi. Näitä vaiheita seuraamalla tietoturvallisuus suunnitelman toteuttaminen voidaan helposti tarkastaa kohta kerrallaan. [Gamma, 2006]

Kolmannessa vaiheessa, joka yleensä on vaiheista pitkäkestoisin, toteutetaan suunnitelman valvonta. Suunnitelman valvonta on tärkeää varsinkin suunnitelman alkuvaiheessa siltä varalta, että laadittu suunnitelma ei ole täysin kattava vastaamaan kaikkia yritystä uhkaavia tekijöitä. Juuri tämän asian takia, koko toimintamalli on tehty siten, että vaiheita toistetaan jatkuvasti sykleissä. Tilanteet yrityksen sisällä eivät ikinä pysy vakiona. Varsinkin kun yrityksen toiminnassa tapahtuu, muutoksia tulee tehdä ainakin niiden alueiden osalta, jotka ovat muuttuneet. Näihin muutoksiin tulee laskea myös henkilöstömuutokset sekä yrityksen ulkoiset henkilöt, jotka toimivat yrityksen piirissä esimerkiksi laitteistojen ja ohjelmistojen tukihenkilöt.

Valvonnan yhteydessä havaitut puutokset tai lisäykset suunnitelmaan käydään läpi seuraavassa vaiheessa, jotta tarvittavat muutokset tai lisäykset saadaan lisättyä uuteen versioon tietoturvallisuuden hallintasuunnitelmaan. Alkuperäiseen suunnitelmaan tehdyt muutokset ja lisäykset on hyvä pitää selkeästi esillä suunnitelmassa esimerkiksi yksityiskohtaisella listalla, jossa on selvästi eritelty kaikki muutokset. Tätä kautta voidaan helpommin arvioida suunnitelman vaikutukset, sekä tätä kautta tarkastella suunnitelman aiheuttamat kustannukset ja tuottamat hyödyt.

Yksityiskohtia suunnitelman laadinnasta on esitetty kuvassa 2.



Kuva 2. Yksityiskohtia hallintasuunnitelman laatimisesta.

3. Tiedon tallennus

Tässä luvussa käsitellään esimerkkinä yhtä tärkeää tietoturvallisuuden osa-alueita, tiedon tallennusta, siihen liittyviä riskejä ja keinoja niiltä suojautumiseksi. Suurin osa yrityksen käytettävissä olevasta tiedosta on tallennettuna johonkin digitaaliseen muotoon. Suurin osa digitaalisessa muodossa olevasta tiedosta on tietokoneilla, palvelimilla tai muilla vastaavilla laitteilla kuten flash-muisteilla. Digitaalisen tiedon tuhoaminen on suhteellisen uusi asia, vaikka siihen erikoistuneita ohjelmistoja ja yrityksiä on ollut markkinoilla jo useita

vuosia. Näitä palveluja tosin ei vielä osata käyttää tarvittavan paljon hyväksi. Osasyys tähän on tietämättömyydessä siitä, että tietoa pystytään helposti ottamaan talteen esimerkiksi kovalevyiltä senkin jälkeen kun se on otettu pois käytöstä, jos tietoa ei ole oikealla tavalla tuhottu. Yleensä paperille tallennetusta tiedosta pidetään paremmin huolta siinä vaiheessa kun sitä tuhoetaan. Paperinsilppurit ovat niin vanha ja yleisesti tiedossa oleva keksintö, että se löytyy lähes jokaisesta yrityksestä. Tiedonsalaukseen erikoistunut Pointsec Mobile Technologies -yritys on tehnyt kattavia tutkimuksia Britanniassa siitä, millaisia ongelmia varsinkin tietokoneiden ja laitteiden käytöstä poistaminen aiheuttaa tietoturvallisuuden kannalta. Vuonna 2007 Pointsecin julkaiseman tutkimuksen mukaan vain alle puolet yrityksistä käyttää ammattilaisten palveluita tuhotessaan käytöstä poistettujen tietokoneiden tiedostoja. Vain 17 % tuhoaa koneeseen tallennetut tiedostot itse, vaikka se on turvallisoin tapa. Tiedon tuhoamiseen suunniteltujen ohjelmistojen vähäiseen käyttöönottamisen syynä ei voi olla osaamattomuus, sillä suurin osa ohjelmistoista on tehty niin helppokäyttöisiksi, että ihmiselle, joka työskentelee muutenkin tietokoneella, pystyy parissa minuutissa opastamaan, miten tiedon tuhoamisen varmistavaa ohjelmistoa käytetään. Pointsecin mukaan suurin syy siihen, ettei tietoa tuhota oikein on pelkästään ajan ja resurssien puute, joten hyvällä syyllä voidaan nähdä asia huonon johtamisen tuloksena. Mikäli yritykselle on luotu tietoturvallisuuden hallintasuunnitelma, on hyvin todennäköistä, että näin ilmeinen tietoturvavauhka on kirjattu siihen.

3.1. Mukana kuljetettavat tallennusvälineet

Suuri yleistyvä tietoturvavauhka, joka on ilmestynyt aivan viimeaikoina, on ihmisten mukana kuljetettavat erilaiset tallennuslaitteet. Tällaisiin laitteisiin on tallennettuna ihmisten henkilökohtaisia tietoja sekä myös yritysten tärkeitä tietoja usein täysin salaamattomassa muodossa. Tällaisia laitteita ovat esimerkiksi USB-muistit, mediasoittimet, kannettavat tietokoneet sekä niin sanotut älypuhelimet. Lontoossa pidetyn kansainvälisen InfoSecurity Europe 2006 -tilaisuuden yhteydessä järjestetyssä kyselyssä it-alan ammattilaisille selvisi, että viimeisen vuoden aikana pelkästään irrotettavien muistilaitteiden käyttö on kaksinkertaistunut. [Pointsec, 2007] Tämän lisäksi yhä useammin ihmisillä on käytössään kannettavia tietokoneita ja älypuhelimia, joiden määrä on myös lisääntynyt, vaikkei yhtä huimaa tahtia. Kaksi kolmasosaa yrityksissä kyselyyn vastanneiden henkilöiden yrityksissä ei ole minkäänlaista tietoturvaohjetta mukana kuljetettavien muistilaitteiden käyttöön. Huolestuttavaksi asian tekee se, että yli puolet vastaajista tallentaa laitteille sopimusten, tarjousten ja muiden liiketoimintaan liittyvien asiakirjojen kaltaista yritystietoa. Joissakin tapauksissa laitteille tallennetaan asiakkaiden yhteystietoja sekä pahimmassa tapauksessa

muutama kyselyyn vastanneista on tallentanut laitteille jopa tilinumeroita ja salasanoja.

3.2. Älykkäiden mobiililaitteiden yleistymisen ongelmat

Älypuhelinien ja kannettavien tietokoneiden joutuminen väärin käsiin on vielä suurempi uhka kuin pelkkien pienten muistilaitteiden. Tietokoneille tallennettu tietomäärä on usein moninkertainen muistitikkuihin nähden. Puhelimia ja kannettavia käytettäessä yksinkertainen tietoturvan parantaminen on tietojen tallentaminen muualle kun kannettavalle laitteelle. Suurimmassa osassa näistä laitteista on tietoliikenneyhteydet olemassa, jolloin selvästi pienempi uhka siinä, että langattomasti lähetettävät tiedot onnistutaan nappaamaan siinä välissä, kun tieto lähetetään esimerkiksi yrityksen tiedostopalvelimelle kuin että tieto jää kannettavalle laitteelle pitkäksi aikaa. Toinen vakava uhka on sähköpostin tallentuminen mobiililaitteelle. Sähköposteissa helposti lähetetään tärkeää tietoa ja niiden lukeminen tulisi vähintään estää salasanalla. Turvallisempi tapa käyttää sähköpostia mobiililaitteelta on käyttää internetin kautta käytettäviä webmaileja, joissa itse sähköpostit on tallennettuna sähköpostipalvelimelle, eikä lainkaan mobiililaitteelle. Ongelman laajuus on helposti havaittavissa esimerkiksi luvuista, kuinka paljon ihmiset unohtavat tavaroita pelkästään taksiin. Viimeisen kuuden kuukauden aikana pelkästään Lontoon taksiin unohdettujen mobiililaitteiden määrät ovat miltei 55 000 matkapuhelinta, yli 4 700 kämmen-tietokonetta, 3 100 kannettavaa tietokonetta ja yli 900 USB-tikkua [Digitoday] Varsinkin ihmisten siirtyminen uusiin älypuhelimisiin tulee moninkertaistamaan riskit salaamattoman tiedon joutumisen väärin käsiin.

4. Tietoturvastandardit BS7799 ja ISO 17799

Brittiläinen standardi-instituutti British Standards Institute (BSI) julkaisi ensimmäisen yrityksen tietoturvastandardin vuonna 1995. Se kirjoitettiin Britannian Kauppa- ja teollisuusministeriön toimesta. Muutaman version jälkeen kansainvälinen standardiorganisaatio ISO hyväksyi standardin ensimmäisen version vuonna 2000. Myöhemmin sitä on päivitetty toisella alkuperäiseen BS7799-standardiin lisätyllä toisella osalla, joka julkaistiin vuonna 1999, ja lisättiin kansainväliseen standardiin vuonna 2002. Standardiin on näiden kahden ensimmäisen osan lisäksi tulossa myös kolmas osa, jonka aiheena on tarkentaa riskien analysointia ja hallintaa. Tämä kolmas osa on julkaistu vasta osana brittiläisessä BS7799-standardia, muttei vielä kansainvälisessä ISO-standardissa. Tämän lisäksi odotettavissa on useita lisäosia ISO 17799 -standardiin, joista seuraavan pitäisi olla julkaisuvalmis vielä vuoden 2007 aikana ja jonka sisältönä

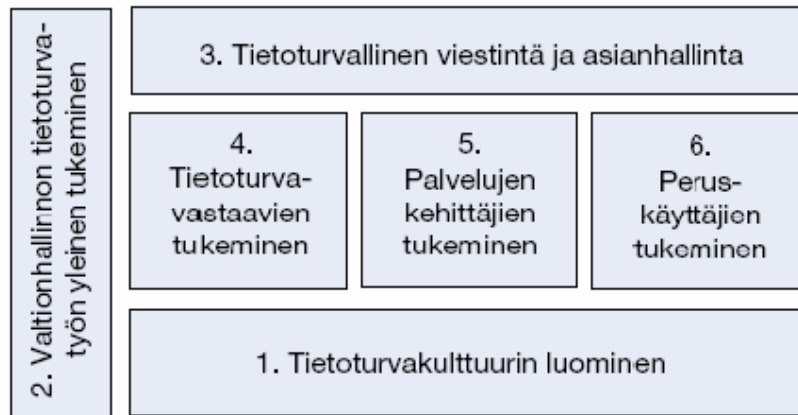
ovat ohjeet sertifiointiprosessiin. Lisäksi valmisteluvaiheessa olevia osia on mm. uusittu versio tietoturvallisuuden hallintasuunnitelman käyttöön otosta sekä ohjeet tietoturvallisuustason mittaamiseksi. [ISO17799Wiki]

Standardin noudattamisen tarkoituksena on rakentaa organisaatiolle tietoturvallisuuden hallintajärjestelmä. Jos organisaation haluaa sertifioida itsensä standardia vasten esimerkiksi todistaakseen sillä tavoin yhteistyökumppaneilleen luotettavuutensa, tullaan organisaation tietoturvallisuuden hallintajärjestelmä tarkistamaan valtuutetun asessorin toimesta. [Johansson, 2003] Saatu sertifikaatti sisältää tiedon siitä, mitkä organisaation osat serfioitiin sekä muita olennaisia tietoja kuten soveltamissuunnitelman. [Gamma, 2006] Sertifiointi kokonaisuudessaan on hyvin laaja prosessi ja yrityksen on hyvin tarkasti arvioida mitä hyötyä siitä yrityksen toiminnalle on ja miten kalliiksi sen tekeminen ja ylläpitäminen tulee maksamaan. Tietoturvallisuuteen sertifiointilla ei käytännössä ole merkitystä, jos standardia muuten noudatetaan, vaikka sitä ei virallisesti serfioida. Standardin toteuttamisen suurin hyöty onkin siinä, että sitä seuraamalla yrityksen ei itse tarvitse kehittää koko suunnitelmaa vaan voi käyttää oman tietoturvallisuuden parantamiseksi valmiiksi sitä varten käytettyä mallia. Standardin avulla tietoturvallisuuden merkitys on helpommin ymmärrettävissä, mikä helpottaa asian merkityksen hahmottamisessa. Esimerkillisenä yrityksenä suomessa toimii Tietoenator, jonka Processing & Network -osasto on ensimmäisenä it-alan yrityksenä saanut BS 7799 -sertifikaatin jo vuonna 1999. Ensimmäisenä valtionhallinnon virastona samainen sertifikaatti on myönnetty väestörekisterikeskukselle vuonna 2002.

5. Valtion tietoturvallisuuden kehitysohjelma

Suomen valtionvarainministeriö on asettanut valtionhallinnon tietoturvallisuuden johtoryhmän (VAHTI) hallinnon tietoturvallisuuden yhteistyön, ja ohjauksen kehittämiseksi. VAHTI asetti vuonna 2003 työryhmän valmistelemaan valtion tietoturvallisuuden kehitysohjelman, jonka tavoitteena on parantaa valtionhallinnon toimintojen luotettavuutta ja jatkuvuutta tietoturvallisuutta kehittämällä sekä edistää tietoturvallisuuden saattamista kiinteäksi osaksi valtionhallinnon kaikkea toimintaa. Tavoitteena on kehittää, yhteen sovittaa ja ohjata valtionhallinnon tietoturvallisuutta tietoyhteiskuntakehityksen, sähköisen hallinnon, toimintojen laadun ja varmistamisen, tietoturvauhkien sekä toimintojen ja tietojenkäsittelyn verkottumisen ja muun kehityksen edellyttämällä tavalla. [VM] Tarkoituksena on, että saatuja tuloksia hyödynnetään myös kunnallishallinnossa, sekä yrityksissä. Kuvassa 3 on nähtävillä, miten ensimmäiseksi tavoitteeksi on asetettu tietoturvakulttuurin luominen, joka sisältää tulostehokkuuden käyttämisen tietoturvallisuuden kehittämiskeinona sekä monipuolista eri taho-

jen tuottamaa turvakoulutusta. Valtion johtama tietoturvallisuuden kehitysohjelma aloittaa Suomessa vastaavan yleisen käytännön, joka Britanniassa on lähtenyt liikkeelle kansallisesta standardista. Opetusministeriölle on annettu tehtäväksi tietoturvallisuuden lisääminen peruskoulun, sekä keskiasteen opetusohjelmaan, jonka lisäksi korkeakoulutasoisessa opetuksessa tulee lisätä tietoturvallisuuden opetusta ja tutkimusta.



Kuva 3. Valtion kehitysohjelmakaavio.

Suurimpina nykyisinä ongelmina työryhmän tuloksina on saatu tärkeimpänä tietoturvatyön resurssien puute, joka on seurausta tietoturvallisuuden liian alhaisesta priorisoinnista suhteessa muihin tavoitteisiin. Toisena asiana on henkilöstön tietoturvatietoisuus, sekä kolmantena ongelmana erilaiset virukset ja muut haittaohjelmat. Julkisuudessa paljon näkyvyyttä saaneet palvelunestohyökkäykset, joita kohdistui toukokuussa 2007 Suomen Yleisradion verkkosivuja vastaan, ei raportissa ollut mainintaa, mikä osoittaa, miten huonosti tilanteeseen oli varauduttu. Tosiasia on kuitenkin se, että tiedostamattomiin uhkiin on vaikea varautua ja tämäkin asia varmasti nousee esille seuraavassa raportissa. Lähitulevaisuudessa ei kuitenkaan ole näkyvissä suuria muutoksia. Virukset, roskapostit ja riittämättömät resurssit jatkavat uhkina myös lähitulevaisuudessa. Uutena asiana lähitulevaisuuden haasteissa nähdään kuitenkin yleisesti tietomurrot ja verkkoterrorismi, mutta vielä niihin ei ole valmista ohjeistusta. Valtionvarainministeriön VAHTI-tietoturvaohjeet ovat tunnettuja ja arvostettuja ohjeita Suomessa. Ohjeet on yleensä pyritty laatimaan, hallinnollisen tietoturvan näkökohdasta ja sellaisiksi, etteivät niissä esitetyt asiat vanhene nopeasti. Näin perusohjeiden rinnalle on tullut käytännönläheisempiä, teknisiä ohjeita ja suosituksia. Tietoturvallisuuden kehittyminen EU:n alueella tullaan johtamaan pitkälti tulevaisuudessa perustetun verkko- ja tietoturvaviraston kautta. Suomi kävi kovan kamppailun, jotta virasto olisi sijoitettu Suo-

meen, koska alan osaaminen Suomessa on aivan maailman huippua, mutta EU päätti sijoittaa viraston Kreikkaan. Tämä ei kuitenkaan vähennä alaan panostamista Suomessa.

6. Yhteenveto

Yrityksen tietoturvaluus on asiana hyvin ajankohtainen varsinkin sen puutteellisuuden takia. Aiheeseen on vasta viimeaikoina tullut kansainvälisiä standardeja ja vielä myöhemmin Suomen valtion omat suunnitelmat, joiden avulla pyritään nostamaan yritysten tietoturvaluuden tasoa kokonaisuudessaan. Monilla suurilla yrityksillä on omat tietoturvasuunnitelmat, joita ne toteuttavat omalla tavallaan, mutta ne eivät välttämättä sovellu helposti varsinkaan toisen alan yrityksille. Tietoturvaluuden kehittäjinä ovat toimineet monikansalliset it-alan yritykset, mutta tietotekniikan yleistymisen laajasti lähes kaikkien alojen yrityksiin vaatii, että saadaan luotua yleiset ohjeet, jotka ovat sovellettavissa helpommin eri aloille. Tällä hetkellä yritykset eivät vielä ole kovin avoimia paljastamaan sitä, miten heidän oma tietoturvaluus järjestelmä toimii vaikka paljastamalla sen toimintaa pystyttäisiin kehittämään yleisesti ottamalla oppia sieltä, missä jokin asia tehdään parhaiten ja yhdistämällä eri yritysten tapoja yleiseksi standardiksi. Valtiollinen johtaminen tietoturvaluuden parantamiseksi tuottaa hitaasti tuloksia, mutta suunta on hyvin selkeä siihen, että tietoturvaluuden parantamisesta ollaan huolissaan ja sen parantamiseksi on vihdoin alettu tehdä työtä. Siksi tällaiset tutkimukset ovat juuri sitä mitä valtion tietoturvaluuden kehitysohjelmassa haluttiin.

Yleiset standardit, sekä valtion tietoturvaluuden kehitysohjelma antavat yrityksille hyvän pohjan parantaa omaa tietoturvaluuttaansa. Hyvä tietoturvaluuden taso ei kuitenkaan synny automaattisesti sitä kautta, että valtio opastaa ihmisiä parempaan tietoturvatietouteen, vaan yrityksiön on osattava käyttää näitä hyväkseen ja sitä kautta varmistettava oma turvallinen tulevaisuutensa.

Viiteluettelo

[Gamma, 2006] Gamma Secure Systems Limited, How 27000 Works. June 2006.

<http://www.gammassl.co.uk/bs7799/works.html>. [12.05.2007]

[Johansson, 2003] Matti Johansson, Tietoturvan hallinta. Tietoturvaluus nykyaikaisessa liiketoimintaympäris tössä *www.cs.helsinki.fi/group/turvasem/papers/johansson_bs7799.pdf* [12.05.2007]

[Pointsec] Pointsec Mobile Technologies

<http://www.pointsec.com/news/newsreleases/release.cfm?PressId=415> [12.05.2007]

[Digitoday] Digitoday Finland

http://www.digitoday.fi/page.php?page_id=14&news_id=200621873 [12.05.2007]

[BS7799Wiki] BS 7799 Part 1. http://en.wikipedia.org/wiki/BS_7799 [12.05.2007]

[ISO17799Wiki] ISO/IEC 17799 http://en.wikipedia.org/wiki/ISO_17799 [12.05.2007]

[VM] Valtionvarainministeriö

http://www.vm.fi/vm/fi/13_hallinnon_kehittaminen/09_Tietoturvallisuus/02_tietoturvaohjeet_ja_maaraykset/index.jsp [12.05.2007]

Watson-Crick-automaatit

Sirkka Eloranta

Tiivistelmä Tämän tutkielman aiheena on Watson-Crick-automaatit. Tutkielmassa esitellään Watson-Crick-automaatteja ja sen variantteja. Yhteydet Chomskyn kieliperheeseen käydään myös läpi ja esitellään myös hieman muita Watson-Crick-automaattien hyväksymien kielten ominaisuuksia. Lisäksi käsitellään rinnakkaisjärjestelmiä lyhyesti.

Avainsanat ja -sanonnat: Watson-Crick-automaatit, DNA-laskenta, komplementaarisuusrelaatio, rinnakkaiset kommunikaatiojärjestelmät.

CR-luokat: F.1.1, F.4.3.

1 Johdanto

Leonard Adlemanin [1996] kuuluisa koe antoi sysäyksen molekyyllilaskennan suosiolle, sillä Adleman pystyi todistamaan, että monimutkaisia kombinatorisia ongelmia voidaan ratkaista tehokkaasti DNA-laskennan avulla. Kokeeseen kuului Hamiltonin syklin etsiminen klassisessa kauppamatkustajan ongelmassa.

Adlemanin kokeessa koodattiin ongelmaan liittyvän graafin solmut ja kaaret DNA-molekyyleiksi ja saatua DNA:ta käsiteltiin koeputkissa DNA-operaatioilla, joiden perusteella saatiin ratkaistua syötteenä annetussa graafissa määritelty kauppamatkustajan ongelman tapaus. Useita molekyylejä voitiin käyttää tekemään laskentoja samanaikaisesti.

DNA-laskenta on kvanttilaskennan ohella yksi kiinnostavimpia tutkimuskohteita tällä hetkellä. Tulevaisuudessa voi olla mahdollista korvata biomolekyyleihin, esimerkiksi DNA:han, perustuvalla tietokoneella piihin pohjautuva tietokone. DNA-laskennan virhealttius ja epävarmuus voivat kuitenkin koitua esteeksi sen tulevaisuudelle. Todennäköisempi skenaario onkin, että biotietokoneita käytettäisiin rinnakkain muun teknologian ohella.

DNA-laskentaan liittyy läheisesti Watson-Crick-automaatin käsite. Nimensä automaatti on saanut DNA:n rakenteen ratkaisijoista ja se muistuttaakin rakenteeltaan DNA:ta. Automaatti voi lukea kahta nauhaa samanaikaisesti ja sen rakenteeseen kuuluu komplementaarisuusrelaatio.

2 Peruskäsitteitä

Watson-Crick-automaatit ovat abstrakteja tietokoneen malleja kuten Turingin kone, tietojenkäsittelytieteen peruskäsite.

Tutkielmassa käytetään soveltuvin osin samoja merkintätapoja, joita Hopcroft ja muut [2001] käyttävät.

2.1 Äärellinen automaatti

Äärellinen automaatti on järjestelmä, joka lukee syötettä ja vaihtaa tilaa siirtymärelaation määrittämällä tavalla. Äärellisen automaatti määrittellään seuraavasti:

$$A = (Q, \Sigma, \delta, q_0, F),$$

missä Q on tilojen joukko, Σ on syöteaakkosto, δ on siirtymärelaatio, q_0 on alkutila ja F on lopputilojen joukko.

Automaatin toiminta alkaa alkutilasta ja luettuaan syötettä se vaihtaa tilaansa siirtymärelaation δ määrittämällä tavalla. Syöte muodostuu aakkoston Σ merkeistä. Automaatti lopettaa toimintansa syötteen luettuaan.

Siirtymärelaatiossa δ tilan ja syötteen yhdistelmä kuvautuu tilaksi $q \in Q$. Esimerkiksi, jos automaatti lukee merkin a ja on tilassa q ja siirtymärelaatiossa on määritelty $\delta(q, a) = q'$, automaatti siirtyy lukemaan syötettä seuraavaan syötemerkkiin ohittaen merkin a ja siirtyy tilaan q' .

Automaatin epädeterministisyys tarkoittaa, että se voi siirtyä samalla (tila, merkki) -parilla useampaan kuin yhteen tilaan. Epädeterminismi ei kuitenkaan lisää äärellisen automaatin ilmaisuvoimaa, vaan lähinnä yksinkertaistaa sen rakennetta.

2.2 Muodolliset kielet

Kieliopit ovat järjestelmiä, joilla tuotetaan aakkostosta kieliä. Automaatit toimivat näihin kielioppeihin nähden toiseen suuntaan, sillä ne tunnistavat kieliä ja hyväksyvät niitä, eivätkä muodosta niitä.

Chomskyn kielihierarkia Chomskyn kielihierarkiaan kuuluvat säännölliset, kontekstittomat, kontekstilliset ja tyyppin 0 kielet. Näistä kielistä yksinkertaisimpia ovat säännölliset kielet ja monimutkaisimpia ovat tyyppin 0 kielet kielet.

Automaatin kieli Automaatit tunnistavat kieliä. Äärellinen automaatti $A = (Q, \Sigma, \delta, q_0, F)$ tunnistaa kielen

$$L(A) = \{w \mid \delta(q_0, w) \in F\}.$$

Tietyt automaattiluokat tunnistavat tiettyjä kieliperheitä. Esimerkiksi, äärelliset automaatit tunnistavat säännöllisten kielten perheen ja Turingin kone tunnistaa tyyppin 0 kielet.

Tässä tutkielmassa esiintulevia kieliperheitä ovat

1. säännölliset kielet: $REG = \{L(A) \mid A \text{ on äärellinen automaatti}\}$,
2. kontekstittomat kielet: $CF = \{L(A) \mid A \text{ on pinoautomaatti}\}$,
3. kontekstiset kielet: $CS = \{L(A) \mid A \text{ on lineaarisesti rajoitettu automaatti}\}$
ja
4. tyyppin 0 kielet: $RE = \{L(A) \mid A \text{ on Turingin kone}\}$.

2.3 Gsm

Gsm on lyhenne sanoista generalized sequential machine ja se on muuten kuin äärellinen automaatti, mutta siihen liittyy tulosnauha syötenauhan lisäksi. Gsm on järjestelmä

$$g = (Q, \Sigma_1, \Sigma_2, q_0, F, \delta),$$

missä Q , F ja q_0 ovat kuten äärellisessä automaatissa, mutta Σ_1 on syöteaakkosto ja Σ_2 on tuloaakkosto ja $\delta : Q \times \Sigma_1 \rightarrow P_f(\Sigma_2^* \times Q)$.

Gsm-kuvausta voi käyttää kuvaamaan syöteaakkoston sanoja tuloaakkoston sanoiksi.

3 Watson-Crick-automaatit

3.1 DNA-molekyyli

DNA on ollut lähtökohtana Watson-Crick-automaatin kehittämiseksi, joten tässä käydään DNA:n rakenne lyhyesti läpi. DNA on ribonukleotideiksi kutsutuista monomeereistä koostuva polymeeri. Ribonukleotidilla on fosforiosa, sokeriosa ja emäsosa, jolla on tunnisteenä DNA-aakkoston merkki aakkostosta $\{A, T, C, G\}$. Nämä tietenkin vastaavat kirjaintunneita emäsosien adenini, tymiini, sytosiini ja guaniini englanninkielisistä nimistä.

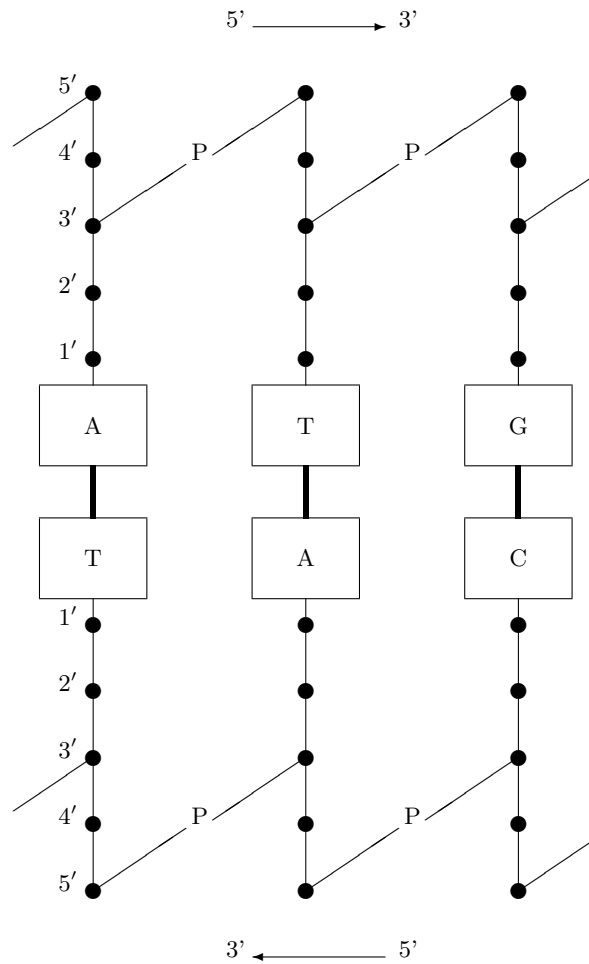
Sokeriosaan kuuluu viisi hiiliatomia, joista 5'-hiiliatomissa on kiinni fosforiosa. Nukleotidit liittyvät toisiinsa fosforisidoksella. 5'-hiiliatomiin kiinnittynyt fosforiosa liittyy toisen nukleotidin sokeriosan 3'-hiiliatomiin kovalenttisidoksella. DNA-molekyylin vastakkaiset säikeet ovat vastakkaisuuntaiset, jolloin toinen kuvataan 5'-3'-suuntaisena ja toinen 3'-5'-suuntaisena.

DNA-molekyylin ribonukleotidit ovat toisissaan kiinni kahdentyyppisillä sidoksilla. Emäsosat ovat kiinni toisissaan vetysidoksin, jotka ovat heikompiä kuin fosforiosan kovalenttisisidos, minkä vuoksi kaksoissäikeen säikeet voi erottaa toisistaan erottamatta nukleotidiketjua.

DNA:lla on kaksoiskierteinen rakenne, joka voidaan automaattien tasolla yksinkertaistaa päällekkäiset ylä- ja alanauhan sisältäväksi rakenteeksi. Kaksoiskierteisellä rakenteella on kuitenkin olennainen merkitys DNA-molekyylin toiminnalle luonnossa. Kuvassa 1 havainnollistetaan DNA:n rakennetta.

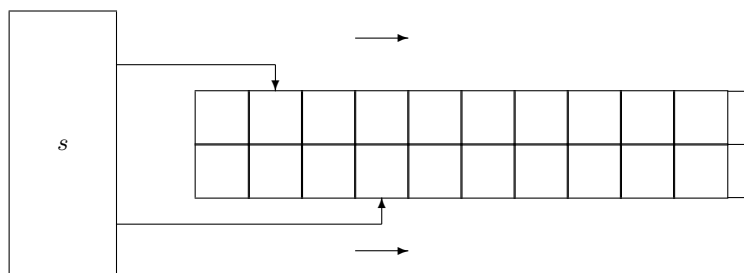
3.2 Watson-Crick-automaatti yleisesti

Watson-Crick-automaatin olennaisin ero edellä esiteltyyn äärelliseen automaattiin on syötenauhan kaksoismerkkijonorakenne, johon kuuluu kaksi nauhaa ja näiden välille kuuluva symmetrinen komplementaarisuusrelaatio. Watson-Crick-automaatilla on kaksi lukupäätä, joista toinen lukee ylänauhaa ja toinen alanauhaa.



Kuva 1. DNA-molekyylin rakenne. Vahvemmat viivat kuvaavat sidoksia emäsosien välillä ja laatikot kuvaavat emäsosia. Emäsosista lähtevät viivat yhdistävät hiiliatomeja, jotka on numeroitu 1',2',3',4',5 ja merkitty mustilla pallukoilla. P merkitsee fosforisidosta nukleotidien välillä.

Tavallisessa versiossa lukupäät etenevät samaan suuntaan. Voidaan myös määrittää Watson-Crick-automaatin versio, jossa lukupäät etenevät vastakkaisiin suuntiin, eli vastakkaisuuntainen Watson-Crick-automaatti. Tällainen versio vastaisi aidommin DNA-molekyylin rakennetta. Muista versioista tutkielmas-
sa esitellään Watson-Crick-muunnin. Kuva 2 esittää Watson-Crick-automaatin rakennetta.



Kuva 2. Äärellinen Watson-Crick-automaatti.

3.3 Watson-Crick-automaatin määritelmä

Täsmällisemmin ilmaistuna Watson-Crick-automaatti on järjestelmä

$$A = (\Sigma, \rho, Q, q_0, F, \delta),$$

missä

1. Σ on syöteakkosto, josta muodostetaan syötemerkkijonot,
2. ρ on komplementaarisuusrelaatio, joka määrittää ylä- ja alanauhan vastinmerkkien välisen relaation, $\rho \subseteq \Sigma \times \Sigma$,
3. Q on automaatin tilojen joukko,
4. q_0 on automaatin alkutila, $q_0 \in Q$,
5. F on lopputilojen joukko, $F \subseteq Q$ ja
6. δ on siirtymärelaatio, $\delta : Q \times \begin{pmatrix} \Sigma^* \\ \Sigma^* \end{pmatrix} \rightarrow P(Q)$, jolle pätee, että $\delta(q, \begin{pmatrix} x \\ y \end{pmatrix}) \neq \emptyset$ äärelliselle määrälle kolmikkoja $(q, x, y) \in Q \times \Sigma^* \times \Sigma^*$.

Siirtymärelaatio δ voidaan korvata ns. uudelleenkirjoitussäännöillä, joissa $q \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \rightarrow \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} q'$ tarkoittaa samaa kuin $q' \in \delta(q, \begin{pmatrix} x_1 \\ x_2 \end{pmatrix})$.

Tällöin saadaan edellisen määritelmän kanssa yhtäpitävä määritelmä Watson-Crick-automaatille:

$$A = (\Sigma, \rho, Q, q_0, F, P),$$

missä P on uudelleenkirjoitussäännöllä korvattu δ .

Jatkossa merkitään Watson-Crick-automaattia lyhyemmin merkinnällä WK-automaatti. Lyhenne WK tulee erisnimen Watson-Crick alku- ja loppukirjaimista, ja sitä käyttävät myös Pään ja muut [1998].

3.4 Watson-Crick-automaatin rajoitetut versiot

WK-automaatin rajoitettujen versioiden avulla voidaan tutkia helposti kielten hyväksymisominaisuuksia. Paunin ja muiden [1998] tapaan rajoittamattomasta WK-automaatista käytetään merkintään $AWK(\alpha)$.

WK-automaatti on

1. NWK , eli tilaton, jos $Q = F = \{q_0\}$,
2. FWK , eli lopputilallinen, jos $F = Q$,
3. SWK , eli yksinkertainen, jos kaikille $q \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \rightarrow \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} q'$ pätee, että $x_1 = \lambda$ tai $x_2 = \lambda$, tai
4. $1WK$, eli 1-rajoitettu, jos kaikille $q \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \rightarrow \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} q'$ pätee, että $|x_1 x_2| = 1$.

Lisäksi WK -automaatilla voi olla useita näistä ehdoista samaan aikaan voimassa, jolloin merkitään WK-automaattia merkinnällä XWK , jossa

$$X \in \{NS, N1, FS, F1\}.$$

3.5 Watson-Crick-muunnin

Watson-Crick-muunnin on kohdassa 2.3 esitellyn gsm:n WK-variantti.

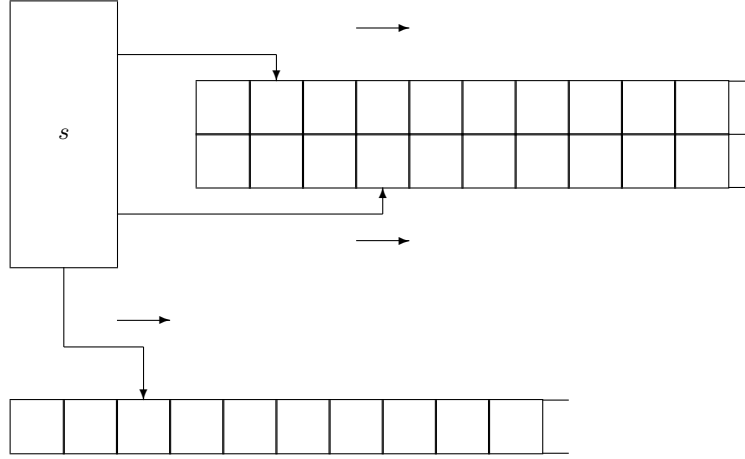
WK-automaatilla on tavallinen yksisäikeinen nauha tulosnauhana, johon kirjoitetaan tietoa. On myös esitelty variantti, jossa tulosnauha on myös WK-nauha, mutta sitä ei tarkastella tässä tutkielmassa.

WK-muunnin määritellään seuraavasti:

$$g = (\Sigma_I, \rho_I, \Sigma_O, Q, q_0, F, \delta),$$

missä Σ_I on syöteakkosto, ρ_I on komplementaarisuusrelaatio syöteakkostolle, Σ_2 on tulosakkosto, F on lopputilojen joukko ja siirtymärelaatiolle δ pätee $\delta : Q \times \begin{pmatrix} \Sigma_I^* \\ \Sigma_I^* \end{pmatrix} \rightarrow P_f(\Sigma_0^* \times Q)$, missä $\delta(q, \begin{pmatrix} u \\ v \end{pmatrix}) \neq \emptyset$ äärelliselle määrälle kolmikoita $(q, u, v) \in Q \times \Sigma_I^* \times \Sigma_I^*$.

Kuva 3 havainnollistaa WK-muuntimen rakennetta.



Kuva 3. Äärellinen Watson-Crick-muunnin

4 WK-automaatin kielet

4.1 Käytetyistä merkinnöistä

Tässä luvussa käytetään Paunin ja muiden [1998] määrittelemiä WK-automaatteihin liittyviä merkintöjä.

Jotta voidaan havainnollistaa paremmin DNA-molekyylin kaksoismerkkijonorakennetta, korvataan $\Sigma^* \times \Sigma^*$ merkinnällä $\begin{pmatrix} \Sigma^* \\ \Sigma^* \end{pmatrix}$ ja sen elementit (x_1, x_2) merkinnällä $\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$.

Watson-Crick-määrittelyjoukko on $WK_\rho(\Sigma)$, joka muodostuu Σ :n ja ρ :n avulla kaikista sallituista yhdistelmistä, jossa siis jokainen kaksoismerkkijono on vastinrelaation mukaisesti muodostettu aakkoston Σ merkeistä. Joukko $WK_\rho(\Sigma)$ määritellään seuraavasti:

$$\begin{bmatrix} V \\ V \end{bmatrix}_\rho = \left\{ \begin{bmatrix} a \\ b \end{bmatrix}, \mid a, b \in \Sigma, (a, b) \in \rho \right\},$$

$$WK_\rho(\Sigma) = \begin{bmatrix} V \\ V \end{bmatrix}_\rho^*.$$

Joukkoon kuuluvat elementit $\begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$ ovat hyvinmääritettyjä kaksoismerkkijonoja komplementaarisuusrelaation suhteen, eli molekyylejä, toisin kuin parit $\begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$, jotka eivät välttämättä ole komplementaarisuussuhteessa toisiinsa.

4.2 WK-automaatin kielet

Kun tahdotaan tarkastella ylemmän nauhan kieliä, WK-automaatin hyväksymä kieli määritellään:

$$L_u(A) = \{w_1 \in \Sigma^* \mid q_0 \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \Rightarrow^* \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} q_f \text{ kun } q_f \in F, \\ \text{ja } w_2 \in \Sigma^*, \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \in WK_\rho(\Sigma)\}.$$

Alemman nauhan kielet tai molekyylien kieliä ei ole tarpeen tarkastella, sillä komplementaarisuusrelaation vuoksi ne voi päätellä ylemmän nauhan kielistä.

WK-automaattien tapauksessa on myös hyödyllistä määritellä niin sanottu ohjauskieli. Ohjauskielessä otetaan tunnistetun merkkijonon sijasta huomioon siirtymärelaatio. Tämä tarkoittaa sitä, että jokaiselle P :n siirtymäsäännölle annetaan tunnus kuvauksella $P \rightarrow Lab$. WK-automaatilla suoritettu onnistunut laskenta on $\sigma : q_0 w \Rightarrow^* w q_f, w \in WK_\rho(\Sigma), q_f \in F$, johon siis kuuluvat laskennat, jotka etenevät alkutilasta lopputilaan. Käytetään merkintää $e(\sigma)$ ohjaussanasta, joka muodostuu laskennassa σ .

Rajoitettujen kokoisella syöteaakkostolle saattaa olla kätevää käyttää tällaisia ohjauskieliä, sillä useimmiten ratkaistavat ongelmat ovat kuitenkin suuremmasta aakkostosta peräisin. WK-automaatin ohjauskieli on

$$L_{ctr}(A) = \{e(\sigma) \mid \sigma : q_0 w \Rightarrow^* w q_f, w \in WK_\rho(\Sigma), q_f \in F\}.$$

Käsiteltävät WK-automaattien kielet ovat siis

$$L_\alpha(A), \alpha \in \{u, ctr\}.$$

4.3 Rajoitettujen WK-automaattien kielet

Merkinnät $AWK(\alpha)$, $NWK(\alpha)$, $FWK(\alpha)$, $SWK(\alpha)$, $NSWK(\alpha)$, $N1WK(\alpha)$, $F1WK(\alpha)$ ja $FSWK(\alpha)$ tarkoittavat vastaavien automaattien hyväksymiä kieliä muotoa $L_\alpha(A)$, $\alpha \in \{u, ctr\}$. Pään ja muut [1998] todistivat useamman tässä käsitellyn sisältyvyyden.

Määritelmien perusteella saadaan seuraavat sisältymiset suoraan, kun kummassakin puolella on samat rajoitukset, mutta toisessa jotain enemmän:

1. $XKW(\alpha) \subseteq AWK(\alpha)$, $\alpha \in \{u, ctr\}$, $X \in \{N, F, S, 1, NS, N1, FS, F1\}$,
2. $NWK(\alpha) \subseteq FWK(\alpha)$, $NSWK(\alpha) \subseteq FSWK(\alpha)$, $N1WK(\alpha) \subseteq F1WK(\alpha)$, $\alpha \in \{u, ctr\}$, ja

3. $XSWK(\alpha) \subseteq SWK(\alpha)$, $X1WK(\alpha) \subseteq 1WK(\alpha)$, $X1WK(\alpha) \subseteq XSWK(\alpha)$,
 $1WK(\alpha) \subseteq SWK(\alpha)$, $\alpha \in \{u, ctr\}$, $X \in \{N, F\}$.

Lisäksi pätevät $REG \subseteq 1WK(\alpha)$, $AWK(\alpha) \subseteq CS$ ja $AWK(u) \subseteq 1WK(u)$, $\alpha \in \{u, ctr\}$, jotka nähdään siitä, että $1WK$ voi toimia kuten äärellinen automaatti, ja siitä, että yksinkertaisilla säännöillä voidaan korvata mielivaltaiset säännöt vähentämättä automaatin ilmaisvoimaa, mitä ei tässä käsitellä sen tarkemmin.

Lisäksi Pään ja muut [1998] todistivat useita lauseita liittyen erikseen kieleen L_u ja L_{ctr} . He todistivat muun muassa, että $TH = 1WK(u)$, eli kaksilukupäinen automaatti on ekvivalentti $1WK(u)$:n kanssa. Myöhemmin Kuske ja Weigel [2005] tekivät joitakin täydennyksiä näihin tuloksiin.

Kieliin L_u liittyvät sisältyvydet

1. $N1WK(u) \subseteq NSWK(u)$,
2. $NSWK(u) \subseteq REG \subseteq F1WK(u) \subseteq FSWK(u)$,
3. $NSWK \subseteq NWK(u)$ ja
4. $NWK(u) \subset FSWK(u) = FWK(u) = AWK(u) = SWK(u) = 1WK(u) \subset CS$.

Kaikkia samoja todistuksia ei ole onnistuttu todistamaan kielelle $L_{ctr}(A)$, sillä osassa todistuksissa $L_{ctr}(A)$ olisi muuttunut. Saatiin sisältyvydet

1. $N1WK(ctr) \subseteq NSWK(ctr)$,
2. $N1WK(ctr) \subseteq F1WK(ctr)$,
3. $NSWK(ctr) \subseteq NWK(ctr) \subseteq FWK(ctr)$,
4. $NSWK(ctr) \subseteq FSWK(ctr)$,
5. $F1WK(ctr) \subseteq FSWK(ctr)$,
6. $F1WK(ctr) \subseteq 1WK(ctr)$,
7. $FSWK(ctr) \subseteq SWK(ctr)$,
8. $FSWK(ctr) \subseteq FWK(ctr) \subseteq AWK(ctr)$,
9. $REG \subseteq 1WK(ctr) \subseteq SWK(ctr) \subseteq AWK(ctr)$, ja
10. $AWK(ctr) \subseteq CS$.

Näin ollen tilattomat versiot WK-automaateista ovat heikoimpia, mitä tulee kielten hyväksymisominaisuuteen.

4.4 Tyypin 0 kielet

Tyypin 0 kielien kielien päästään käyttämällä kutistuskoneita, jotka DNA-laskennan mielessä ovat muutenkin mielekkäitä rajoitetun DNA-aakkoston koon vuoksi.

Tällaisista kutistuskoneista on tutkittu heikkoa koodausta ja determinististä gsm:ää eri WK-automaattien varianttien suhteen. Näin on saatu kyseisten automaattien kielet kattamaan kaikki tyypin 0 kielet.

Heikolla koodauksella tarkoitetaan morfismia $h : V^* \rightarrow U^*$, jolle pätee, että $h(a) \in U \cup \{\lambda\}$, jokaiselle $a \in V$. Jos

$$h : (V_1 \cup V_2)^* \rightarrow V_1^*$$

on määritelty $h(a) = a$ kaikille $a \in \Sigma$ ja $h(a) = \lambda$ muuten, niin koodaus on projektio.

Toinen kutistusmekanismi, jota tässä käsitellään on deterministinen gsm. Gsm on esitelty kohdassa 2.3.

Heikko koodaus Heikkoa koodausta käyttäen saadaan eri WK-variantit kuvattua tyyppin 0 kieliin.

Jokainen $L \in RE$ voidaan muodostaa projektioilla kielten $L_1 \in NWK(u)$ ja $L_2 \in REG$ leikkauksesta. Lisäksi se voidaan muodostaa projektioilla kielestä $L' \in AWK(u)$ tai kielten $L_1 \in FSWK(u)$ ja $L_2 \in REG$ leikkauksesta projektioilla.

Jokainen $L \in RE$ on heikko koodaus jostain kielestä $AWK(u)$, $FSWK(u)$, $SWK(u)$ ja $1WK(u)$.

Deterministisen gms:n kuvaus Gsm-kuvausta voi käyttää kuvaamaan kieliä tyyppin 0 kielten perheeseen. Jokainen tyyppin 0 kieli on gsm:n kuva kielestä perheessä $XWK(ctr)$ missä $X \in \{A, N, F, S, 1, NS, NF, F1, FS\}$.

Päun ja muut [1998] todistivat, että jokainen $L \in RE$ voidaan esittää muodossa $L = g(g_\Sigma(WK_\rho(\Sigma)))$, missä g on tavallinen gsm-kuvaus ja g_Σ on WK-gsm-kuvaus.

Watson-Crick-muuttimella on mahdollista kuvata hyvin yksinkertaisia kieliä tyyppin 0 kielten perheeseen.

5 Komplementaarisuusrelaation rooli

Komplementaarisuusrelaation rooli WK-automaatissa on edelleen epäselvä siitähän huolimatta, että komplementaarisuuden avulla saadaan käyttöön DNA-laskennan malleille Twin-Shuffle-kielen universaalisuus, mitä muun muassa Rozenberg ja Salomaa [1999] tähdentävät. Tämä johtuu siitä, että Twin-Shuffle-kielen perusvariantti aakkostolla $\{0, 1, \bar{0}, \bar{1}\}$ on analoginen Watson-Crick-automaatin kanssa DNA-aakkostolla $\{A, T, C, G\}$.

Kuske ja Weigel [2005] kuitenkin todistivat, että jokainen WK-automaatin kieli voidaan generoida WK-automaatilla, jolla on injektiivinen komplementaarisuusrelaatio, ja että yksittäisen automaatin tasolla komplementaarisuusrelaation rooli on vähäinen.

6 Rinnakkaiset kommunikaatiojärjestelmät ja WK-automaatit

6.1 Rinnakkaiset kommunikaatiojärjestelmät

WK-automaateissa ei itsessään ole ominaisuuksia tukemaan massiivista rinnakkaisuutta, joka nähdään avaintekijänä DNA-laskennan tulevaisuudelle. Ratkai-

suksi on esitetty ainakin rinnakkaisia kommunikaatiojärjestelmiä Watson-Crick-automaateilla.

Rinnakkaiset kommunikaatiojärjestelmät n komponentilla ovat järjestelmiä PCFAS, joissa komponentteina ovat kyselysymboleilla kommunikoivat äärelliset rinnakkain toimivat automaattit. PCFAS n komponentilla määritellään:

$$\Phi = (\Sigma, A_1, A_2, \dots, A_n, K),$$

missä

1. Σ on syöteaakkosto,
2. jokaiselle $1 \leq i \leq n$, A_i on äärellinen automaatti $A_i = (Q_i, \Sigma, \delta_i, q_i, F_i)$, jossa Q_i on automaatin A_i tilojen joukko, q_i on sen alkutila, F_i on sen lopputilojen joukko ja δ_i on sen siirtymärelaatio $\delta_i : Q_i \times (\Sigma \cup \{\lambda\}) \rightarrow 2^{Q_i}$ ja
3. $K = \{K_1, K_2, \dots, K_n\} \subseteq \bigcup_{i=1}^n Q_i$ on kyselytilojen joukko.

Watson-Crick-automaattien yhteyttä rinnakkaisiin kommunikaatiojärjestelmiin tutki jo Cojocarú [2004], joka todisti, että WK-automaatti on ekvivalentti kaksikomponenttisen PCFAS:n kanssa. Tästä seurasi, että erilaiset kaksikomponenttiset PCFAS-järjestelmät hyväksyvät kaikki tyyppin 0 kielet.

6.2 Rinnakkaiset kommunikaatiojärjestelmät WK-automaateilla

Vastaava järjestelmä kuin PCFAS WK-automaateilla on PCWKS(n), joka on järjestelmä

$$\Phi = (\Sigma, \rho, A_1, A_2, \dots, A_n, K),$$

missä Σ on syöteaakkosto, ρ on komplementaarisuusrelaatio,

$$A_i = (\Sigma, \rho, Q_i, q_i, F_i, \delta_i), \quad 1 \leq i \leq n,$$

ovat WK-automaatteja ja K on kuten järjestelmässä PCFAS.

Edellisessä luvussa käsiteltiin komplementaarisuusrelaation roolia. Huomattavaa on, että Czeizler ja Czeizler [2006] todistivat, että järjestelmässä PCWKS(n) komplementaarisuusrelaatiolla on aktiivinen rooli ja se hyväksyy enemmän kieliä kuin vastaava järjestelmä injektiiivisellä komplementaarisuusrelaatiolla. Komplementaarisuusrelaatiolla, joka ei ole injektiiivinen, varustettu PCWKS(n) hyväksyy myös joitakin ei-säännöllisiä kieliä, joita vastaava järjestelmä injektiiivisellä relaatiolla ei hyväksy.

7 Lopuksi

Watson-Crick-automaatteja tullaan tarkastelemaan tulevaisuudessa paljonkin, sillä vieläkin ei tiedetä riittävästi puhtaiden WK-kielten liitännöistä eri kieli-perheisiin.

Jonkin verran on tuloksia siitä kuinka WK-automaatteja voisi hyödyntää rinnakkaisjärjestelmissä, mutta tuloksia on vielä vähäisesti. Saadut tulokset ovat kuitenkin lupaavia DNA-laskennan tulevaisuuden kannalta.

Vaikka DNA-laskenta ei osoittautuisikaan kilpailukykyiseksi ja riittävän potentiaalisiksi tutkimusalueeksi, tarjoavat ne kuitenkin lähestymistavan tutustua luonnon omaan laskentatapaan.

Viitteet

- [Adleman, 1996] Leonard M. Adleman, On constructing a molecular computer. In: Richard J. Lipton and Eric B. Baum (eds.), *DNA Based Computers*. American Mathematical Society, 1996, 1-21.
- [Cojocaru, 2004] Lilian Cojocaru, Computational Models: Watson-Crick automata and PCFAS with two components: a computational power analogy. In: *Proceedings of the 1st Conference on Computing Frontiers*, 2004, 150-161, ACM Press.
- [Czeizler and Czeizler, 2005] Elena Czeizler and Eugen Czeizler, On the power of parallel communicating Watson-Crick automata systems. *Theoret. Comput. Sci.*, **358**, 2006, 142-147.
- [Hopcroft *et al.*, 2001] John E. Hopcroft, Rajeew Motwani and Jeffrey D. Ullman, *Introduction to Automata Theory, Languages, and Computation*, Addison-Wesley, 2001.
- [Kuske and Weigel, 2005] D. Kuske and P. Weigel, The role of the complementarity relation in Watson-Crick automata and sticker systems, *Lecture Notes in Computer Science*, **3340**, 2004, 272-283.
- [Păun and Păun, 1999] A. Păun and M. Păun, State and transition complexity of Watson-Crick finite automata, *FCT'99 Lecture Notes in Computer Science*, **1684**, 1999, 409-420.
- [Păun *et al.*, 1998] G. Păun, G. Rozenberg and A. Salomaa, *DNA Computing. New Computing Paradigms*, Springer, Berlin, 1998.
- [Petre, 2003] E. Petre, Watson-Crick ω -automata, *J. Autom. Lang. Comb.*, **8**(1), 2003, 59-70.
- [Rozenberg and Salomaa, 1999] G. Rozenberg, A. Salomaa, DNA computing: New ideas and paradigms, *Lecture Notes in Computer Science*, **1644**, 1999, 106-118.

Käytettävyys ja visuaalinen suunnittelu

Tuomo Hakaoja

Tiivistelmä.

Tutkielmassa tarkastelen visuaalisen suunnittelun antamia eväitä käytettävyydelle sekä mietin, miten käytettävyys on soveltanut periaatteita omiin tarkoituksiinsa. Tutkielmassa haen sekä yhtäläisyyksiä että eroja visuaalisen suunnittelun ja käytettävyyden välille.

Avainsanat ja -sanonnat: Visuaalinen suunnittelu, käytettävyys, hahmolait.

CR-luokat: H.5

1. Johdanto

Tässä tutkielmassa perehdyn visuaaliseen suunnitteluun ja tutkin, mitä visuaalisen suunnittelun periaatteet antavat käytettävyyden tutkimiselle. Visuaalinen suunnittelu on ollut olemassa jo pitkään ennen käsitettä käytettävyys. Voidaan sanoa, että käytettävyys pohjautuu osaltaan visuaalisen suunnittelun periaatteisiin. Käytettävyys suuntautuu kuitenkin voimakkaasti tekniikkaan ja vuorovaikutukseen sen toimintaympäristön asettamien vaatimusten johdosta.

Luvussa kaksi syvennyn visuaaliseen suunnitteluun, käyn läpi tarvittavia termejä ja luon pohjan visuaalisen suunnittelun lähtökohtiin. Visuaalinen suunnittelu käsittää lukemattoman määrän erilaisia julkaisumuotoja. Tässä tutkimuksessa tarkoitan visuaalisella suunnittelulla pääasiassa printtijulkaisuja, kuten mainosmateriaaleja.

Kolmannessa luvussa pohdin, missä visuaalisen suunnittelun ja käytettävyyden periaatteet kohtaavat. Lisäksi haen yhtäläisyyksiä suunnittelun menetelmistä sekä toteutusmetodeista. Tutkin myös, millä tavalla visuaalisen suunnittelun periaatteet ovat muuntuneet käytettävyyden tutkimisen tarpeisiin.

Neljännessä luvussa tarkastelen, miten käytettävyys on tähän mennessä otettu huomioon visuaalisissa julkaisuissa. Käytettävyys on paljon muutakin kuin visuaalista suunnittelua. Tämän vuoksi yritän vastavuoroisesti tuoda esiin tapoja, joilla visuaaliset suunnittelijat voisivat parantaa visuaalisten suunnitteluperiaatteiden mukaan tehtyjä julkaisuja hyväksi käyttäen käytettävyyden periaatteita.

Mielenkiintoiseksi tutkimuksen tekee se, että vaikka käytettävyyttä ja visuaalista suunnittelua on tutkittu paljon, en ole löytänyt materiaalia, joka viittaisi suoraan visuaaliseen suunnitteluun ja käytettävyyteen muualla kuin web-sivustoilla. Tutkimuksen mielenkiintoinen kysymys kuuluu: ”Tarkoittaako vi-

suaalisesti hyvännäköinen samaa kuin käytettävä ja tarvitseeko visuaalisen tuotoksen aina edes olla käytettävä?” Ja toisaalta, voiko tuote olla käytettävä, muttei visuaalisesti hyvännäköinen?

Tavoitteeni on, että tutkielmasta käy ilmi visuaalisen suunnittelun ja käytettävyyden yhtäläisyydet sekä visuaalisen suunnittelun periaatteiden muuntautuminen käytettävyyden tutkimustarkoituksiin.

2. Visuaalinen suunnittelu

Tässä luvussa kerron tarkemmin visuaalisen suunnittelun perusteista. Käyn läpi suunnittelun eri lähtökohtia sekä kerron hyvän julkaisun perusteet. Haen esiin niitä asioita, joita myöhemmin lähestyn enemmän käytettävyyden näkökulmasta. Selvitän aluksi muutamia visuaalisen suunnittelun perustermejä.

2.1. Missä visuaalista suunnittelua tarvitaan?

Tänä päivänä graafisia tuotteita näkee kaiken aikaa. Bussipysäkit, kauppojen ikkunat, maitopurkit, sanoma- ja aikakauslehdet, vessapaperit ja tietenkin kaikki esitteet ja mainokset. Listaa voisi jatkaa koko sivun mittaiseksi. Markkinointi ja mainonta elävät parhaillaan nousukautta. Yritykset yrittävät mainonnan avulla vaikuttaa kuluttajiin ja sitä kautta saada lisää myyntiä tai tunnettuutta omille tuotteilleen. Jokainen yritys haluaisi olla muita näkyvämpi ja siksi ne panostavat mainontaan. Toikkanen (2003, s.13) kertoo eri julkaisujen suunnittelusta, että kaikki kopioitavat ja monistettavat tiedotteetkin, joita toimistoissa syntyy, täytyy suunnitella ja toteuttaa yhtä huolellisesti kuin painettavaksi tarkoitetut painotuotteet.

Vaikka itse visuaalinen suunnittelu ei ole mitään avaruustiedettä, on asiaan perehtymättömän todella vaikea saada aikaan mainos, joka vetoaa kohdeyleisöön. Jokainen osaa mielestään valmistaa keskinkertaista mainosmateriaalia, mutta mainosten timantit hiotaan osaavissa mainostoimistoissa. Kaikissa laadukkaissa mainoksissa on käytetty hyväksi visuaalista suunnittelua. Se on suorastaan elinehto, jotta mainos on laadukas. Jokainen itse mainosta näperrellyt tietää, kuinka hankalaa se loppujen lopuksi on. Amatöörin tehdessä mainosta kannattaa panostaa yksinkertaisuuteen, selkeyteen ja kieleen. Kielellisesti selkeä mainos ajaa varmasti osittain asiansa. Tekstillä ja kuvilla täyteen ahdettu mainos, niin kirjavalta kuin se näyttääkin, saattaa jäädä kokonaan ymmärtämättä tai näkemättä.

Kun paneudutaan tarkemmin visuaalisen suunnittelun periaatteisiin, voidaan huomata, että lähestymistapoja ja suunnittelun eri vaiheita on monia. Roger C. Parker (1998, s.1) toteaa, että eräs graafisen suunnittelun haasteista on,

ettei sillä ole vakiosääntöjä. Kaikki on suhteellista, eikä suunnittelun sääntöjä voi typistää ” jos... niin...” – lausekkeisiin. Tekniikat ja työkalut, joita voi käyttää menestyksekkäästi yhdessä tilanteessa, eivät välttämättä toimi yhtä hyvin jossakin toisessa tapauksessa. Parker (1998, s.1) jatkaa vielä myöhemmässä kappaleessa suunnittelun säännöistä. Jos suunnittelu voitaisiin tiivistää tarkasti määritetyiksi säännöiksi, graafiset suunnittelijat voitaisiin korvata tietokoneohjelmilla, ja kaikki mainokset, kirjat, esitteet, tiedotteet ja julisteet näyttäisivät jotakuinkin samoilta. On siis hyvä, että ihmiset ovat erilaisia, ja että erilaiset ihmiset suunnittelevat erinäköisiä mainoksia.

2.2. Visuaalisen suunnittelun lähtökohtia

Edellä todettiin, että erilaisuus, luovuus ja omat näkemykset ovat graafisen suunnittelun rikkauksia. Visuaalinen suunnittelu vaatii lisäksi joukon muita elementtejä, joiden avulla voidaan näyttävään lopputulokseen vaikuttaa.

Parker (1998) listaa seuraavia hyvän suunnittelun taustoja:

- ole valmis kokeilemaan eri vaihtoehtoja,
- luota omiin näkemyksiisi,
- huomaa, että hyvä suunnittelu on jatkuva prosessi eikä yksittäinen tapahtuma,
- omistaudu yksityiskohdille.

Toikkanen (2003) puolestaan antaa seuraavia eväitä julkaisun valmistamiseen:

- Viestintään liittyviä asioita kannattaa pohtia myös laajempina kokonaisuuksina. Esimerkiksi työpaikan graafinen ohjeisto auttaa suuresti töiden suunnittelussa ja toteutuksessa ja luo hyvän perustan viestinnän työtehtävissä toimivien henkilöiden pyrkimyksille. Jollakin osa-alueella nähty vaiva kantaa yleensä hedelmää myöhemmin jossain muualla.
- Hyvä viestintä kertoo asioista lukijaa tukien, auttaen ja kunnioittaen. Suhdetoimintamielessä hyvällä viestinnällä on ratkaiseva vaikutus suuren yleisön mielikuvaan yrityksen/yhteisön toimintatapakulttuurista.
- Graafisen suunnittelun tarkoituksena on antaa viestille sellainen ulkoasu, että lukijan on miellyttävää ja helppoa löytää viestistä oleellisin asia.
- Töiden valmistamiseen tarvittavat kunnolliset tietokoneet ja lisälaitteet, oikeat tietokoneohjelmat ja käyttökoulutus antavat työntekijöille mahdollisuuden toteuttaa suunnitelmat ja ideat.

Kuten listoista voi huomata, tarvitaan luovien ajatuksien lisäksi myös kunnolliset työkalut suunnittelun tueksi niin tietokoneelle kuin suunnittelutyöpöydälle.

Seuraavaksi käyn lyhyesti läpi suunnittelun eri työvaiheita mukailen Parke-
rin (1998) suunnittelun pääperiaatteita.

Konteksti

Ensimmäiseksi täytyy ymmärtää, mitä ollaan tekemässä. Pelkkä pintapuolinen tarkastelu ei kerro vielä kaikkea. Asiayhteyden selvittäminen on elintärkeää. On otettava huomioon myös se, että suunnitteluprosessi on saattanut pyöriä pitkään jonkun päässä ennen kuin se on saavuttanut sinut. Pyri tarkkaan selvittämään, mitä asiakas haluaa. Jos olet epävarma, kysy. Selvitä, kuinka vapaat kädet on suunnitteluun annettu.

Kohderyhmä

Asiayhteyden selvittyä, on joukko suunnittelun eri vaihtoehtoja jo karsiutunut automaattisesti pois. Jos suunnittelet viralliselle taholle julkaisua, täytyy ulko-
asun näyttää jämpiltä ja johdonmukaiselta. Jos taas suunnittelet nuorille kohdistettua julkaisua, saa ulkoasu olla rento ja räväkkä. Kiinnitä joka tapauksessa huomiota selkeyteen, järjestelmällisyyteen ja yksinkertaisuuteen. Ne luovat perustan näyttävälle julkaisulle.

Hahmotelmat

Ennen tietokoneelle siirtymistä on hyvä hahmotella lopputulosta ensin paperil-
le. Vapaalla kädellä piirtäminen käy paljon helpommin kuin hiirellä tuhraami-
nen. Tarkoitus ei ole vielä tehdä lopullista jälkeä, vaan viivat ja tekstit saavat
olla suurpiirteisiä. Tässä vaiheessa on helppo kokeilla eri vaihtoehtoja. Jos si-
nulla on mahdollisuus näyttää asiakkaalle raakileita, ei se ainakaan heikennä
lopullista tuotosta.

Hahmolait

Visuaalisuuteen vaikuttavia tekijöitä ovat mm. tekstien ja kuvien väliset suhteet
yhdessä ja erikseen, liikkeen tai ajatuksen suunta julkaisussa, kontrastin käyttö,
eri visuaalisten komponenttien yhteenkuuluvuus, johdonmukaisuus esitteessä
alusta loppuun ja tarkkaan harkittujen yksityiskohtien korostaminen. Muuta-
malla suunnittelun kurssilla olen törmännyt vastaaviin tekijöihin. Itse tunnen
mainitut tekijät paremmin nimellä hahmolait. Hahmolait ovat mielestäni yksi
tämän tutkimuksen suola. Hahmolakeja tarkastellaan tässä tutkimuksessa tar-
kemmin käytettävyyden rinnalla.

Analysointi

Omien ja muiden tuotosten analysointi auttaa tekemään parempia julkaisuja. Hyvien ja huonojen esimerkkien tutkiminen auttaa sinua välttämään suden kuoppia ja toisaalta inspiroi sinua muovaamaan omia tulevia mainoksia parempaan suuntaan. Maailma on täynnä käyttökelpoisia ideoita, joita oikein soveltamalla saa aikaan vetoavaa jälkeä.

Palaute

Palautteen antaminen ja toisaalta vastaanotto on monille vaikeaa. On kuitenkin hyvä pyrkiä aina kysymään palautetta, jos sellaista ei ole tuotoksesta annettu. Palaute ei aina ole rakentavaa, mutta pitää pyrkiä löytämään huonostakin palautteesta se suola, joka auttaa seuraavalla kerralla suunnittelemaan ja toteuttamaan entistä laadukkaampaa tavaraa.

3. Visuaalisen suunnittelun hyödyt käytettävyyden tutkimiselle

Tässä luvussa kerron, mitä visuaalinen suunnittelu antaa käytettävyydelle. Käyn lyhyesti käytettävyyden määritelmän läpi, jonka jälkeen ryhdyn etsimään yhtäläisyyksiä molempien toimintatavoista. Tarkastelen myös, millä tavalla visuaalisen suunnittelun suunnittelu- ja toteutusperiaatteet ovat muuntuneet käytettävyyden tutkimiseen.

3.1. Käytettävyyden määritelmä

Käytettävyyden voi määritellä muutamalla eri tavalla, mutta tässä on esitelty kaksi usein käytettyä määritelmää. Sinkkonen ja muut (2002, s.19) määrittelevät, että käytettävyys on menetelmä- ja teoriakenttä, jonka kautta käyttäjän ja laitteen yhteistoimintaa pyritään saamaan tehokkaammaksi ja käyttäjän kannalta miellyttävämmäksi. Käytettävyys käyttää hyväksi kognitiivisen psykologian sekä ihmisen ja koneen vuorovaikutuksen tutkimusta. Käytettävyyden isä Jakob Nielsen (2000, ss.10-12) määrittelee käytettävyyden osaksi tuotteen käyttökelpoisuutta. Hänen mukaansa hyvän käytettävyyden muodostavat käyttötilanteen opittavuus, virheettömyys, muistettavuus, tehokkuus ja miellyttävyys.

Esimerkiksi Internetissä käytettävyys on tänä päivänä ase muita kilpailijoita vastaan. Nopeat laajakaistayhteydet ja verkossa toimivat lukuisat saman alan yritykset luovat tiukan kilpailutilanteen, joista harva selviää voittajana.

Käytettävyys on etusijalla vaikka internetsivustolla ei yritettäisikään myydä mitään. Uskolliset käyttäjät ovat Internetin ainoa pysyvä arvo, eivätkä käyttäjät palaa sivustoille, joita on liian vaikea käyttää. Käyttäjät – ja kassavirrat – palaa-

vat sivustolle vain, jos sivustosta jää ensivisiitin jälkeen mieleen positiivinen mielikuva. (Nielsen 2000, s. 389)

Samat argumentit sopivat myös printtimedian suunnitteluun. Visuaalinen suunnittelu tarvitsee käytettävyyden lisäksi mainontaan liittyvää vetovoimaa, joilla ostajat saadaan toimimaan.

Kuten voidaan huomata, nämä käytettävyyden määritelmät viittaavat jollain tasolla aina erilaisiin koneisiin. Puhuttaessa pelkästään visuaalisesta julkaisusta, ei välttämättä käytössä ole konetta, vaan julkaisu on tehty pelkästään paperiversioksi. Kotisivut puolestaan ovat käyttökelpoisia esimerkkejä visuaalisen suunnittelun ja käytettävyyden keskinäisestä vuorovaikutuksesta tietokoneiden ja mobiililaitteiden maailmassa.

3.2. Visuaalisen suunnittelun ja käytettävyyden yhtäläisyyksiä

Käytettävyyttä tutkittaessa tarkastellaan tuotteen vuorovaikutusta ihmisen kanssa, ymmärrettävyyttä, loogisuutta, ulkoasua, virhetilanteiden käsittelyä, luotettavuutta, palautteen antoa ja ohjeistusta. Moni edellä mainittu tutkintakohde sisältää tai on yhteydessä visuaalisuuteen. Oli kyseessä sitten käyttöliitymä tai kahviautomaatti, videonauhurin ohjekirja tai ohjainpaneeli, sisältää tuote jonkinlaisen visuaalisen ilmeen.

Visuaalisuus on käytettävyydessä yksi merkittävimmistä tarkastelun kohteista. Ihminen joko näkee, kuulee tai tuntee, miten tuote reagoi hänen tekemiinsä. Sinkkonen ja muut (2002, s.77) toteavat, että pystyäkseen käyttämään tuotetta, käyttäjän pitää pystyä havaitsemaan tuotteessa kaikki tehtävän suorittamisen kannalta oleellinen. Hänen pitää pystyä seuraamaan myös toimenpiteidensä vaikutusta tuotteen tilaan. Useimmiten palaute, jonka ihminen saa, on visuaalinen. Näin ollen suurin osa visuaalisen suunnittelun säännöistä pätee käytettävyyteen.

Molempien suunnitteluissa lähtökohdat ovat paljolti yleismaailmalliset, jotka pätevät lähes suunnitteluun kuin suunnitteluun. Eroavaisuudet löytyvät käytettävyyden tarkemmasta toiminnallisuuden ja vuorovaikutuksen tutkimisesta ja visuaalisen suunnittelun (mainoksen)idean toimivuuden tarkastelusta.

Yhtäläisyys löytyy taas kaupalliselta puolelta. Molempien takana toimintaa pyörittävät raha ja markkinat. Molemmissa tapauksissa suunnittelulla ja toteutuksella pyritään siihen, että tuotteella olisi hyvät markkinat. Visuaalisella puolella pyritään herättämään ihmisten mielenkiinto. Käytettävyyden puolella pyritään parantamaan tuotteen toimivuutta ja yksinkertaistaa sitä niin, että ihminen ymmärtäisi paremmin laitetta ja toispäin.

Ehkä merkittävimpanä konkreettisena yhtäläisyytenä mainittakoon hahmolait, joita tästä syystä tarkastellaan tarkemmin seuraavassa kohdassa.

3.3. Hahmolait

Kuten aiemmin on todettu, paperille päätyvään julkaisuun ei voida kovin helposti liittää interaktiivisuutta. Tästä johtuen käytettävyyden tutkiminen rajautuu pääosin ulkoasun ja taiton sekä fyysisen koon ja muodon parantamiseen. Fyysinen koko ja muoto ovat luku sinänsä, mutta ulkoasun ja taiton parantamiseksi voidaan ottaa käyttöön hahmolait.

Nielsenin (1993) heuristiikat pohjautuvat visuaalisuuden osalta pitkälti hahmolakien antamiin ohjeisiin. Hahmolakien sanotaan silti olevan vain osa heuristiikkoja. Suunnittelua ei kannata perustaa pelkästään niiden varaan. Ne käyvät mainiosti julkaisujen suunnittelun tueksi tasapainottamaan lopputulosta ja selittämään suunnittelijoille ihmisen vaistomaista havaitsemista.

Sinkkosen ja muiden (2002) mukaan ihmisen havaintojärjestelmä ryhmittelee yksittäiset ärsykkeet isommiksi kokonaisuuksiksi. Kun valonsäteet kohtaavat silmän, mikään säteet poimineista hermoimpulsseista ei ole fyysisesti liitetty mihinkään toiseen hermoimpulssiin, vaan yhdistely tapahtuu aivoissa. Havait-sijan odotukset sekä kohteen tuttuus vaikuttavat siihen, kuinka nopeasti kohde pystytään tunnistamaan ja luokittelemaan. Yksityiskohtien täytyy myös pysyä yhdessä. Kaikista mahdollisista ärsykeyksityiskohtien yhdistelmistä hahmote-taan yleensä yksinkertaisin tai tutuin. Erillään olevat kuva-alkiot liittyvät yh-teen ja ne havaitaan saman kohteen osina tai ominaisuuksiksi. Hahmolait ku-vaavat näitä ihmisten ilmeisesti pitkälti synnynnäisiä piirteiden yhdistelytapo-ja.

Hahmolakeja ovat Sinkkosen ja muiden (2002) mukaan läheisyys, samanlai-suus, jatkuvuus, tuttuus, valiomuotoisuus, yhteinen liike, yhteenliittyminen ja sulkeutuvuus. Rähän ja Ovaskan (2000) mukaan läheisyyden, samanlaisuuden, jatkuvuuden, sulkeutuvuuden ja symmetrisyyden lisäksi on olemassa alueen laki. Seuraavassa luettelen tiiviisti hahmolait selityksineen. Kuvaukset ovat Sinkkosen ja muiden (2002) sekä Rähän ja Ovaskan (2000) mukaisia.

Läheisyys (proximity)

Kaksi visuaalista ärsykettä, jotka sijaitsevat lähellä toisiaan, mielletään yhteen-kuuluviksi. Kohteet, jotka ovat lähellä toisiaan, nähdään ryhmänä.

Samanlaisuus (similarity)

Kaksi samanlaista visuaalista ärsykettä mielletään yhteen tai samaan ryhmään kuuluviksi.

Jatkuvuus (continuity)

Jos viivat leikkaavat toisiaan, katsoja jakaa kokonaisuuden selkeästi jatkuviin osiin. Yhtenäinen viiva koetaan kuvioksi. Kuvio ja tausta pyritään näkemään tavalla, joka aiheuttaa kuvan käyriin tai viivoihin vähiten äkkinäisiä muutoksia.

Tuttuus (familiarity)

Tutut ja merkitykselliset alueet nähdään kuviona.

Valiomuotoisuus (good shape)

Ymmärrämme kuviot mahdollisimman yksinkertaisina, "hyvämuotoisina".

Yhteinen liike (common fate)

Kohteet, jotka liikkuvat samaan suuntaan samalla nopeudella kuuluvat yhteen ryhmään tai kohteeseen.

Yhteenliittyminen (connectedness, connectness)

Kohteet, jotka liittyvät toisiinsa eli ovat toisissaan kiinni, kuuluvat yhteen ryhmään tai kohteeseen. Tämä on ainoa havaintolaeista, jonka voi sanoa "voittavan" muut lait eli tämä on yleensä vahvempi kuin muut lait.

Sulkeutuvuus (closure)

Jos visuaaliset ärsykkeet näyttävät ikään kuin sulkevan sisäänsä jonkin alueen, katsoja näkee sen alueena ja ärsykkeet siihen kuuluvina rajoina, toisin sanoen yhteenkuuluviksi.

Alue (area)

Laajempi (ympäröivä) alueista mielletään taustaksi, pienempi taustasta erilliseksi kohteeksi.

Hahmolait ovat siis vahvasti molemmissa tapauksissa käytössä. Jos lakeja pohditaan maalaisjärjellä, voidaan huomata niiden olevan yksinkertaisia sääntöjä luoda ymmärrettävää ja selkeää tuotosta, mitä hyvänsä onkin tekemässä. Samoja lakeja voidaan käyttää esimerkiksi sisustussuunnittelussa tai työpisteen järjestystä mietittäessä.

3.4. Miten visuaalisen suunnittelun lähtökohdat ovat muovautuneet käytettävyyden tarpeisiin?

Merkittävä ero perinteisen visuaalisen suunnittelun ja käytettävyyden tutkimisen välillä on toiminnallisuus. Näin ollen visuaalisen suunnittelun säännöt eivät välttämättä aina päde suoraan käytettävyydessä. Paperimainoksen ainoana ”toiminnallisuutena” voidaan pitää houkuttelevaa ulkoasua. Kahviautomaatissa toiminnallisuuden tutkimista tarvitaan, jotta kahvi saadaan valittua halutuilla mausteilla. Jotta käyttäjä tietäisi millaista kahvia hän on tilaamassa, tarvitaan ärsykeitä, joista ihminen saa palautteen. Käyttäjä saa peruspalautteen esimerkiksi syttyvillä tai sammuvilla merkkivaloilla.

Liike ja vaihtuvat signaalit ovat sellaisia, joita suoraan ei ole ohjeistettu visuaalisen suunnittelun puolella. Jotta näitä huomion herättäjiä voidaan käyttää oikein, tarvitaan tietoa ihmisen fysiologisista ja psyykkisistä toiminnoista liittyen liikkuvaan tai muuttuvaan näköaistiärsykkeeseen. Näitä ominaisuuksia on tutkittu paljon käytettävyyttä silmällä pitäen. Ulkomaailman valonlähteet ja fysiikkaaliset kohteet säteilevät tai heijastavat valon eri aallonpituuksia. Kun aallot sattuvat silmän verkkokalvolle, silmän aistinsolut reagoivat ärsykkeeseen lähettämällä signaaleja eteenpäin aivoihin – Silmän verkkokalvolle tulee koko ajan uusia ärsykeitä, mutta vain pieni osa aistituista asioista nousee tietoisuuden tasolle käsiteltäväksi. (Sinkkonen ja muut 2002, s.81)

Koska ihminen ei rekisteröi kaikkea näkemäänsä, on käytettävyyden kannalta tärkeää, että annettu palaute tulee varmasti rekisteröityä aivoissa asti. Näin ollen merkin täytyy olla selkeä ja huomiota herättävä. Punainen merkkivalo kahviautomaatissa sekä mahdollinen merkkiääni edesauttavat ymmärryksen muodostumista. Vaikka visuaalisen suunnittelun periaatteista ei suoraan löydy vastausta yllä olevaan, voidaan silti todeta, että punaisen värin käyttö huomion herättäjänä ja merkkinä vaarasta tai virheestä on yksi värien käytön perusolettamuksista.

Äänet, jotka edellä mainitsin, ovat toinen kategoria, joihin ei vastausta visuaaliselta puolelta löydy. Kuten näköaistin, myös kuuloaistin kanssa täytyy perehtyä ensin ihmisen kuuloaistin ja aivojen toimintaan, jotta voidaan toteuttaa ihmisystävällisiä ratkaisuja. Kuuloaistin rajallisuuden voi jokainen kokea hetkellä, jolloin yrittää kuunnella ruokapöydässä kahta eri keskustelua. Toinen hyvä esimerkki on puhua puhelimesta ja katsoa televisiota samaan aikaan. Televisiota katseleva henkilö ei välttämättä huomaa ongelmaa, mutta linjan toisessa päässä ongelma tiedostetaan lähes poikkeuksetta.

4. Käytettävyys visuaalisten materiaalien suunnittelussa

Vastavuoroisesti tässä luvussa kerron miten ja missä muodossa käytettävyyttä on tähän mennessä ilmennyt visuaalisessa suunnittelussa sekä miten ja missä visuaaliset suunnittelijat voisivat ottaa käytettävyyden vielä paremmin huomioon. Haen seikkoja mahdollisimman monesta eri visuaalisesta tuotteesta. Lisäksi pohdin, onko visuaalisesti näyttävä sama asia kuin käytettävä.

4.1. Käytettävyysnäkökohtia visuaalisessa suunnittelussa

Puhuttaessa paperijulkaisuista voidaan unohtaa interaktiivisuus käyttäjän ja tuotteen välillä. Mainoslehtisen tarkoitus on informoida kohderyhmää. Paperimainoksen tarkoitus on saada haluttu kohderyhmä kiinnostumaan ja tätä kautta saamaan lisää tunnettuutta ja ostajia mainostetulle tuotteelle.

Käytettävyyttä sen sijaan voidaan löytää paperimainoksesta mm. materiaalin toimivuuden, fyysisen koon tai värien käytöstä. Esimerkiksi jos taskuun jaettavaksi tarkoitettu mainos on A4-kokoinen, se ei ehjänä mahdu taskuun. Jos uimastadionilla jaettavat lehtiset ovat laminoimattomia, paperi menee käyttökelvottomaksi joutuessaan kosketuksiin veden kanssa tai jos iloisen ja reippaan kansanmusiikkitapahtuman mainos on painettu mustalle pohjalle, ei tapahtuma siitä tietämättömälle tunnu kovin iloiselta.

Käytettävyyttä graafisessa suunnittelussa voidaan tarkastella sekä valmistusvaiheessa että valmiin tuotoksen toimivuudessa. Käytännössä suunnittelijat pyrkivät suunnitteluvaiheessa muuntautumaan kohderyhmän asemaan ja sitä kautta toteuttamaan julkaisu toimivana juuri tälle ryhmälle. Julkaisun valmistuttua nähdään, onko työssä onnistuttu. Prosessia voidaan hyvin verrata ohjelmistosuunnitteluun. Käytössä on tietyt työkalut ja oma tiimi sekä asiakkaalta saatu tieto siitä, mitä pitäisi tehdä. Lopputulosta lähdetään rakentamaan tiettyjen suunnitteluperiaatteiden mukaan iteraatio iteraatiolta ja toivotaan, että lopputulos olisi haluttu ja toimiva.

Käytettävyys paperisissa versioissa viittaa siis luonnollisesti kaikkeen muuhun kuin interaktiivisuuteen. Suunnittelulähtökohtia, joissa käytettävyyttä voidaan konkreettisesti miettiä julkaisussa, ovat mm.

- muoto,
- värit,
- kuviot,
- koko,
- materiaali,
- jakelu,
- mielikuvatavoite,
- säilytettävyys/kertatuote ja

- houkuttelevuus.

Ensimmäiset kolme näkökohtaa viittaavat hahmolakeihin. Loput viisi viittaavat käytännön ratkaisuihin.

Schriver (1997, ss. 211, 445) antaa mainion esimerkin visuaalisen suunnittelun ja käytettävyyden suhteesta. Esimerkki liittyy suureen japanilaiseen viihdeelektroniikka yritykseen, joka oli saanut paljon palautetta tuotteittensa vaikeasta käytettävyydestä. Syy ei niinkään johtunut tuotteen epäloogisuudesta vaan ohjekirjan puutteellisista ja epäselvistä ohjeista. Ongelman todettiin käyttäjien palautteiden ja heurististen arviointien perusteella ja ratkaistiin käytettävyytestien avulla. Tämä esimerkki kertoo hyvin siitä, miten eri tavalla peruskäyttäjä ja laitteen suunnittelijat näkevät asiat. Suunnittelija on sokea omalle työlleen, jolloin ohjeista saattaa jäädä oleellista pois, jota suunnittelija on pitänyt itseltään selvyytenä.

Edellä mainittu esimerkki kuvastaa hyvin, kuinka ongelma ei johtunut laitteen huonosta käytettävyydestä, vaan ohjeista, joilla laitetta piti osata käyttää. Yksinkertaisin laitekin tarvitsee visuaalisesti ja sisällöllisesti selkeät ja yksinkertaiset ohjeet, joilla tyhminkin osaa sitä ongelmitta käyttää.

Kotisivuilla pätee lähes kaikki käytettävyyden normit. Vuorovaikutusta on lähes poikkeuksetta jokaisella sivulla. Jokainen käyttäjä luo sivustosta oman tarinansa käydessään sivuja läpi valitsemassaan järjestyksessä. Kotisivujen käytettävyyden näkökohtiin löytyy laajalti kirjallisuutta. Käytetyin kirja lienee Jakob Nielsenin kirjoittama WWW-suunnittelu, jossa Nielsenin käytettävyys heuristiikat tulevat käytyä läpi.

Nielsenin (2000, s. 9) sanoo, että webissä tärkeintä on käytettävyys. Tämä tarkoittaa yksinkertaisesti sitä, että jos käyttäjä ei löydä jotakin tuotetta, hän ei myöskään osta sitä. Webissä asiakas on kuningas, hiiri aseenaan hän päättää kaikesta. Jos palvelu ei tyydytä, asiakkaan on helppo mennä muualle, koska myös kilpailijat ovat vain hiiren liikautuksen päässä.

4.2. Käytettävyyden ja näyttävyyden erot

Visuaaliset suunnittelijat harvoin puhuvat käytettävyydestä, kun he suunnittelevat paperimainosta. Tällöin he puhuvat mainoksen näyttävyydestä ja toimivuudesta. Mielestäni nuo kaksi ominaisuutta kuvaavat hyvin myös käytettävyyden termiä. Tässä alaluvussa olen kuitenkin tehnyt hieman pesäeroa käytettävyyden ja näyttävyyden sekä käytettävyyden ja toimivuuden välille.

Miten käytettävyys loppujen lopuksi eroaa mainoksista näyttävyydestä puhuttaessa? Käytettävä mainos on tietenkin selkeä, hyvin jäsenneily ja looginen. Samat argumentit pätee myös näyttävään mainokseen. Vielä emme silti tiedä, toimiiko mainos. Nopean johtopäätöksen voi silti tehdä jo edellisistä lauseista.

Molemmissa lauseissa pätee, että mainos on selkeä ja johdonmukainen. Tässä oli jo eväät hyvään julkaisuun, joka tulee ainakin ymmärrettyä.

Parker (1998, s. 23) täsmentää, että tehokas graafinen suunnittelu perustuu hyvään järjestykseen, joka ohjaa lukijan silmää ja selventää ensi- ja toissijaisten seikkojen suhteet julkaisussa. Monilta osin käytettävyyden ja hyvän graafisen suunnittelun periaatteet menevät päällekkäin.

Asia, johon käytettävyys ei pysty mainoksissa vaikuttamaan, on ihmisen mielenkiinnon herättäminen. Vaikka mainos olisi kaikkien käytettävyyssääntöjen mukaan tehty, ei siinä välttämättä ole vielä sellaisia elementtejä, joista ihmiset kiinnostuvat. Tähän ongelmaan tarvitaan mainonnan asiantuntijoiden tietämystä ihmisten käyttäytymisestä mainosten suhteen.

Jokaisessa mielenkiintoa herättäneessä julkaisussa on aina jokin selkeä keino, jolla ihmisten mielenkiinto on herätetty. Voidaan ajatella, että julkaisu on täytynyt kokonaisuudessaan olla selkeä ja johdonmukainen (~käytettävä), jotta se on tullut ymmärretyksi. Lisäksi esimerkiksi huumoria hyväksikäyttäen on julkaisusta saatu hilpeyttä ja keskustelua herättävä.

Mielestäni paperijulkaisu voi olla sekä käytettävä että näyttävä samaan aikaan, koska molemmissa vaaditut kriteerit ovat hyvin lähellä toisiaan. Mutta julkaisu voi mielestäni yksin olla näyttävä ja sitä kautta yleisöön vetoava ilman, että se on käytettävä. Varsinkin sellaisissa julkaisuissa, joissa interaktiivisuutta ei ole ollenkaan ja se sisältää hyvin vähän käytettävyyteen liittyviä seikkoja. Esimerkiksi nuorille suunnattu banderolli-mainos, jossa on sekavalla fonttityylillä kirjoitettu www-osoite (ks. kuva 1). Kirjasin tyyli voi olla jostain tietystä televisioformaattista, jolloin sen tunnistaa vain ne, jotka ovat jo ennestään tuttuja kyseisen kirjasin tyylin kanssa. Mainos on monille täyttä hepreaa, mutta silti se toimii.



Kuva 1. Metropoliz.net -logo 10.5.2007 (www.metropoliz.net)

Nuorille tai lapsille suunnatut kotisivut voivat olla hyvinkin epäkäytettäviä, mutta silti toimivia (ks. kuva 2). Nuoret eivät välttämättä välitä, jos esimerkiksi jokin linkki ei aukea loogisesti, tai jos joku teksti avautuu väärään paikkaan. He todennäköisesti klikkaavat kaikkea mahdollista sivuilla ja kun mielenkiinto

loppuu, siirrytään seuraavalla sivustolle. Nuori saa kulutettua haluamaansa aikaa sivustolla ja on tyytyväinen. Mahdollisesti hän palaa sivuille myöhemmin katsomaan, mitä uutta sinne on ilmestynyt. Myös sivuston sponsorit ovat tyytyväisiä, sillä sponsorien bannerit vilkkuivat koko sen ajan ruudulla, kun käyttäjä selasi sivustoa.



Kuva 2. Coca-Colan Suomen internetsivujen etusivu. 10.5.2007
(www.coke.fi)

5. Yhteenveto

Käytettävyyden periaatteet perustuvat pitkälti visuaalisen suunnittelun perusolettamuksiin selkeästä ja toimivasta julkaisusta. Käytettävyydessä tutkiminen viedään pidemmälle toiminnallisissa seikoissa, kuten loogisuudessa, ymmärrettävyydessä ja vuorovaikutuksessa.

Käytettävyyden suurelta osalta visuaalisuuteen pohjautuva rakenne edesauttaa käytettävyyden tutkimista visuaalisen suunnittelun pääperiaatteita noudattaen. Hahmolait ovat yksi merkittävä kokonaisuus, joita molemmissa tapauksissa käytetään.

Visuaalinen julkaisu voi mielestäni olla näyttävä ilman, että se on käytettävä. Päinvastoin julkaisu voi olla käytettävä, mutta siitä puuttuu visuaalinen veto voima. Visuaalinen suunnittelu ei välttämättä vaadi käytettävyyden tuntemista, vaikka se antaakin julkaisuun monesti lisäarvoa.

Liikkeen ja äänien osalta visuaalisen suunnittelun periaatteet eivät anna vedenpitäviä vastauksia. Näissä tapauksissa vastaukset löytyvät käytettävyyden puolen periaatteista. Kuten visuaalisessa suunnittelussa, myös käytettävyydessä täytyy tuntea ihmisen luonnollinen käyttäytyminen, jotta voidaan aktivoida ihminen toimimaan halutulla tavalla.

Molemmat täydentävät toisiaan. Visuaaliset suunnittelijat voivat hyötyä käytettävyyden näkökohdista. Vastavuoroisesti käytettävyyssiantuntijoiden on mahdollista saada vinkkejä visuaalisen suunnittelun puolelta käytettävyysohjelmiin. Lisäksi käytettävyyssiantuntijat voisivat oppia enemmän ihmisten markkinakäyttäytymistä visuaalisilta suunnittelijoilta. He omaavat taidon luoda mielenkiintoisia ja huomiota herättäviä tuotoksia. Tästä ominaisuudesta olisi hyötyä käytettävyyden puolellakin. Voidaan siis todeta visuaalisen suunnittelun ja käytettävyyden kulkevan käsi kädessä.

Viiteluettelo

- Nielsen, Jakob. (1993). Usability Engineering. Academic Press.
- Nielsen, Jakob. (2000). WWW-suunnittelu. IT-Press.
- Parker, Roger C. (1998). Hyvältä näyttää. suomen Atk-kustannus.
- Räihä, Kari-Jouko., Ovaska, Saila. (2000). Ihmisen ominaisuudet ja niistä johdetut suunnitteluperiaatteet. Kurssin Ihmisen ja tietokoneen vuorovaikutus luennon Ihmisen ominaisuudet luku 3. Saatavilla HTML-muodossa <http://www.cs.uta.fi/~ov/itv/luennot/kalvot/ihminen/index.html>, Tampereen yliopisto, tietojenkäsittelytieteiden laitos, syyskuu 2000. (Luettu 10.5.2007)
- Schriver, Karen A. (1997). Dynamics in Document Design. John Wiley & Sons.
- Sinkkonen, Irmeli., Kuoppala, Hannu., Parkkinen, Jarmo., Vastamäki, Raino. (2002). Käytettävyyden psykologia. Edita Publishing Oy.
- Toikkanen, Rita. (2003). Tyylikäs julkaisu: painotyön ja verkkosivujen suunnittelu ja toteutus. Edita Publishing Oy.

Yliopisto-opiskelijoiden opiskelutaidon kehittäminen verkko-oppimisympäristöjen avulla

Johanna Hiden

Tiivistelmä

Tutkielma koskee opiskelutaitojen tietoverkkoperustaista opettamista yliopisto-opiskelijoille. Työssä tarkastellaan opiskelutaitojen oppimisen pedagogisia lähtökohtia, verkkoympäristöjen soveltumista opiskelutaidon opetukseen sekä esitellään ja arvioidaan kolmea olemassa olevaa opetusympäristöä.

Avainsanat ja -sanonnat: Verkkopedagogiikka, verkko-opetus, opiskelutaito.

CR-luokat: K3.1, K3.2

1. Johdanto

Tutkielmassa käsitellään opiskelutaitojen opettamista verkossa. Opiskelutaitojen hallinta on nyky-yhteiskunnassa keskeisessä asemassa, koska uusien tietojen ja taitojen omaksumista ei enää edellytetä pelkästään opiskeluaikana, vaan myös jatkuvasti muuttuvilla työmarkkinoilla. Verkko-oppimisympäristöjen kehittyminen on tuonut opetukseen uusia mahdollisuuksia, joita voidaan soveltaa substanssiaineiden lisäksi myös erilaisten metataitojen kuten opiskelutekniikoiden opetukseen. Tutkielman tarkemman tarkastelun kohteeksi on valittu yliopistojen oppimisympäristöt, koska akateemiset opinnot asettavat vaativan luonteensa vuoksi opiskelutaitojen hallinnalle erityisiä haasteita.

Työssä taustoitetaan ensin opiskelutaitojen kasvatustieteellistä pohjaa ja esitellään, miten opiskelutaidot vaikuttavat oppimistuloksiin. Opiskelutaitojen opettamiseksi verkossa on tiedettävä ensin, millaisilla pedagogisilla keinoilla niitä on tarkoituksenmukaista opettaa. Lisäksi tehdään lyhyt katsaus siihen, mitä ovat korkea-asteen koulutuksessa vaadittavat opiskelutaidot, ja millaiset tosiasialliset valmiudet yliopisto-opiskelijoilla on tutkimusten valossa suoriutua opinnoistaan.

Verkko-oppimisympäristöistä käsitellään sellaisia ominaispiirteitä, jotka on syytä ottaa suunnittelussa huomioon niiden rajoittavien tai mahdollistavien vaikutusten vuoksi. Laadukkaan verkkoympäristön tunnusmerkkejä taustoitetaan niin ikään pohjaksi olemassa olevien verkkosovellusten arvioimiselle. Vertailukohdaksi esitellään myös verkko-opetuksen luonnetta ja laajuutta yliopistoissa yleisesti.

Opiskelutaitojen opettamisesta verkossa käydään läpi niitä etuja, joita verk-

ko-opetus voi tuoda perinteiseen opetukseen verrattuna. Toisaalta pohditaan myös haasteita, jotka täytyy huomioida tai ratkaista, mikäli opiskelutaitoja halutaan menestyksekkäästi opettaa verkossa.

Lopuksi tutkimuksessa esitellään kolme erilaista oppimiskäsitykseen ja tekniseen toteutukseen perustuvaa opiskelutaitojen kehittämiseen suunnattua verkkoympäristöä. Vertailevalla arvioinnilla tarkasteltavat esimerkit ovat Teekkarin tehopenaali, IQ Form ja Akateemiset opiskelutaidot -kurssi. Esimerkkien valossa pohditaan myös yleisiä suuntaviivoja oppimisympäristöjen kehittämiseksi.

2. Taustaa oppimaan oppimisesta

Opiskelua ei nykykäsitysten valossa ymmärretä pelkästään koulunkäyntiin tai tutkinnon suorittamiseen rajoittuvana aikajaksona, vaan puhutaan läpi ihmisiän jatkuvasta elinikäisestä oppimisesta. Nopeasti muuttuvan yhteiskunnan haasteet ja opiskelun jatkuminen aikuisiällä ovat nostaneet opiskelutaitojen hallinnan aikaisempaa keskeisempään rooliin (Pantzar, 1986). Ei ole kuitenkaan yksiselitteistä, mitä opiskelutaitoihin sisältyy ja kuinka niitä voidaan parhaiten oppia ja opettaa. Kysymys opiskelijoiden opiskelutaitojen hallinnasta on erityisen keskeinen korkeakouluissa, joissa tarve tehostaa ja nopeuttaa opiskelua on viime vuosina ollut voimakkaasti esillä.

2.1. Opiskelutaito ja -tekniikka

Opiskelutekniikaksi voidaan Pantzarin (1986) mukaan määritellä opetustilanteessa sekä itseopiskelussa tarvittavat taidot, joihin liittyvät esimerkiksi kyky hyödyntää opetusta, lukunopeus, muistiinpanotekniikka ja kokeisiin valmistautuminen.

Opiskelutaito voidaan Pantzarin (1986) jaottelussa käsittää joko suppeasti tai laajasti. Suppea käsitys opiskelutaidosta rajautuu opiskelutekniikan taitamiseen ja tietopuolisen opiskelun vaatimukseen, kun taas laajempi määritelmä sisältää opiskeluasenteet, opiskelun suunnittelun, opiskelutekniikan sekä jatkuvan opiskelun valmiuksien kehittämisen ja ylläpitämisen. Lukiolaisten korkeakouluvalmiuksia koskevassa Välijärven (1997) tutkimuksessa asiantuntijoiden määrittelyt opiskelutaidosta taas ovat kyky loogiseen ajatteluun, järjestelmällisen tiedonhankinnan taidot ja valmius uusien kokonaisuuksien tuottamiseen ja esittämiseen.

Opiskelutaitojen ja -tekniikoiden merkityksestä oppimisprosessissa on kasvatustieteen historiassa ristiriitaisia näkemyksiä. Kriittisemmän näkemyksen mukaan ei ole itsestään selvää, mitkä asiat edistävät oppimista. Gibbs (1981) esittää, että vaihtelevien tutkimustulosten perusteella ei voida osoittaa selkeää yhteyttä tiettyjen opiskelutekniikoiden ja oppimistulosten välillä. Keskeisem-

pää on opiskelijan lähestymistapa asiaan, tekniikoilla itsessään voidaan katsoa olevan melko pieni merkitys. Vaikka opiskelijoilla ei olisi puutteita opiskelutaidoissa, oppiminen ei silti aina ole tehokasta.

Väljärven (1997) mukaan tehokkaan opiskelun edellytyksiä ovat tiedon esittäminen, esittäminen ja tuottaminen. Akateemisessa ympäristössä kiinnostus valittuun opintoalaan nähdään yleensä tärkeimpänä korkeakouluopinnoissa menestymisen edellytyksenä, ja ilman sitä hyväkin tietopohja ja kehittynyt opiskelutekniikka jäävät ilman käyttöä.

Toisaalta voidaan osoittaa, että vaikka opiskelutaidot yksinään eivät ratkaise oppimisen ongelmia, oppimistulokset kuitenkin paranevat opiskelutaitojen kehittymisen myötä. Lisäksi opiskelutaitojen ohjausta ja opetusta saaneet opiskelijat kokeneet saavuttaneensa aiempaa parempia tuloksia. (Pantzar, 1986) Vallitsevien oppimiskäsitysten mukaan opiskelutaidot ja -strategiat ovat merkittävän tärkeitä oppimisprosessille (Virtanen, 2001).

Opiskelutekniikoita opetetaan muun muassa erillisillä kursseilla, integroituna substanssiopetukseen, itseopiskelumateriaaleissa, harjoitteluun perustuvilla menetelmillä ja henkilökohtaisessa ohjauksessa. Opetus voi perustua joko normatiivisiin ohjeisiin ja neuvoihin, opiskelijan omiin lähtökohtiin ja niiden reflektointiin tai molempiin (Gibbs, 1981).

Pantzarin (1986) mukaan opiskelutaitoja voidaan oppia sekä teoriassa että käytännössä, joskin tekeminen on tehokkaampaa kuin kirjojen lukeminen. Taidot opitaan sisäisiä tietoon tai tekemiseen perustuvia malleja käyttämällä. Oppimistyyliä on erilaisia, joten olennaista on kykyä erittelemään oma tyylinsä ja arvioimaan sen tehokkuutta. Gibbsin (1981) mukaan yhtä ainoaa menestymisen tapaa ei ole, vaan eri oppijoille soveltuvat erilaiset tekniikat.

2.2. Korkeakouluopiskelijoiden valmius oppimiseen

Väljärven (1997) mukaan opiskelutaidot ovat olennainen osa korkeakouluvalmiutta yliopisto-opettajien näkökulmasta katsottuna. Samoin ovat asenteet, laadukas tietoperusta, myönteinen asennoituminen sekä ajattelun ja tiedon tuottamisen valmiudet. Opettajat olettavat opiskelutaitojen ilmenevän omaaloitteisena kykynä viedä päätökseen erimuotoisia oppimistehtäviä, minkä lisäksi kriteerit voivat vaihdella tieteenalakohtaisesti.

Oppiminen korkeakouluissa taas ymmärretään pitkälti kriittisen reflektoinnin ja ajattelun taidoiksi, jotka opiskelijan tulisi omaksua. Yliopiston pääasiallisena tehtävänä on edistää oppimisen olosuhteita ja mahdollistaa transformatiivinen oppiminen. (Brockbank & McGill, 1998)

Oppimisympäristö muuttuukin huomattavasti siirryttäessä lukiosta yliopisto-opiskeluun, joka on sisällöllisesti kohdennetumpaa ja korostaa itseohjautuvuutta. Tietoverkkojen ja modernien teknologioiden hallinta on akateemisen

opiskelun kannalta keskeistä, mutta kehittynyt opiskelutekniikka sisältää edelleen myös valmiudet seurata luentoja ja harjoituksia, tehdä muistiinpanoja ja rakentaa kokonaiskuvaa opetuksessa käsiteltävistä teemoista. Etenkin akateemisen lukutaidon merkitys korostuu, koska luento-opetusta korvataan laajoilla kirjatenteillä ja monimuotoisilla kirjallisilla tehtävillä. (Väljjarvi, 1997)

Monelle nuorelle korkeakouluun siirtyminen on kriisien ja ylivoimaiselta tuntuvien vaatimusten aikaa, joka voi usein johtaa opintojen keskeytymiseen. Opiskelutekniset valmiudet kehittyvät ja hioutuvat monesti vasta kokemusten myötä, kun motivoitunut opiskelija vähitellen integroituu korkeakoulun sosiaaliseen yhteisöön. Lukiolaisten valmiudet oppia ovat lähtökohtaisesti hyvin erilaisia. (Väljjarvi, 1997)

Korkeakouluopiskelijat ylipäänsä ovat heterogeeninen joukko niin taustoiltaan kuin taidoiltaan. Gibbssin (1981) mukaan opiskelutekniikoiden opetuksessa tulisikin aina huomioida opiskelijan olemassa olevat tavat ja käsitykset, koska opiskelutekniikat omaksutaan aiemmin opitun kehyksessä.

3. Taustaa verkko-oppimisympäristöistä

Verkko-oppimisympäristöjen käyttö opetuksessa on lisääntynyt hiljalleen. Niiden eduiksi perinteiseen opetukseen verrattuna nähdään usein ajasta ja paikasta riippumattomuus, helppo saavutettavuus ja monipuoliset tavat yhdistää eri medioita. Verkko ja sen mahdollisuudet eivät kuitenkaan yksinään riitä parantamaan opetusta, vaan oppimisympäristöjen tulisi perustua harkitusti valittuihin pedagogisiin menetelmiin ja tavoitteisiin. Oppimisympäristöjä voidaankin suunnitella ja tarkastella erilaisten laatuksiteereiden valossa, joissa on huomioitu verkon erityispiirteet ja opetukselle tuomat haasteet.

Yliopistoissa tieto- ja viestintäteknikan käyttö on toistaiseksi ollut melko maltillista. Verkkokursseja on toteutettu erilaisina pilottihankkeina ja kokeiluinna. Käytetyissä oppimisalustoissa on paljon eroja niin pedagogisen lähestymistavan kuin teknisen toteutuksenkin suhteen. Verkko-opetuksen laatua tarkastelleessa hankkeessa todettiin, että valtiovalta on ohjannut yliopistoja kehittämään verkko-opetusta osaksi tarjontaansa (Nevgi & Heikkilä, 2005). Katsauksen mukaan verkko-opetusta on toteutettu etupäässä avoimessa yliopistossa, kun taas perusopetuksen puolella tarjonta ei ole kovin laajaa. Lisäksi verkko-opetusta annetaan enimmäkseen osana monimuoto-opetusta, vain harvassa kurssissa lähiopetus on korvattu kokonaan verkon kautta opiskelulla.

3.1. Verkko-oppimisympäristöjen tunnuspiirteitä

Verkko-oppimisympäristö voidaan määritellä usealla tavalla. Tässä tutkielmassa käytetään Mannisen (2003) määritelmää, jonka mukaan verkkopohjainen oppimisympäristö on toteutettu käyttäen hyödyksi Internetiä ja verkkoteknologi-

aa, ja se muodostuu usein hypertekstistä ja -mediasta, linkeistä sekä erilaisista vuorovaikutuskanavista.

Verkkokurssi voi olla verkkopohjaisen oppimisympäristön alakäsite. Sillä tarkoitetaan tiettyihin oppimistavoitteisiin perustuvaa rajattua verkkosivustoa. Kurssi on yleensä aikataulutettu ja ohjattu opetussuunnitelmaan perustuva kokonaisuus, jossa on mukana vuorovaikutusmahdollisuuksia sekä opiskelua tukevaa oppimateriaalia (Manninen, 2003). Verkkokurssi voidaan myös käsitellä avoimena tai suljettuna oppimisympäristönä, jossa valtaosa opetuksen ja opiskelun vuorovaikutuksesta toteutuu opettajan ja opiskelijoiden välillä tietoverkkojen välityksellä (Nevgi & Heikkilä, 2005).

Verkko-oppimateriaali puolestaan voi olla tulostettavaa lineaarista tekstiä tai pidemmälle kehitettyä itseopiskelumateriaalia, jossa on oppimista tukevia hypertekstiominaisuuksia tai vastaustautomaattikkaa (Manninen, 2003).

Verkko-oppimisympäristöjä ja -opetusta voidaan arvioida ja suunnitella erilaisten laatukriteereiden mukaan. Laadun arviointi ei ole yksiselitteinen tehtävä, koska opetuksen tasoa yleisesti on vaikea arvioida. Eri toimijoilla on myös erilaisia käsityksiä verkossa tapahtuvasta opetuksesta. Arviointiin on kehitetty erilaisia kansainvälisiä kriteeristöjä, mutta ne ovat ainakin toistaiseksi luonteeltaan enemmän tarkistuslistoja kuin kattavia tieteellisiä mittareita. (Heikkilä *et al.*, 2005)

Nokelainen ja Sointu (2003) ovat listanneet hyvien ohjaavien materiaalien melko konkreettisia tunnusmerkkejä. Heidän mukaansa laadukas materiaali:

- perustuu asetettuihin oppimistavoitteisiin
- muodostaa mielekkään ja yhdenmukaisen kokonaisuuden
- on sisällöltään selkeä, kiinnostava ja helposti ymmärrettävä sekä esitystavaltaan informatiivinen
- hyödyntää verkon hypertekstuaalisuutta
- synnyttää vuorovaikutteisia reflektiota ja dialogia
- luo opiskelijassa motivaatiota, aktiivisuutta ja itseohjautuvuutta oppimiseen ja opiskeluun
- on ulkoasultaan ja tekniikaltaan käyttäjäystävällinen
- sisältää riittävät ohjeet ja opastuksen koko koulutuksen ajan
- hyödyntää verkon monipuolisuutta tarkoituksenmukaisesti
- on sellainen, että opiskelija löytää opiskelunsa kannalta merkittävän tiedon ja hänelle määritellyt tehtävät sujuvasti
- soveltuu erilaisille oppijoille.

Verkko-opetuksen laatua arvioineessa hankkeessa puolestaan nostetaan esille viisi tapaa tarkastella verkko-opetuksen laatua laajemmin (Heikkilä *et al.*, 2005):

- *Poikkeuksellisuuden ja erinomaisuuden* näkökulmasta verkko-opetuksen laatu on yhteydessä sitä antavan yksikön laatuun, jolloin korkealaatui-

sen yksikön verkko-opetuksen voi olettaa niin ikään olevan laadukasta.

- *Tasaisuuden ja virheettömyyden* kautta arvioidessa tarkastellaan opetuksen virheitä ja puutteita, ja pyritään mahdollisimman laadukkaaseen oppimisprosessiin standardeilla.
- *Tarkoituksenmukaisuuden* kautta arvioidaan asiakkaan tarpeiden ja toiveiden toteutumista, jolloin laatu ilmenee suhteessa tarkoituksiin ja tavoitteisiin.
- *Kustannustehokkuuden* pohjalta laatu on käytettyjen resurssien ja saavutettujen tuloksien mahdollisimman hyvä suhde.
- *Transformatiivisessa* tarkastelussa katsotaan laadun syntyvän muutosprosessissa opettajan ja opiskelijan vuorovaikutuksen tuloksena, ja mittarina on opiskelijan asiantuntemuksen kehittyminen.

3.2. Verkko-opetus korkeakouluissa

Yliopisto-opetus on viime vuosikymmenien aikana muuttunut voimakkaasti. Oppimisen ja opetuksen tutkimus on vallannut alaa, tieto- ja viestintätekniset edellytykset ovat parantuneet sekä opiskelijajoukon määrä on kasvanut ja opiskelijakenttä muuttunut heterogeenisemmäksi. Nämä muutokset sekä opintojen vaihduttamista ajava tutkinonuudistusprosessi ovat kasvattaneet tarvetta kehittää opiskelijoille yksilöllisiä ja joustavia opiskelumahdollisuuksia. (Nevgi & Heikkilä, 2005)

Verkko-opiskelun voidaankin nähdä olevan eräänlainen uusi akateemisen vapauden sovellus. Opiskelijat voivat vaihtoehtoisista opiskelumuodoista valita elämäntilanteensa, oppimiskykynsä ja opetussuunnitelmansa mukaisesti parhaat oppimismuodot ja -tavat. Verkko-opetuksen kautta on mahdollista yhdistää yliopisto-opetuksen perinteet uuden teknologian mahdollisuuksiin, huomioida opettajien ja opiskelijoiden erilaisuus sekä saavuttaa hyviä oppimistuloksia. (Tervaportti, 1999)

Ajasta ja paikasta riippumattomuus ei kuitenkaan aina ole itsestään selvästi tavoiteltava suunnitteluperiaate. Monet verkkokurssit on toteutettu yksilöllisinä itseopiskelukursseina, koska on uskottu sen poistavan opiskelun esteitä, tuovan paremmat mahdollisuudet opiskeluun ja parantavan oppimisen tasoa. Verkkokurssien laatua kartoittaneen hankkeen pilottikursseilla on kuitenkin todettu, että opintojen aikatauluttaminen ja yhteiset verkko-opiskelutilanteet edistävät oppimista paremmin kuin mahdollisuus itsenäiseen opiskeluun (Nevgi & Heikkilä, 2005).

Verkko-oppimisympäristöjä käytetään opetuksessa useilla eri tavoilla. Niitä voidaan käyttää esimerkiksi lähiopetuksen rinnalla osana muuta opiskelua, tai opiskella pelkästään verkossa. Opettajalla voi olla kurssilla ohjaava tai passiivinen rooli. Joissakin tapauksissa ympäristö perustuu pelkkään vastaanotetta-

vaan materiaaliin, toisissa siihen on sisällytetty vuorovaikutusta. Materiaalit voivat myös olla luonteeltaan ohjaavia tai ei-ohjaavia. (Manninen, 2003) Erottelu voidaan myös tehdä sen mukaan, onko ympäristö ajallisesti tai osallistujamääriltään rajattu, vai onko kyseessä kaikille avoin oppimisolusta.

Perinteisesti yliopisto-opetus on ollut tiedon pariin ohjaamista, ajattelun ja argumentaatiotaitojen kehittämistä sekä ohjaamista uuden tiedon löytämiseen. Yhtenä tärkeimmistä lähtökohdista on ollut vuorovaikutus opettajan ja opiskelijan välillä. Myös opiskelijoiden välinen vuorovaikutus on tärkeää oppimisprosessin kannalta. Tietoverkkojen myötä yliopistojen opetus on kehittymässä monimuotoiseksi ja yhä erilaistuvaksi. Verkko-opetukselle ominaisia piirteitä ovat muun muassa joustavuus, avoimuus ja teknologian monipuolinen hyödyntäminen. (Nevgi & Heikkilä, 2005)

Verkko-opiskelu vaatii opiskelijoilta Korhosen (2005) mukaan joukon erilaisia taitoja. Tärkeimmiksi nousee, että opiskelijoiden tulisi ymmärtää omia oppimistapojaan ja -käsitteisiään sekä arvioida niitä, suhtautua uusiin oppimislanteisiin tavoitteellisesti, soveltaa sopivia strategioita ja taitoja oppimisessaan sekä reflektoida kokemuksiaan ja toimia niiden pohjalta.

4. Oppimistaitojen verkko-opetus

4.1. Verkon käytön etuja

Opiskelutekniikoissa on seikkoja, joiden vuoksi verkko-oppimisympäristö on erinomainen ympäristö niiden opettamiselle. Verkko tarjoaa paljon sellaista potentiaalia, jota ei vielä ole laajamittaisesti hyödynnetty.

Yksi tällainen seikka on, että opiskelijat tarvitsevat aikaa opiskeluprosessin omaksumiseen. Orientaatiot ja opiskelijan ymmärrys omasta oppimisestaan ovat hitaasti muuttuvia, fundamentaalisia lähestymistapoja, joita on vaikea muuttaa kertaheitolla (Gibbs, 1981). Avoin verkko-oppimisympäristö voi tarjota tukea opiskelijalle koko hänen opintopolkunsa ajan, ja ympäristöstä voi käyttää aina kulloinkin oppijan tilanteeseen soveltuvia osia.

Nykyisten oppimisenäkemyksien valossa oppija onkin aktiivinen toimija, joka valikoi itse, mitä haluaa oppia. Hän konstruoi oppimaansa tarpeidensa, kiinnostustensa ja näkemystensä perusteella. Verkko tarjoaa erinomaisen toimintaympäristön itsenäiselle ja tavoitteelliselle opiskelulle. (Nokelainen & Sointu, 2003) Avoimessa ympäristössä kukaan ei määrää tavoitteita ulkoahjautuvasti, vaan oppijan on kehitettävä omia valmiuksiaan asettaa tavoitteita, määrittellä tehtäviä ja ohjata toimintaansa (Koli & Silander, 2002).

Verkko myös mahdollistaa oppimisprosessin juonen visualisoinnin (Nokelainen & Sointu, 2003). Kirjoittamalla tai visualisoimalla tietoa esimerkiksi käsitteiden muotoon opiskelija voi tulla tietoiseksi omasta oppimisprosessis-

taan. Myös vuorovaikutus on tärkeässä roolissa, koska tarkastelemalla muiden ulkoistettua ajattelua ja ongelmanratkaisua oppija voi kehittää omiakin taitojaan. (Koli & Silander, 2002)

4.2. Suunnittelun haasteita

Opiskelutaidon ja -tekniikoiden opetuksesta on esitetty käsityksiä, jotka on hyvä huomioida myös suunnitellessa niiden verkkopohjaista opetusta.

Neuvoihin ja ohjeisiin perustuva normatiivinen materiaali ei välttämättä ole onnistunut suunnitteluvaihtoehto. Neuvoista ei aina ilmene prosessi, jolla suositeltavaan toimintamalliin on päädytty, ja neuvon vastaanottajan ymmärrys voi siksi jäädä pintapuoliseksi. Ohjeet saattavat myös olla mahdottomia toteuttaa tai epärealistisia oppijan lähtökohtiin nähden. Neuvomisen kohde saattaa myös kokea ehdotukset hyökkäyksenä omia oppimistapoja vastaan, jolloin hän ei muuta käytöstään niiden perusteella. (Gibbs, 1981)

Erilaiset opiskelutaitoa käsittelevät oppaat ja ohjeet puolestaan ovat omiaan muistuttamaan relevanteista oppimiseen liittyvistä tekijöistä ja lisäämään kiinnostuneisuutta oppimisen kysymyksiä kohtaan. Oppaat eivät kuitenkaan auta spesifeissä ongelmissa. Lisäksi ne vaativat itsetuntemusta ja itsekuria, jotta oppija tunnistaa materiaalin joukosta mikä on olennaista ja pystyy omaksumaan ohjeet. Tutkimukset erilaisten opaskirjojen tehosta eivät olekaan antaneet kovin hyviä tuloksia. (Wilson, 1981)

Myös kurssimuotoisella opetuksella ja harjoittelulla on ongelmansa. Lyhyellä aikajaksolla omaksutut asiat eivät välttämättä adaptoidu osaksi arkipäivän rutiineita. Jos uuden taidon käyttäminen on työläämpää kuin vanha tottumus, pitäisi opiskelijan motivoitua harjoittelemaan sitä säännöllisesti. Lisäksi taidoista on hyötyä vain, jos niitä käytetään tarkoituksella ja relevantteihin asioihin. (Gibbs, 1981)

Yliopistojen osalta keskeinen haaste on opiskelijoiden roolin muuttaminen aktiiviseksi oman työnsä suunnittelijoiksi ja toteuttajiksi. Opiskelijoita tulisi kannustaa arvioimaan kriittisesti omaa työtään ja oppimisympäristöään. (Väljäjärvi, 1997)

Yhdeksi opiskelutaitojen osaksi voidaan laskea asenne opiskelua kohtaan. Koska motivaatio on tärkeää menestyksessä opinnoissa, erilaisten tekniikoiden hallitsemisesta ei ilman opiskelumotivaatiota ole hyötyä. Verkko-oppimisympäristöjen laatijoille on mielenkiintoinen haaste pohtia, miten niiden avulla pystytään kohentamaan tai ylläpitämään opiskelumotivaatiota tai vaikuttamaan opiskelijan asenteisiin opiskelua kohtaan.

5. Esimerkkejä opiskelutaidon verkko-oppimisympäristöistä

Suomalaisten yliopistojen verkko-opetuksessa ei opiskelutaitojen opetukseen

ole kiinnitetty systemaattista huomiota. Ilman tarkempaa selvitystä on mahdollista sanoa, onko opiskelutaitojen hallinta jäänyt tahattomasti huomioimatta, vai mielletäänkö opiskelutaidot taustalla oleviksi metataidoiksi eikä opiskeltavaksi asiaksi itsessään.

Olemassa olevat opiskelutaidon kurssit ja oppimisympäristöt on tuotettu erilaisina hankkeina tai opintojen tukipalveluina. Teknillisessä korkeakoulussa, Helsingin yliopistossa ja Oulun yliopistossa on kussakin toteutettu tietoverkkoperustainen opiskelutaitoja edistävä verkkopalvelu hieman eri lähtökohdista käsin. Oulun yliopiston kurssia on sovellettu myös neljässä muussa yliopistossa. Seuraavaksi esitellään tarkemmin kyseiset palvelut.

5.1. Näkökulma esittelyyn ja arviointiin

Kutakin sovellusta tarkastellaan sekä pedagogisten kysymysten että hyvän verkkoympäristön ominaisuuksien näkökulmasta. Sovellusten rakennetta ja toiminnallisuutta arvioidaan siinä laajuudessa, kuin se on tämän työn puitteissa mahdollista. Tarkastelun kriteerit on valittu siten, että arviointi on mahdollista suorittaa sovellusten käyttöliittymään ja toiminnallisuuteen tutustumalla tai niistä laadittujen ohje- ja esittelymateriaalien perusteella. Valitut kriteerit ovat:

- *Kattavuus.* Vastaako sovellus suppeaa vai laajaa käsitystä opiskelutaidosta?
- *Opetusmetodi.* Perustuuko opetus normatiivisiin ohjeisiin vai opiskelijan omista lähtökohdista käsin opettamiselle?
- *Oppimistyylyt.* Otetaanko erilaiset oppimistyylyt sovelluksessa huomioon?
- *Tieteenalaerot.* Kiinnitetäänkö opiskelutaidoissa huomiota tieteenalakohdittaisiin eroihin?
- *Tietopohja.* Rakentuuko opetus opiskelijan aiempien tietojen ja taitojen ja niiden muodostaman kehiksen pohjalle?
- *Opetusmuoto.* Onko oppimisympäristö osa monimuoto-opetusta, johon sisältyy myös lähiovetusjaksoja, vai onko se tarkoitettu pelkästään verkossa suoritettavaksi?
- *Avoimuus.* Onko kyseessä itseopiskeluun perustuva avoin oppimisympäristö vai aikataulutettu ja suljettu kurssi?
- *Vuorovaikutus.* Mahdollistaako sovellus vuorovaikutuksen opettajan tai muiden opiskelijoiden kanssa?
- *Aktiivointi.* Sisältääkö toteutus tekstin lisäksi muita medioita ja interaktiivisia osuuksia?

5.2. Teekkarin tehopenaali

Teekkarin tehopenaali on Teknillisen korkeakoulun opiskelijoille suunnattu verkkosivusto, joka sisältää opiskelua helpottavia ja tehostavia työvälineitä.

Tavoitteena on tukea oppimista helpottamalla rutiinien suorittamista, auttaa opiskelijoita sisäistämään korkeakouluopiskelun periaatteet ja pehmentää siirtymää lukiosta korkeakouluun. (Hartikainen, 2007)

Oppimisympäristö pohjautuu staattisiin www-sivuihin. Sivut on jaettu kategorioihin, jotka keskittyvät pääasiassa opiskelutekniikkaan ja konkreettisiin opiskelurutiineihin. Jokainen kategoria sisältää lyhyen teoriaosuuden ja lopuksi linkkejä aihetta käsitteleviin työkaluihin.

Tehopenaali sisältää sekä normatiivisia ohjeita että oppijakeskeisiä elementtejä. Normatiivisesti ohjataan esimerkiksi muistiinpanojen tekemistä listaamalla, mitä asioita muistiinpanoja tehdessä tulee kirjata ylös ja millä tekniikalla. Oppijakeskeistä lähtökohtaa puolestaan edustaa interaktiivinen oppimistyyli. Valtaosa sivujen sisällöstä on kuitenkin vinkkejä ja ohjeita, joiden soveltuvuus erilaisille oppijoille on kyseenalainen. Opiskelijan aiempia tietoja aiheesta ei huomioida muuten kuin mahdollistamalla linkitetyllä rakenteella se, että kukaan voi edetä sivustolla haluamiensa polkujen mukaan.

Sivusto on tarkoitettu yksittäisten oppijoiden käyttöön, eikä sillä ole mahdollisuutta saada ohjausta tai toimia vuorovaikutuksessa muiden käyttäjien kanssa. Tehopenaalin lähtökohta on itsenäisessä opiskelussa, eikä se yhdisty minkään tieteenalan tai kurssin sisältöihin lukuun ottamatta yleisesti käsiteltyä matematiikan ja fysiikan opiskelua. Avoin ympäristö tukee oppimisprosessin kypsyttelyä ja monimuotoista kehittymistä, mutta oppijan itseohjautuvuuden varaan voi jäädä liikaakin, koska sivut eivät tarjoa juuri minkäänlaista ohjausta, tukea tai edes palautetta oppijalle.

Interaktiivisuutta sivuilla on huomattavan vähän, eikä verkkoympäristön etuja ole hyödynnetty juuri lainkaan. Sivuilla on paljon osuuksia, joihin interaktiivisuus sopisi erinomaisesti, mutta toteutus on tehty staattisena tekstinä ja kuvina. Hyvä esimerkki on SWOT-analyysi, jossa kehoitetaan oppijaa täyttämään nelikenttä, mutta interaktiivisen verkkotehtävän sijasta nelikenttä on esitetty verkkosivuilla kuvatiedostona, jota ei ole mahdollista muokata.

Sisällöt ovat melko hajanaisia, eikä sivustolla kerrota suunnittelutavoitteesta tai lähtökohdista, joille Teekkarin tehopenaali perustuu. Sivut sisältävät materiaalia eri lähteistä ja linkkejä ulkoisille sivuille, minkä vuoksi sivusto muistuttaa enemmän oppimisportaalia kuin varsinaista toiminnallista oppimisympäristöä. Sivut sisältävät kattavan valikoiman tietoa, mutta tekninen toteutus ei tarkastele kohteeksi valittujen kriteereiden valossa erityisesti tue oppimista tai motivoi käyttäjää.

5.3. IQ Form

IQ Form on Helsingin yliopiston kasvatustieteen ja tietojenkäsittelytieteen laitosten yhteistyöprojekti. Projektin lopputuote on verkossa toimiva vuorovai-

kutteinen palautejärjestelmä, IQ Form -ohjelmisto, joka hyödyntää uusia älykkeitä tekniikoita oppimisprosessin tukena. Ohjelmisto muotoutuu dynaamisesti oppijoiden käytöksen mukaan. (*IQ form*)

IQ Learn -järjestelmä on itsenäisesti toimiva ohjelmiston osa. Sen tavoitteena on kannustaa opiskelijaa kehittämään vahvuuksiaan ja opiskelutaitojaan itsearvioinnin pohjalta. Järjestelmä edesauttaa strategisten oppimistaitojen kehittämistä ja lisää opiskelijan itsetuntemusta. IQ Team puolestaan keskittyy oppimisen sosiaaliseen puoleen. (*IQ form*)

Järjestelmä on toteutettu dynaamisena www-sivustona, joka sisältää sekä tekstiosuuksia että interaktiivista toiminnallisuutta. Teemat käsittelevät monipuolisesti yksin ja ryhmässä oppimisen teemoja. Oppijakeskeisyys on koko järjestelmän toiminnan perusta, koska sovellus luo käyttäjistä yksilöllisiä profiileja. Tieteenalakysymyksiin ei merkittävässä määrin oteta kantaa, vaan järjestelmä on tehty yleisluontoiseksi.

Palvelujärjestelmä sisältää interaktiivisia testejä ja päiväkirjan, joiden tiedot jäävät rekisteröityneille käyttäjille talteen. Vuorovaikutusta muiden käyttäjien kanssa tapahtuu suoraan järjestelmän yhteiskäytön myötä, ja epäsuorasti sen kautta kun järjestelmä välittää käyttäjille tietoa muiden tekemisistä.

IQ Form on verkossa vapaasti käytettävissä, ja sen voi sovittaa toimimaan minkä tahansa oppimisalustan rinnalle. Järjestelmä on tarkoitettu käytettäväksi itsenäisillä verkkokursseilla, ja tavoitteena on tuottaa opiskelijoille opiskelun yhteydessä tietoa heidän vahvuuksistaan ja heikkouksistaan opiskelutaitojen suhteen. (*IQ form*)

5.4. Akateemiset opiskelutaidot -verkkokurssi

Akateemiset opiskelutaidot -verkkokurssi on Oulun yliopistossa vuonna 1997 ensimmäistä kertaa järjestetty kurssi, joka on sittemmin poikunut jatkokseen alkuohjaukseen keskittyvän viiden yliopiston yhteisen virtuaaliyliopistohankkeen. Akateemiset opiskelutaidot -kurssi luotiin antamaan opiskelijalle valmiuksia opintojen aloittamiseen ja helpottamaan integroitumista tiedeyhteisöön. (*Akateemiset opiskelutaidot.*)

Kurssi koostuu tekstimuotoisista html-sivuista, jotka sisältävät kurssiin liittyviä ohjeita ja opetussisältöjä. Lisäksi kurssilla on käytössä Discendum Optima -oppimisalusta, jonka avulla on toteutettu kurssin interaktiiviset osuudet. Sisältö jakaantuu kolmeen osaan, joista ensimmäisessä käsitellään akateemisuutta ja tiedeyhteisöä, toisessa opiskelun suunnittelua ja kolmannessa tiedonhaun perusteita. (*Akateemiset opiskelutaidot.*) Teemat ovat laajoja ja pohdiskeluvia, ja niissä tulee hyvin esille opiskelutaitojen koko kirjo. Kurssi sisältää sekä teoriaa että toiminnallisia osuuksia.

Kurssin lähestymistapa on hyvin oppijakeskeinen. Tehtävät sisältävät pal-

jon elementtejä, joissa oppija ohjataan pohtimaan omaa rooliaan ja toimintatapojaan. Tämän kautta erilaiset oppijat ja heidän taustansa tulevat hyvin huomioiduiksi. Keskeinen osa kurssista muodostuu myös oman tieteenalan erityispiirteiden hahmottamisesta.

Kurssiin kuuluu monenlaisia interaktiivisia osuuksia lomakkeiden täyttämistä materiaalien tuottamiseen ja organisoimiseen, ja opiskelijat voivat tietyissä puitteissa valita itselleen parhaiten soveltuvia työskentelymuotoja. Vuorovaikutus sivuilla on toteutettu keskustelualueilla, joita käytetään kurssin kuussa yleiseen keskusteluun sekä ohjattuihin teemakeskusteluihin. Jokaisella kurssille osallistuvalla opiskelijaryhmällä on oma tuutori, joka ohjaa kurssilaisia, antaa palautetta ja toimii taustatukena. Kurssi etenee määrätyissä aikarameissa, joiden sisällä opiskelijat voivat itsenäisesti suunnitella aikataulunsa ja suoritustapansa. Tekstimuotoiset materiaalit ovat vapaasti nähtävillä, mutta oppimisalustalle pääsy vaatii kirjautumisen. Kurssi suoritetaan kokonaan tietoverkkoperustaisesti. (*Akateemiset opiskelutaidot.*)

Akateemiset opiskelutaidot -kurssin tavoitteet on ilmaistu sivuilla selkeästi, ja kurssin tekninen toteutus palvelee valittuja oppimistavoitteita. Sivusto muodostaa yhtenäisen ja johdonmukaisen kokonaisuuden, jonka sisällä opiskelijan on mahdollista muodostaa omia polkujaan ja hakea lisää tietoa häntä kiinnostavista aiheista. Vaikka kurssi suoritetaan tietyn aikajakson sisällä, sen taustalla on kuitenkin ajatus opiskelutaitojen jatkuvasta kehittymisestä, jolle kurssin kuussa tehdyt pohdinnat voivat toimia hyvänä alkuvirikkeenä (Koski, 2000).

Kurssille myönnettiin vuonna 2006 Opetusministeriön ja Opetushallituksen verkko-opetuksen laatupalkinto. Perusteina olivat opiskelijoiden tarpeista lähtenyt kehitystyö, materiaalien avoimuus ja sovellettavuus eri aloille sekä laadun jatkuva kehittäminen palautteiden pohjalta. (Opetusministeriö, 2006)

5.5. Yhteenveto esimerkeistä

Opiskelutekniikoita voidaan opettaa monella eri tavalla. Taulukkoon 1 on koottu yhteenveto edellä esitellyistä oppimisympäristöistä ja niiden ominaisuuksista.

	Teekkarin tehopenaali	IQ Form	Akateemiset opiskelutaidot
Kattavuus	Tekniikat	Tekniikat, itsetuntemus, ryhmätyötaidot	Tekniikat, itsetuntemus, identiteetti
Opetusmetodi	Normatiivinen	Oppijalähtöinen, yksilölliset profiilit	Oppijalähtöinen, poh- dinta ja kes- kustelut
Oppimistyyli	Linkit testeihin	Testit, testitulosten dynaaminen käyttö	Oppimistehtävät
Tieteenalaerot	Sivutaan tekstissä, tehty teekkareille	Sivutaan tekstissä, soveltuu kaikille aloille	Käsitellään, sovel- tuu kaikille aloille
Tietopohja	Ei kerää tietoa	Kerää dynaamisesti tietoa oppijasta	Oppija tuottaa ja järjestelee tietoa itse
Opetusmuoto	Itseopiskelu, vain verkkokurssi	Itsenäinen tai ohjat- tu opiskelu, liitet- tävissä verkkokurs- seihin tai lähiope- tukseen	Ohjattu opiskelu, vain verkkokurssi
Avoimuus	Vapaasti saatavilla	Tekstit saatavilla, oppimisalustalle rekisteröityminen	Tekstit saatavilla, oppimisalusta sul- jettu
Vuorovaikutus	Ei	Ryhmätyöskentely, tietojen välittyminen	Verkkotuutori, keskustelupalstat
Aktivointi	Vähäinen	Testit, päiväkirja, tilastot	Testit, keskustelu, dokumenttien tuot- taminen ja hallinta

Taulukko 1. Yhteenveto oppimisympäristöjen ominaisuuksista.

Oppimisympäristöjen vertailu jättää avoimeksi useita kiinnostavia jatkokysymyksiä. Pinnallisen vertailun perusteella ei ole mahdollista kuin havainnoida oppimisympäristöjen eroja ja vertailla toteutusratkaisuita. Erilaisten menetelmien paremmuuden selvittäminen edellyttäisi jo huomattavasti laajempaa tutkimusta. Tässä tutkielmassa tarkastelun ulkopuolelle rajattiin kaikki sellaiset näkökulmat, jotka olisivat vaatineet taustatietoa sovellusten laatimisperiaatteis-

ta tai suunnitteluratkaisista. Arviossa ei niin ikään käsitelty kysymyksiä, joiden selvittäminen olisi edellyttänyt käyttäjätutkimuksia tai sovelluksen tarkastelua osana laajempaa ympäristöä. Tällaisia kysymyksiä olivat muun muassa:

- Kattaako yksittäinen tarkasteltava sovellus keskeiset opiskelutaidot ja korkeakouluopiskelussa tarvittavat valmiudet?
- Mikä on tarkasteltavan oppimisympäristön suhde opetuksen järjestävän yksikön laatuun ja muuhun opetukseen?
- Onko sovelluksen suunnittelussa käytetty joitakin standardeja tai laatu-kriteereitä, ja miten virheettömästi sovellus toimii niihin peilattuna?
- Miten sovellus onnistuu toteuttamaan asiakkaiden tarpeet ja mikä on sen käyttäjätyytyväisyys?
- Onko sovellus toteutettu kustannustehokkaasti?
- Millaisia vaikutuksia sovelluksella on opiskelijan oppimiseen ja opiskelutaidoissa tapahtuvaan muutokseen?
- Onko opiskelutaitojen erityispiirteistä johdettavissa sellaisia verkko-opetuksen suunnitteluperiaatteita, jotka poikkeaisivat olennaisesti muun verkko-opetuksen suunnitteluperiaatteista?

Osin näitä kysymyksiä on käsitelty erilaisten opiskelutaidon opetusta koskevien hankkeiden arviointien ja raportoinnin yhteydessä. Kattavaa ja systemaattista tilannekatsausta kentästä ei kuitenkaan ole, vaan oppimaan oppimisen teemaa käsitellään usein sivujuonteena jonkin muun aiheen, kuten ohjauksen tai pedagogisten kysymysten ohessa.

6. Yhteenveto

Opiskelutaitojen hallitseminen nähdään yliopisto-opintojen kannalta tärkeänä asiana. Taitoja on mahdollista oppia ja opettaa ja siten parantaa oppimistuloksia, joskin myös muilla seikoilla on vaikutusta opintomenestykseen. Korkeakoulussa opiskelutaidot ovat erityisen tärkeitä, koska akateeminen opiskelu on vaativaa, ja opiskelijoilla voi olla opintoja suorittamaan tullessaan heikot korkeakouluvalmiudet.

Verkkoperustainen toteutus sopii erinomaisesti opiskelutaitojen opetukseen joustavuutensa ja vuorovaikutteisuutensa ansiosta, mutta suunnittelussa on huomioitava erilaiset haasteet ja kompastuskivet sekä noudatettava yleisiä hyvän suunnittelun periaatteita. Oppimistaitojen opetusta ei vielä laajamittaisesti ole korkeakoulukentällä viety verkkoon, kuten ei muutakaan opetusta, mutta käytössä olevissa sovelluksissa on potentiaalia. Opiskelutaitoja tukevissa ympäristöissä tulisi kiinnittää huomiota erityisesti vuorovaikutteisuuteen, oppijalähtöiseen lähestymistapaan ja opiskelijan motivaation ylläpitämiseen.

Viiteluettelo

- Akateemiset opiskelutaidot*. Noudettu 15.5.2007, osoitteesta <http://www.oulu.fi/verkostovatti/>.
- Brockbank, A., & McGill, I. (1998). *Facilitating reflective learning in higher education*. Buckingham: Society for Research into Higher Education.
- Gibbs, G. (1981). *Teaching students to learn a student-centred approach*. Milton Keynes: Open University Press.
- Hartikainen, J. *Teekkarin tehopenaali*. Noudettu 11.4.2007, osoitteesta <http://www.tkk.fi/Yksikot/Opintotoimisto/Opetuki/tehopenaali/index.htm>.
- Heikkilä, M., Nevgi, A., & Haarala-Muhonen, A. (2005). Verkko-opetuksen laatu työ. Teoksessa A. Nevgi, E. Löfström & A. Evälä (Toim.), *Laadukkaasti verkossa yliopistollisen verkko-opetuksen ulottuvuudet* (s. 33-43). Helsinki: Helsingin yliopisto.
- IQ form - verkko-opiskelun ohjaus- ja arviointijärjestelmä*. Noudettu 11.4.2007, osoitteesta <http://www.edu.helsinki.fi/iqform/>
- Koli, H., & Silander, P. (2002). *Verkko-oppiminen oppimisprosessin suunnittelu ja ohjaus*. Hämeenlinna: Hämeen ammattikorkeakoulu.
- Korhonen, V. (2005). Oppiminen ja sen ohjaaminen verkko-opiskelussa. Teoksessa A. R. Nummenmaa, M. Lairio, V. Korhonen & S. Eerola (Toim.), *Ohjaus yliopiston oppimisympäristöissä* (s. 161-177). Tampere: Tampere University Press.
- Koski, T. (2000). Akateemisten opiskelutaitojen tukeminen. Teoksessa R. Vuorinen (Toim.), *Opiskelun ohjaus ja arviointi verkkoympäristöissä korkea-asteen koulutuksessa* (s. 65-74). Joensuu: Joensuun yliopisto.
- Manninen, J. (2003). Ohjaus verkkopohjaisessa oppimisympäristössä. Teoksessa J. Matikainen (Toim.), *Oppimisen ohjaus verkossa* (s. 27-40). Helsinki: Palmenia-kustannus.
- Nevgi, A., & Heikkilä, M. (2005). Yliopistollinen verkko-opetus. Teoksessa A. Nevgi, E. Löfström & A. Evälä (Toim.), *Laadukkaasti verkossa yliopistollisen verkko-opetuksen ulottuvuudet* (s. 19-31). Helsinki: Helsingin yliopisto.
- Nokelainen, L., & Sointu, L. (2003). Oppimista ja opiskelua ohjaavat materiaalit. Teoksessa J. Matikainen (Toim.), *Oppimisen ohjaus verkossa* (s. 69-87). Helsinki: Palmenia-kustannus.
- Opetusministeriö (2006). *Verkko-opetuksen laatupalkinnot kolmelle hankkeelle tiedote*. Noudettu 15.5.2007, osoitteesta http://www.minedu.fi/OPM/Tiedotteet/2006/4/verkko-opetuksen_laatupalkinnot_kolmelle_hankkeelle?lang=fi.
- Pantzar, E. (1986). *Aikuisen opiskelutaito perusteet ja kehittäminen*. Helsinki: Valtion painatuskeskus.
- Tervaportti, M. (Toim.). (1999). *Yliopisto-opetus ja opintoaineistot verkossa opintoaineistot verkossa -hankkeen raportti 31.3.1999*. Helsinki: Helsingin yliopiston

opiskelijakirjasto.

- Virtanen, P. (2001). *Oppimisen eväitä opiskelijalle ja tukea opettajalle ohjausta opiskelutaitoihin IQ-FORM- virtuaaliyliopistohankkeen virittämänä*. Helsinki: Helsingin yliopisto, Kasvatustieteellinen tiedekunta.
- Väljärvi, J. (1997). *Millä eväillä lukiosta yliopistoon? lukiolaisten opiskeluvalmiudet korkeakoulujen opettajien arvioimina*. Jyväskylä: Koulutuksen tutkimuslaitos.
- Wilson, J. D. (1981). *Student learning in higher education*. London: Croom Helm.

Ei-visuaaliset vuorovaikutuskeinot ja multimodaalisuus mobiililaitteissa

Eveliina Houni

Tiivistelmä.

Tässä tutkielmassa esitellään käytössä olevia sekä parhaillaan suunnitteilla olevia kannettavien laitteiden ei-visuaalisia vuorovaikutuskeinoja. Normaalisti näkevien käyttäjien kannalta visuaalista kanavaa ei voi unohtaa kokonaan, joten sen roolia multimodaalisessa käyttöliittymässä tarkastellaan multimodaalisuuden yhtenä osana. Lisäksi tarkastellaan sitä miten muut vuorovaikutustavat tukevat visuaalista käyttöliittymän perustaa. Käyttäjillä voi olla mitä erilaisimpia rajoitteita laitteiden käytössä, joten työssä tarkastellaan myös laitteiden universaaliala suunnittelua eli kaikille soveltuvan käyttöliittymän ominaisuuksia.

Multimodaalisia käyttöliittymiä koskevat tutkimukset ovat antaneet positiivisia tuloksia. Monet meistä ovat jo niin tottuneita käyttöliittymiin, joissa näyttö on dominoiva, että voisimme hämmästyä, kuinka ilman näyttöä toimiva käyttöliittymä voi menestyä käyttäjätesteissä jopa paremmin kuin näytölliset käyttöliittymät.

Avainsanat ja -sanonnat: Multimodaalisuus, mobiili, ei-visuaalinen vuorovaikutus.

CR-luokat: C5.3, F1.2, H1.2, H5.1, H5.2

1. Johdanto

Kannettavat laitteet ovat viime vuosikymmenenä yleistyneet valtavasti langattomien teknologioiden kehittyessä. Lähes jokainen länsimaalainen kantaa mukanaan ainakin kännykkää tai kämmentietokonetta ja suuri osa sen lisäksi kannettavaa tietokonetta. Joka viidennellä ihmisellä maailmassa on kännykkä [Plos and Buisine, 2006].

Työskentelyn siirtyessä ihmisten mukana toimiston ja kodin ulkopuolelle vaaditaan laitteilta uusiin tilanteisiin sopivia ominaisuuksia. Käyttäjiltä vaaditaan huomion jakamista erilaisten kognitiivisten tehtävien kesken. Koska ihminen kuitenkin kykenee havaitsemaan vain rajallisen määrän tietoa ympäristöstään, täytyy mobiililaitteita kehittää niin, että tarpeen tullen niiden täytyy kiinnittää käyttäjän huomio, mutta toisaalta niiden täytyy myös olla huomaamattomia. Liikkeessä oleva käyttäjä ei voi suunnata kaikkea visuaalista huomiotaan mobiililaitteen käyttämiseen, joten modaliteetteja ja multimodaalisuutta on tarpeen lisätä.

Modaliteetti tarkoittaa aistikanavaa, kuten näkö, tunto tai kuulo. *Multimodaalinen vuorovaikutus* (multimodal interaction) tarkoittaa sitä, että käyttäjä on vuorovaikutuksessa tuotteen kanssa käyttäen useampaa kuin yhtä modaliteettia. Esimerkki multimodaalisesta toiminnosta on autolla ajaminen [Sinkkonen et al., 2006].

Perinteisesti käyttöliittymäsuunnittelijat ovat keskittyneet informaation visuaaliseen esitystapaan. Se johtunee siitä, että suurin osa käyttäjistä on näkökykyisiä ja siitä, että visuaalinen kanava on informatiivisin. Mobiililaitteiden näyttötila on kuitenkin rajallinen, joten visuaalinen informaatio jää usein vähäiseksi.

Uusien vuorovaikutuskeinojen käytännön tarve on suuri erityisesti niillä, joilla jotkut aistikanavat on varattu tai ei ole lainkaan käytössä. Tällaisia käyttäjiä ovat esimerkiksi näkörajoitteiset ja kielellisesti tai motorisesti rajoittuneet ihmiset. Saavutettavuus on yksi tavoitelluimmista ominaisuuksista uusissa käyttöliittymissä ja erityisen tärkeä niille, joilta puuttuu kyky havainnoida ympäristöä enemmistön tavoin. Näkörajoitteisille esimerkiksi on kehitetty paljon työvälineitä erilaisen tietotekniikan käytön helpottamiseksi, kuten ääni-, ele- ja kosketuskäyttöliittymät. Tutkielmassa perehdytään niiden käyttöön mobiilissa maailmassa.

2. Mobiilin vuorovaikutuksen taustaa

Matkapuhelinvalmistajat ovat yhä tietoisempia laajasta käyttäjäkunnastaan ja heidän erityistarpeistaan. Ne ovat myös alkaneet käyttää enemmän ääntä ja tuntopalautteita tuotteissaan. Tietokoneista tutut ruudunlukijat ovat myös saatavilla joihinkin kännyköihin ja puheesyötteillä voi käyttää matkapuhelimien monia toimintoja. Mobiilissa vuorovaikutuksessa pyritään kuitenkin myös laitteen turvalliseen käyttöön joskus jopa vaarallisessa ympäristössä.

Mobiilin vuorovaikutuksen näkyvin ero verrattuna esimerkiksi pöytätietokoneen vuorovaikutustapoihin on hiiren ja näppäimistön puuttuminen. Mobiilin laitteen käyttöympäristö ei salli niiden käyttöä jo turvallisuuden kannalta: käyttäjän täytyy pystyä liikkumaan ympäristössään vaarantamatta terveyttään, joten mobiililaitteen samanaikaisen käytön täytyy olla toissijainen tehtävä, eikä se saa viedä käyttäjän huomiota kokonaan [Lumsden and Brewster, 2003].

Kuten suurikokoinen pistenäyttö, on myös näppäimistö ja hiiri suunniteltu niin, että tietokoneen käyttö olisi mahdollisimman tehokasta. Tehokkuutta saadaan täydellisellä keskittymisellä laitteen käyttöön, jolloin esimerkiksi molemmat kädet ovat käytössä. Mobiililaitteiden käyttö ei perustu kokonaisvaltaiseen keskittymiseen – päinvastoin. Niiden käytön täytyy olla mahdollista vaikka käyttäjän huomio ei olisikaan täydellistä tai vain yksi käsi tai vain yksi aisti olisi

käytössä. Multimodaalisuus antaa käyttäjälle mahdollisuuden valita käytettävän aistikanavansa, jolloin laitteen käyttö on entistä turvallisempaa käyttöympäristöönsä nähden.

3. Ei-visuaaliset vuorovaikutuskeinot

Laitteiden kehittyessä kannettaviksi niiden kokoa on täytynyt pienentää huomattavasti. Tämä on johtanut näyttötilan vähenemiseen ja näin ollen käytettävyyden heikkenemiseen, koska suurikokoisten laitteiden fyysisiä vuorovaikutuskeinoja on säilytetty pienissä laitteissa. Monet tutkijat ovat yrittäneet erottaa kannettavien laitteiden vuorovaikutuskeinot perinteisimpien suurten laitteiden vuorovaikutuskeinoista. Kannettavissa laitteissa ei pitäisi turvautua yksinomaan näppäimistöön ja näyttöön.

Visuaalisen näyttötilan puute ei ole ainoa ongelma, joka kohdataan suunniteltaessa pieniä mobiililaitteita. Näytöt voivat sisältää rajallisen määrän tietoa ja kun näytölle yritetään saada enemmän ja enemmän informaatiota, jää jotain välttämättä piiloon [Brewster, 1998]. Myös fyysinen syötteiden antaminen vaikeutuu laitteiden pienentyessä ja näppäimistön tiivistyessä. Äänen ja eleiden käyttö lisäinformaation tuojana saattaa olla ratkaisu näihin ongelmiin.

Ei-puhepohjainen äänipalaute on osoittautunut tehokkaaksi vuorovaikutuksen tehostajaksi mobiililaitteissa [Pirhonen et al., 2002]. Se sallii informaation esittämisen käyttäjälle häiritsemättä hänen visuaalista keskittymistään ympäristöönsä, eikä vie arvokasta tilaa pienellä mobiilinäytöllä. Tällainen äänen käyttö voi olla esimerkiksi niinkin tuttu kuin soittoaäni, varoitusääni tai nykyään 3D-ääni. 3D-ääni tarkoittaa sitä, että käyttäjä saa vaikutelman äänen tulosuunnasta ja jopa paikasta, vaikka äänen lähde ei sijaitsisikaan kyseisessä paikassa [Mereu and Kazman, 1996].

O'Neill ja muut [2006] ovat yhdistäneet ele- ja puhekäyttöliittymän testataksseen, rajoittaako visuaalisen käyttöliittymän puute vuorovaikutusta multimodaalisessa käyttöliittymässä. Näytön puuttuminen tai näytön rajoittuneisuus on tavallista mobiili- ja kannettavissa laitteissa, mutta se ei tarkoita, etteikö vuorovaikutus voisi olla multimodaalista ja menestyksekkästä.

3.1. Ei-puhepohjainen äänipalaute

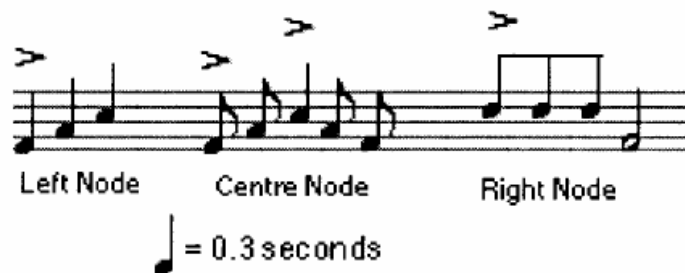
3D-äänen käytöllä on saatu hyviä tuloksia kohteen paikantamistehtävissä. Mereu ja Kazman [1997] testasivat sekä näkörajoitteisten että normaalisti näkevien kykyä paikantaa kohteita 3D-äänen avulla ja saivat merkittäviä tuloksia: äänipalaute tehostaa kaikkien käyttäjien suoritusta 3D-sovellusten käytössä, näkörajoitteiset voivat käyttää 3D-sovelluksia yhtä tehokkaasti kuin näkevät käyttä-

jät, ja näkörajoitteiset käyttäjät saavuttavat suuremman tarkkuuden paikantamisessa kuin näkevät käyttäjät pelkän äänen avulla.

Brewster [1998] kuvasi testejä, joiden avulla tutkittiin, onko mahdollista käyttää ei-puhepohjaisia audioviestejä helpottamaan navigointia valikkohierarkiassa. Ensimmäisessä testissä tavoitteena oli saada selville, tarjoavatko äänimerkit navigointiapua yleisissä valikkorakenteissa. Toisessa testissä tutkittiin ongelmia, joita kohdataan käytettäessä puhelinperustaisia käyttöliittymiä (TBI, telephone based interaction [Brewster, 1998]) ja kolmannessa testissä tutkittiin vaihtoehtoisia tapoja äänten käytössä navigointiapuna.

Brewsterin testeissä testihenkilöiden täytyi tunnistaa sijaintinsa valikoissa kuuntelemalla äänimerkkejä. Ensimmäisen testin perusteella testihenkilöt kykenivät tunnistamaan sijaintinsa yli 80%:n tarkkuudella, mikä osoittaa äänimerkkien hyödyllisyyden informaation antajina [Brewster, 2006]. Brewster mainitseekin, että tällaisten äänimerkkien käyttö olisi hyödyllistä puhelimen kautta käytettävissä käyttöliittymissä, kuten puhelinvastaajissa tai automaattisissa puhelinpalveluissa. Tällaista navigointiapua kannattaisi tarjota erityisesti näkörajoitteisille. Koska kännyköiden käyttöliittymissä käytetään paljon näppäinääniä oletuksena, saisi niistä jopa informatiivisempia esimerkiksi lisäämällä tai vähentämällä äänenkorkeutta tai muuntelemalla äänen taajuutta valikkohierarkian eri tasoilla.

Testeissä äänen laatu aiheutti äänimerkkien muistamiseen liittyviä ongelmia. Esimerkiksi puhelinautomaatista äänet tulevat puhelinlinjaa pitkin, joten luonnollisesti äänen laatu kärsii. Matkapuhelimien käyttöliittymissä tällaista ongelmaa ei olisi, koska nykyajan kännykät tuottavat todella hyvälaatuista stereo- ja jopa 3D-ääntä [Nokia, 2007]. Brewsterin testituloksissa mainittiin kuitenkin eräs mielenkiintoinen huomio, jota ei oltu tietoisesti ajateltu testiä suunniteltaessa: käyttäjät muodostivat mielessään kuulemastaan äänestä kuvan (Kuva 1). Nouseva ääni yhdistettiin valikon vasempaan solmuun. Keskimmäiseen yhdistettiin ääni, joka nousee ja laskee nopeasti ja oikeaan solmuun yhdistettiin laskeva ääni (solmulla tarkoitetaan puumaisesti kuvattua valikkohierarkian kohtaa, jonka valitsemalla käyttäjä pääsee syvemmälle valikkoon, lukemaan kyseisen kohdan tietoja tai muuttamaan niitä). Mielikuvan avulla testihenkilöt muistivat sijaintinsa valikossa paremmin.



Kuva 1. Valikkojen äänimerkit

3.2. Eleet ja puhepohjainen äänipalaute

O'Neillin ja muiden [2006] tutkimuksessa yhdistettiin ele- ja puhekäyttöliittymä ja testattiin, vaikuttaako näytön läsnäolo tai sen puuttuminen potilaiden ja heidän omaistensa tiedonsaantiin sairaalan informaatiojärjestelmässä potilaiden odottaessa hoitoa. Testiryhmiä oli kaksi: toisella ryhmällä oli käytössään näyttö varsinaisen käyttöliittymän lisäksi, toisella ryhmällä ei. Testihenkilöiden tuli käyttää järjestelmän palveluja elesyötteillä hiirtä liikuttamalla ja palaute saatiin automatisoidulla puheella. Ryhmä, jolla oli käytössään myös näyttö, sai käytettävissä olevat palvelut ja niitä vastaavat eleet visuaalisesti esitettynä. Toinen ryhmä suoritti samat tehtävät kuin ensimmäinen, mutta ilman näyttölaitetta. Osa testihenkilöistä sai kokeilla testattavaa sovellusta ennen varsinaista koetta.

Hiirieleet suoritettiin niin, että vasen hiiren näppäin pidettiin pohjassa eleen eli hiiren liikuttamisen ajan. Ele ei ollut pitkä: hiirtä liikutettiin lyhyesti yhteen suuntaan ja hiiren painike vapautettiin. Kun ele saatiin syötettyä, annettiin käyttäjälle synteettinen puheäänipalaute sen mukaan, oliko ele tunnistettu oikeaksi tai vääräksi. Myös tunnistamattoman eleen jälkeen käyttäjä sai sen mukaisen palautteen.

Tutkijat ennustivat, että jälkimmäinen ryhmä suoriutuisi tehtävistä suuremmalla määrällä virheitä kuin ensimmäinen ryhmä, jolla oli apunaan näyttö. Merkittävää eroa suorituksen virheiden määrässä ei odotusten vastaisesti havaittu. Huomattiin kuitenkin, että jälkimmäisellä ryhmällä, jolla ei ollut apunaan näyttöä, meni vähemmän aikaa palveluiden käyttämisessä kuin ensimmäisellä ryhmällä. Voidaan siis olettaa, että visuaalisen kanavan puuttuminen ei vaikuta ele- ja puhekäyttöliittymän käytettävyyteen negatiivisesti, jos palveluja ja elesyötteitä on pieni määrä ja ne ovat toisistaan erottuvia ja helposti muistettavia. [O'Neill et al., 2006]

Edellisessä tutkimuksessa ei keskitytty siihen, millä tavoin eleet saadaan syötettyä järjestelmään, vaan siihen, kuinka helppokäyttöisiä eleet ovat erilaisissa ympäristöissä. Hiirieleet ovat olleet jo pitkään käytössä joissain internet-

selaimissa, kuten Firefoxissa [Mozilla, 2007]. Mobiilissa maailmassa ei ole mahdollista tai suositeltavaa käyttää erillistä syöttölaitetta laitteiden pienen koon ja kannettavuuden vuoksi, mutta eleitä voitaisiin hyödyntää esimerkiksi liikuttamalla itse laitetta ilmassa tavalla, jonka laite voi tunnistaa. Esimerkkinä tästä on PDA-laite, jonka avulla voi selata suurtakin karttaa liikuttamalla laitetta, jolloin kartta näyttää pysyvän paikoillaan ja käyttäjä saa vaikutelman suurennuslasista ja pöydällä olevasta kartasta [Ka-Ping, 2003]. Toinen käytännön esimerkki liiketunnistimessa peliympäristössä on Nintendo Wii -pelikonsoli, jonka langattoman ohjaimen asento ja liike tunnistetaan ja vaikutus on liikettä vastaava esimerkiksi tennispelissä [Nintendo, 2007].

3.3. Näkörajoitteiset ja kosketus

Plosin ja Buisinen [2006] tekemän käytettävyyssanalyysin mukaan näkörajoitteiset ja sokeat käyttäjät käyttävät matkapuhelintaan molemmilla käsillä. Toinen käsi pitelee matkapuhelinta ja toinen tunnustelee yhdellä tai kolmella sormella näppäimiä.

Digitaalisissa käyttöliittymissä näkörajoitteisille ongelmia tuottaa tiedon jäsentely ja valikoiden rakenne. Pitkään sokeina olleet käyttäjät kehittävät mentaalimallin valikkorakenteesta ja toiminnoista, mutta vähemmän aikaa sokeina olleet käyttäjät eivät välttämättä saa selkeää mielikuvaa valikoista [Plos and Buisine, 2006]. Vaihtoehtona aiemmin mainituille valikkoäänille sokeiden käyttäjien näkökulmasta voi käyttää värinää, jonka avulla käyttäjä oppisi tunnistamaan valikkohierarkian tasot. Värinän ominaisuuksia, kuten intensiivisyyttä tai jaksotusta, muuntelemalla käyttäjälle voidaan viestiä esimerkiksi valikoiden rakennetta ja syvyyttä tai jopa kellon aikaa [Töyssy, 2007], jolloin matkapuhelinta ei tarvitse ottaa esille taskusta kellon tarkistuksen vuoksi. Töyssyn [2007] tutkimuksen mukaan värinällä tuotettu kellon ajan esitys toimii käyttäjän kokemuksesta riippuen 80-100 -prosenttisesti.

3.3.1. Ruudunlukijat

Tietokoneista tuttu tekniikka näkörajoitteisille on pistenäyttö, jota voisi hyödyntää tulevaisuudessa muissakin mobiililaitteissa. Pistenäyttö toimii niin, että näytön pinnalla on pieniä tappeja, jotka nousevat ja laskevat sitä mukaa, mitä ruudunlukija sille kertoo (Kuva 2). Tappeja on 40 – 80 jaettuna muutamalle riville, joita luetaan sormien päillä kuten pistekirjoitusta [Nkl, 2007]. Pienikokoinen pistenäyttö voi matkapuhelimen kokoisessa mobiililaitteessa antaa käyttäjälle informaatiota, kuten kellonajan tai soittajan nimen.



Kuva 2. Pistenäyttö kannettavassa tietokoneessa

Sokeille on olemassa pistenäyttöisiä matkapuhelimia, mutta niiden ongelmana on pieni kohdeyleisö. Yhdelle käyttäjäryhmälle suunnattu tuote karkoittaa helposti loput käyttäjät, jolloin tuotteen hinta nousee ja on siten saavuttamaton myös monelle sokealle.

3.4. Eleet

Kannettavan laitteen hallinnasta multimodaalisesti eleillä ja äänellä on tehty tutkimus, jossa kartoitettiin musiikkisoittimen käytettävyyttä Microsoftin Pocket PC:llä [Pirhonen et al., 2002]. Laitteessa soitetään musiikkia Windows Media Playerillä, jota käytetään laitteeseen kuuluvalla kynällä tai vaihtoehtoisesti laitteen näppäimillä (Kuva 3).



Kuva 3. Windows Media Playerin käyttöliittymä Microsoft Pocket PC:lla

Laitteen käyttäminen liikkeessä vaatii käyttäjältä paljon visuaalista huomiota [Pirhonen et al., 2002]. Pocket PC täytyy ottaa taskusta esille, jotta soitinta voi hallita. Nykyään matkapuhelimissa on lähes poikkeuksetta myös musiikkisoitin, ja esimerkiksi Nokia laittaa myyntipakkauksiin mukaan kuulokkeet, joissa musiikkisoittimen hallintaan on erillinen pieni laite liitettynä kuulokkeiden johon, jotta välttyttäisiin laitteen esille ottamiselta.

Pirhonen ja muut [2002] suunnittelivat Pocket PC:n käyttötestien perusteella musiikkisoittimen, jota voidaan hallita yksinkertaisilla eleillä ja palaute saadaan ei-puhepohjaisilla ääniviesteillä. Sitä kutsuttiin TouchPlayeriksi. Mahdolliset toiminnot olivat toisto, seuraava, edellinen, pysäytys sekä äänen voimakkuuden muuttaminen. Valintoja tehtiin liikuttamalla sormeja ruudulla: Toisto tai pysäytys tapahtui napauttamalla ruutua kerran. Seuraavaan raitaan siirryttiin pyyhkäisemällä ruutua vasemmalta oikealle ja edelliseen raitaan siirryttiin pyyhkäisemällä oikealta vasemmalle. Äänenvoimakkuutta säädettiin pyyhkäisemällä ruutua ylhäältä alas- tai ylöspäin. Eteen onnistumisen merkiksi annettiin käyttäjälle kuulokkeisiin jokaiselle toiminnolle ominainen äänimerkki, jotta käyttäjä saisi tiedon laitteen tilasta. Edelliseen raitaan siirtyminen soitettiin vasemmasta kuulokkeesta, seuraavaan raitaan siirtyminen oikeasta kuulokkeesta ja toisto sekä pysäytys keskeltä eli molemmista kuulokkeista. Äänenvoimakkuudelle ei ollut omaa äänimerkkiään, koska käyttäjä kuulee toiminnon onnistumisen heti musiikin voimakkuudessa.

Laitteen sijoittuminen käyttäjään nähden aiheutti kuitenkin ongelmia. Jos laite olisi käyttäjän vyötäröllä näyttö ulos ja eteenpäin suunnattuna, ei seuraava/edellinen -metafora toimisi, koska käyttäjän näkökulmasta vasen ja oikea olisivat eri suunnassa kuin laitteen näkökulmasta. Laitteen täytyisi olla aina käyttäjän oikealla sivulla ja pystyasennossa, jotta käyttäjä voisi miettimättä hallita laitetta eleillään. [Pirhonen et al., 2006]

4. Mobiiliuden tuomia ongelmia

Kannettavissa laitteissa pieni koko on valttia, joten se karsii käytettävyyttä tietyiltä osin suuresti. Näyttöön ei mahdu samaa määrää informaatiota kuin tietokoneen näytölle. Informaation syöttö laitteelle hidastuu näppäimistön pienuuden ja suppeuden vuoksi, eikä puheentunnistusteknologia ole vielä aivan siinä pisteessä, että laitetta voisi käyttää ainoastaan äänellä. Matkapuhelimien QWERTY-näppäimistö taas vie enemmän tilaa, jolloin koko kasvaa tai näppäimet jäävät kooltaan liian pieniksi. Eräs suurimmista ongelmista varsinkin sokeille on näppäimistön huono esiintuonti.

Tietyille käyttäjäryhmille suunnittelu karsii muiden käyttäjäryhmien tyytyväisyyttä tuotteisiin. Tuotteiden personointimahdollisuuksien täytyisi olla huomattavan laajat, mikä lisää väistämättä laitteiden hintaa. Matkapuhelinvalmistajilla on tarve suunnitella mahdollisimman monelle erilaiselle käyttäjäryhmälle omanlaisensa tuotteet, jolloin ominaisuudet voivat olla kohderyhmälle sopivia, mutta toisaalta hyvinkin rajoittuneita. Muutaman poikkeuksen yleisimpään matkapuhelintrendiin tekevä puhelinmallikin nojaa käytössään koske-

tusnäyttöön, jota hallitaan kynällä [Nokia, 2007], mutta kyseiselläkään mallilla ei ole erityistä tarjottavaa erilaisille käyttäjäryhmille.

Uusien ominaisuuksien tulviessa matkapuhelimiin niiden valikot laajenevat ja monimutkaistuvat. Matkapuhelimet tuskin enää kooltaan pienenevät, mutta käytettävyydeltään niiden suunnittelu vaikeutuu uusien ominaisuuksien myötä, kun näppäimiäkään ei voi loputtomasti lisätä. Laajoja kokonaisuuksia on vaikea hallita, varsinkin jos näköaistin tuomaa laajaa informaatiota ei ole tarjolla.

5. Suunnitteluperiaatteita

Ohjelmistosuunnittelijoilla on ollut taipumus luoda visuaalisesti hallitsevia käyttöliittymiä, osaksi koska niiden on koettu olevan erinomaisesti sulautuneita päivittäiseen ympäristöömme ja siksi looginen perusta käyttöliittymälle. Seuraavat suunnitteluperiaatteet perustuvat esimerkkisovellukseen, jonka avulla sokeat oppivat nuottikirjoitusta ja kyseisen sovelluksen tutkimuksellisiin tuloksiin [Challis, 2000]. Suunnitteluperiaatteet koskevat käyttöliittymiä, jotka ovat vähän tai eivät lainkaan visuaalisuuteen nojautuvia. Suunnitteluperiaatteet ovat seuraavat:

1. Datan kartoituksen yhdenmukaisuus tulisi säilyttää. Tämä periaate varmistaa sen, että yleiskieli säilytetään sekä visuaalisessa että ei-visuaalisessa informaation esityksessä niin, ettei mikään käyttäjäryhmä vieraannu sovelluksesta.
2. Kosketukseen perustuvan informaation esityksen tulisi keskittyä muuttumattomaan dataan. Tämän periaatteen tarkoitus on luoda perustoimintatapa sille, mikä informaatiotyyppi sopii millekin kommunikointikanavalle. On sopivampaa esittää muuttuva data esimerkiksi auditiivisena, koska se sopeutuu muutoksiin paremmin.
3. Korkeusominaisuutta voi käyttää tarjoamaan perusta informaation suodatukselle. Tämä periaate tarkoitettiin muodostamaan perusta informaation suodatukselle. Suuri korkeusero esimerkiksi kosketusnäytöllä on helpompi tuntea ja siten jättää pienemmät korkeuserot vähemmälle huomiolle.
4. Tyhjä tila todennäköisesti hämmentää sokeita käyttäjiä. Tehokkaan suunnittelun tulisi minimoida tyhjän tilan käyttö, mikä eroaa selkeästi graafisen käyttöliittymän suunnittelusta.

Erityisesti matkapuhelimille tarkoitettuja fyysisiä suunnitteluperiaatteita ovat esitelleet Plos ja Buisine [2006]. He ovat tutkineet matkapuhelimien soveltuvuutta erilaisista rajoitteista kärsiville ihmisille kuten kuulorajoitteisille, nä-

körajoitteisille tai motoriikaltaan rajoittuneille. Suunnittelussa on otettu huomioon universaali suunnittelu (*Universal Design* [Plos and Buisine, 2006]), eli kaikille käyttäjille ystävällinen käyttöliittymä, joka ei sulje pois kohderyhmän ulkopuolisia käyttäjiä. Suunnitteluperiaatteet on johdettu niin, että ensin on analysoitu jo olemassa olevia tuotteita, jonka jälkeen niistä on tehty käytettävyyssanalyysi. Tulokset on jaettu käyttäjäryhmien mukaan:

5. Sokeille käyttäjille on tarjottava vahva fyysinen käyttöliittymä, jossa näppäimien määrä on rajoitettu ja näppäimet erottuvat selvästi. Erilaisten näppäinten pinta voidaan helposti erottaa muusta laitteen rungosta muuttamalla pinnan karheutta tai pehmeyttä.
6. Näkörajoitteisille käyttäjille fonttikoko ja kontrasti on tärkeä saada riittävän suureksi. Puna-viher-yhdistelmää kannattaa välttää punavihersokeuden vuoksi.
7. Kuulorajoitteiset käyttäjät käyttävät matkapuhelinta pääasiassa tekstiviestien lähettämiseen, joten T9 eli ennustava tekstinsyöttö helpottaa kirjoittamista huomattavasti, kuten myös täysi QWERTY-näppäimistökin. Näppäimet eivät saa olla liian lähellä toisiansa tai liian pieniä.
8. Seniorikäyttäjät pitävät tärkeänä laitteen kestävyyttä ja hyvää äänentoistoa, joka ei välttämättä ole riittävällä tasolla edes voimakkaimmillaan nykymatkapuhelimissa.

Universaali suunnittelu pyrkii yhdistämään kaikki edellä mainitut kohdat niin, että laite olisi käytettävissä mahdollisimman monelle käyttäjälle sulkematta ketään pois. Mikäli laitteista esimerkiksi jätetään jokin ominaisuus pois jota tietynlainen käyttäjäryhmä ei tarvitse – kuten näyttö sokeilla – jäävät todennäköisesti suurimmat markkinat saavuttamatta.

6. Yhteenveto

Mobiililaitteideissa ei-puhuttujen äänien tai tuntopalautteen käyttö ei ole sidoksissa käytettyyn kieleen. Siksi tällaisen yleisen informaation esitystavan käyttö on suositeltavaa ja soveltuu käytettäväksi erilaisille käyttäjäryhmille. Suunniteltaessa käyttöliittymiä täytyy huomioida myös se, ettei ratkaisu sulje pois erilaisia kohderyhmiä, vaan että laite ja sen ominaisuudet ovat muokattavissa mahdollisimman monelle käyttäjälle.

Ihmiselle on luonnollinen tapa kommunikoida multimodaalisesti, mutta mobiililaitteissa vasta uusimmat ominaisuudet viittaavat tällaiseen mahdollisuuteen. Multimodaalisesta vuorovaikutuksesta on tulossa vaihtoehto, kun auditiivinen ja tuntopalaute lisääntyvät matkapuhelimissa ja PDA-laitteissa.

Vaikka niiden tarjoama informaatio ei olekaan vielä kovin monipuolista, on tästä tilanteesta hyvä lähteä kehittämään mobiilikäyttöliittymiä kaikkien saataville.

Viiteluettelo

- [Brewster, 1998] Stephen A. Brewster, Using nonspeech sounds to provide navigation cues. *ACM Transactions on Computer-Human Interaction* **5**, 3, (Sept. 1998), 224-259.
- [Challis, 2000] Ben Challis, Doctoral consortium: Design principles for non-visual interaction. In: *CHI '00 Extended Abstracts on Human Factors in Computing Systems CHI '00*, ACM Press, 73-74.
- [Hoggan and Brewster, 2006] Eve E. Hoggan and Stephen A. Brewster, Work-in-progress: Crossmodal icons for information display. In: *CHI '06 Extended Abstracts on Human Factors in Computing Systems*, ACM Press, 857 - 862.
- [Ka-Ping, 2003] Yee Ka-Ping, Interaction techniques for handheld devices: Peephole displays: pen interaction on spatially aware handheld computers. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems CHI '03*, ACM Press, 1-8.
- [Kientz et al., 2006] Julie A. Kientz, Shwetak N. Patel, Arwa Z. Tyebkhan, Brian Gane, Jennifer Wiley and Gregory D. Abowd, Navigational assistance: Where's my stuff?: design and evaluation of a mobile system for locating lost items for the visually impaired. In: *Proceedings of the 8th International ACM SIGACCESS Conference on Computers and Accessibility Assets '06*, ACM Press, 103-110.
- [Lumsden and Brewster, 2003] Joanna Lumsden and Stephen Brewster, A paradigm shift: alternative interaction techniques for use with mobile & wearable devices. In: *Proceedings of the 2003 Conference of the Centre for Advanced Studies on Collaborative Research CASCON '03*, IBM Press, 197-210.
- [Mereu and Kazman, 1996] Stephen W. Mereu and Rick Kazman, Audio enhanced 3D interfaces for visually impaired users. In: *ACM SIGCAPH Computers and the Physically Handicapped* Issue 57, (Jan 1997), ACM Press 10-15.
- [Mozilla, 2007] Mozilla, Mouse Gestures Add-ons. Available at <https://addons.mozilla.org/firefox/39/>. Checked 22.2.2007.
- [Nintendo, 2007] Nintendo Wii. Saatavilla <http://www.nintendo.fi/?path=wii>. Tarkistettu 23.4.2007)
- [Nkl, 2007] Näkövammaisten keskusliitto, Tietotekniset ratkaisut. Saatavilla <http://www.nkl.fi>. Tarkistettu 3.3.2007.

- [Nokia, 2007] Nokia Suomi, Nokia N92. Saatavilla <http://www.nokia.fi>. Tarkistettu 5.3.2007.
- [Nokia, 2007] Nokia Suomi, Nokia N800 Internet Tablet. Saatavilla <http://www.nokia.fi>. Tarkistettu 23.4.2007.
- [O'Neill et al., 2006] Eamonn O'Neill, Manasawee Kaenampornpan, Vassilis Kostakos, Andrew Warr and Dawn Woodgate, Can we do without GUIs? Gesture and speech interaction with a patient information system. *Personal and Ubiquitous Computing* **10**, 5, (Jul 2006), 269-283.
- [Pirhonen et al., 2002] Antti Pirhonen, Stephen Brewster and Christopher Holguin, Speech, Audio, Gesture: Gestural and audio metaphors as a means of control for mobile devices. In: *Proceedings of the SIGCHI conference on Human Factors in Computing Systems: Changing our World, Changing Ourselves CHI '02*, ACM Press 291-298.
- [Plos and Buisine, 2006] Ornella Plos and Stéphanie Buisine, Work-in-progress: Universal design for mobile phones: a case study. In: *CHI '06 Extended Abstracts on Human Factors in Computing Systems CHI '06*, ACM Press 1229-1234.
- [Sinkkonen et al., 2006] Irmeli Sinkkonen, Hannu Kuoppala, Jarmo Parkkinen ja Raino Vastamäki, Käytettävyyssanasto. Saatavilla <http://www.adage.fi/julkaisut/arkisto/kaytettavyysanasto.html#m>. Tarkistettu 26.2.2007.
- [Töyssy, 2007] Sampo Töyssy, henkilökohtainen tiedonanto, aiheina: Ajan kertominen värinällä. 23.4.2007

IP-liikenteen luokittelu

Mikko Lammi

Tiivistelmä.

IP-verkko ei suoraan tarjoa mitään menetelmiä liikenteen lajitteluun, vaan kohtelee kaikkia paketeja samanarvoisesti. Haluttujen sovellusten tai käyttäjien liikenne onkin ensin tunnistettava ja luokiteltava, jotta verkkolaitteet voisivat priorisoida sitä, tai vaihtoehtoisesti estää ei-toivotun liikenteen. Jatkuvasti kasvavat liikennemäärät ja yhä monimutkaisemmat luokitteluvaatimukset asettavat kuitenkin haasteita luokittelualgoritmien tehokkuudelle.

Avainsanat ja -sanonnat: IP-verkot, QoS, pakettien luokittelu, luokittelualgoritmit

CR-luokat: C.2.2, C.2.3, C.2.6, F2.2

1. Johdanto

Tässä tutkielmassa tarkastellaan lähestymistapoja ja menetelmiä IP-paketeista koostuvan verkkoliikenteen luokitteluun, keskittyen erityisesti Internetin runkoverkkotasolle. Liikenteen luokittelulla pyritään yleensä kahteen päämäärään: takaamaan valitulle liikenteelle riittävät resurssit määriteltyjen raja-arvojen puitteissa (kaistanleveys, siirtoviive) tai valitsemaan, päästetäänkö liikennettä lainkaan läpi. Luokittelu on itsenäinen operaatio, jolla vain tunnistetaan haluttu liikenne, eikä se ota kantaa siihen, mitä paketeille tehdään seuraavaksi. Tavallisesti kuitenkin luokittelutietoa hyödynnetään erilaisiin pakettien välitykseen vaikuttaviin toimenpiteisiin, kuten reititykseen, palomuuraukseen tai kuristamiseen haluttuun kaistanleveyteen.

Sovellusten tunnistaminen, palvelunlaadun takaaminen ja haittaliikenteen suodattaminen ovat keskeisimmät vaatimukset luokittelun toteuttamiselle. Tehokkaat algoritmit pakettien luokittelua varten ovatkin keskeisessä asemassa uusissa verkkolaitteissa tai verkkoliikennettä käsittelevissä sovelluksissa. Olennaista ei ole enää pelkkä raaka suorituskyky liikenteen sisällöstä välittämättä, vaan *suorituskyky liikenne luokitellen*. Toisaalta hajautettujen luokittelumenetelmien ja verkkojen optimoinnin myötä pyritään myös siihen, että kaikilta laitteilta ei vaadita samoja ominaisuuksia, vaan luokittelua voidaan keskittää vain tiettyihin komponentteihin verkon reunalla tai ruuhkaisimmissa paikoissa.

Seuraavassa luvussa kuvataan lyhyesti IP-verkkojen rakenne keskittyen luokittelun kannalta olennaisiin seikkoihin. Vaikka tutkielma esitteleekin suppeasti IP-verkon toimintaperiaatteen, on aiempi verkkotekniikan tuntemus kuitenkin melko pakollinen aihepiirin ja käsiteltyjen asioiden ymmärtämiseksi.

Luvussa kolme puolestaan käydään läpi tärkeimmät syyt luokittelun toteuttamiselle sekä palvelunlaadun toteuttamisen lähestymistavat. Neljännessä luvussa kuvataan luokittelijan toimintaa sekä esitellään kriteerit luokittelualgoritmiin arviointiin. Lisäksi tutustaan lyhyesti myös tilastollisiin menetelmiin sekä liikenteen sisältöön perustuvaan luokitteluun. Viidennessä luvussa tutustaan joukkoon algoritmeja ja menetelmiä, jotka pyrkivät toteuttamaan pakettien luokittelua ja pohditaan niiden ominaisuuksia aiemmin määriteltyjen kriteerien valossa. Kuudes luku sisältää yhteenvedon luokittelusta ja joitakin havaintoja ja pohdintoja luokittelun tarpeesta tulevaisuudessa.

2. Liikenteen välittäminen IP-verkoissa

2.1. TCP/IP-protokolla Internetin perustana

Internet Protocol (IP) on nykypäivän tietoverkoissa tärkein verkkotason protokolla. Kiinteän verkon dataliikenteen lisäksi sitä käytetään yhä enemmän myös puheen ja liikkuvan kuvan siirtämiseen sekä kiinteässä että mobiilissa verkossa. IP-protokollaa käytettäessä siirrettävä data pilkotaan paketeiksi, joihin liitetään vähintään otsikkotieto paketin lähde- ja kohdeosoitteesta, yleensä myös muita ylempien verkkokerrosten tietoja. Näiden avulla verkon reitittimet ohjaavat paketin perille kohteeseen tai pystyvät palauttamaan lähettäjälle tiedon lähe-tyksen epäonnistumisesta. On tärkeätä huomata, että jokainen yksittäinen paketti reititetään erikseen, ja onkin teknisesti mahdollista, että samaan tietovirtaan kuuluvat paketit kulkevat kohteeseensa eri reittejä pitkin.

Tietoverkkoa voidaan kuvata erilaisilla kerrosmalleilla, joissa jokaisella tasolla on omat tehtävänsä, ja ylempät tasot nojaavat alempien tasojen tarjoamiin palveluihin. Kansainvälisen standardointijärjestön ISO:n kehittämässä OSI-mallissa IP sijaitsee tasolla kolme (verkkokerros). Internet-verkkojen liikenteen yhteydessä tarkoitetaan useimmiten TCP/IP –protokollaperhettä, jolloin mukaan otetaan myös verkkotason yläpuolella oleva kuljetuskerros. TCP (Transfer Control Protocol) tarjoaa välineet siirretyn tiedon eheyden varmistamiselle, mutta mikäli tästä voidaan tinkiä, on myös mahdollista käyttää yksinkertaisempaa UDP-protokollaa kuljetuskerroksessa [Wikström, 2006]. Sovellukset voivat lisäksi lisätä omia otsikkotietojaan ylemmille tasoille, ja toisaalta ennen verkkokerroksen otsikoita pakettiin on tavallisesti liitetty myös linkin kuljetuskerroksen (esimerkiksi Ethernetin) otsikkotiedot, mutta yleensä luokittelussa keskitytään vain tasoihin kolme ja neljä.

Tässä tutkielmassa IP-protokollalla viitataan sen tällä hetkellä ylivoimaisesti eniten käytössä olevaan IPv4-versioon, jossa jokainen osoite on 32 bittiä pitkä. Sen seuraajaksi on kuitenkin jo kehitetty IPv6-protokolla, joka laajentaa osoi-

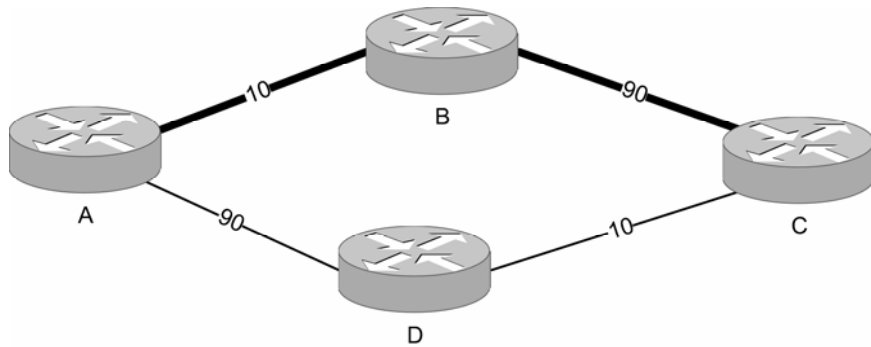
teavaruuden kokoa siten, että yksittäiset osoitteet ovat 128-bittisiä. Lisäksi IPv6 sisältää huomattavasti parannuksia verkon reititykseen ja turvallisuuteen liittyen, joista monet myös vaikuttavat luokitteluun ja palvelunlaatuun. Koska kuitenkin IPv6:tta ei ole vielä laajalti tuotantokäytössä, vaan Internet toimii pääasiassa IPv4:n varassa, ei tässä tutkielmassa puututa IPv6-liikenteen luokitteluun. Useimmat luokittelumenetelmät kuitenkin voidaan suoraan ottaa käyttöön myös IPv6-maailmassa, kunhan aika- ja tilakompleksisuuden näkökulmasta huomioidaan, että IPv6-osoiteavaruuden koko on IPv4:n avaruus neljänteen potenssiin korotettuna.

2.2. IP-verkon komponentit ja reititys

IP-verkoissa jokaisella verkkoon kytketyllä laitteella tulee olla oma, yksilöllinen IP-osoite, jonka perusteella paketit osataan ohjata oikeaan paikkaan. Globaali Internet-verkko koostuu tuhansista pienemmistä verkoista, jotka puolestaan on edelleen pilkottu pienemmiksi osasiksi, aliverkoiksi (jotka taas voidaan edelleen jakaa pienemmiksi osasiksi tarpeen mukaan). Liikennettä eri verkkojen välillä välittävät reitittimet, ja liikennettä voidaan suodattaa palomuuureilla, jotka voivat sijaita joko reitittimien yhteydessä tai keskellä verkkoa. Näistä verkon aktiivilaitteista voidaan käyttää yleisnimitystä *verkkoelementit* (Network Elements).

Verkoissa on yleensä useita vaihtoehtoisia reittejä liikenteen kuljettamiselle, ja suurissa verkoissa reitittimet vaihtavat jatkuvasti keskenään automaattisesti tietoja eri reiteistä tarkoitusta varten kehitettyjen reititysprotokollien avulla. Verkossa kulkeva liikenne voidaanankin jakaa kahteen tasoon: varsinainen verkon läpi kuljetettava eri sovellusten liikenne *datakerrokseen* ja verkkoelementtien välinen reititystieto *kontrollikerrokseen* [Wang, 2001]. Nämä eivät välttämättä aina kulje samoja reittejä, sillä kontrollikerroksen voidaan nähdä käsittävän yhteydet kaikkien reititysalueen reitittimien välillä, kun taas datakerros on reitittimien liikenteelle valitsema reitti verkon halki.

Perinteiset reititys algoritmit ovat pohjautuneet graafiteoriaan, jossa reitittimet ovat solmuja ja niiden väliset linkit kaaria. Tämä on kuitenkin varsin rajoittunut tapa kuvata verkkoa, sillä kaarille on voitu antaa vain yksinkertainen painoarvoluku, jonka pohjalta reititin on valinnut käytettävän reitin. Pelkkä solmujen määrä tai kaaren painoarvo ei kerro vielä mitään valitun reitin tilasta. Kuva 1 havainnollistaa hyvin yksinkertaista, neljästä reitittimestä muodostuvaa verkkoa, jossa linkkien kokonaispainoarvo reitittimien A ja C välillä on sama. Tämän vuoksi reitittimisissä pyritäänkin nykyään löytämään luokitellulle liikenteelle parhaiten sille sopiva väylä joko ennalta määritellyistä reiteistä tai automaattisesti reititysprotokollien perusteella.



Kuva 1. Reititys reitittimien A ja C välillä. Kahdesta kokonaisuutena samanarvoisesta reitistä on valittu B:n kautta kulkeva.

IP-reitityksessä verkoista käytetään usein ilmausta prefiksi tai CIDR-muoto (Classless Inter-Domain Routing). Prefiksin pituus on vasemmalta käsin lasketuna merkitsevien bittien määrä [Opperman, 2005]. Esimerkiksi verkko 192.168.0.0 – 192.168.3.255 voidaan ilmaista prefiksillä 192.168.0.0/22. Taulukosta 1 nähdään selkeästi, miten binäärimuodossa esitetty osoite 192.168.2.31 osuu prefiksiin, mutta sen sijaan osoite 192.168.4.1 ei osu.

IP-osoite/prefiksi	Binäärimuoto	Selite
192.168.0.0/22	11000000.10101000.000000xx.xxxxxxxx	/22 –notaatio tarkoittaa, että prefiksi kattaa 22 bittiä osoitteen alusta.
192.168.2.31	11000000.10101000.00000010.00011111	Ensimmäiset 22 bittiä ovat samat kuin prefiksissä, osoite kuuluu verkkoon.
192.168.4.1	11000000.10101000.00000100.0000001	Ensimmäiset 22 bittiä eivät täsmää, osoite ei kuulu verkkoon

Taulukko 1. Esimerkki CIDR-prefiksien käytöstä.

2.3. Pakettiverkon ongelmat

Internet on edeltäjänsä ARPANETin (Yhdysvaltain Puolustusministeriön 1960-luvulla rakentama verkko, jonka toteutuksessa pyrittiin kestävyyyteen sotatilanteiden varalta) tapaan pakettimallin varaan rakennettu. Pakettimallin etuina ovat yksinkertaisuus ja kyky mukautua automaattisesti ja nopeasti verkkotopologian muutoksiin. Kun verkko laajeni 1990-luvulla tiedeyhteisön tutkimusverkosta maailman suurimmaksi julkiseksi verkoksi pääasiassa World Wide Webin ja sen sovellusten myötä, havaittiin erinäisiä puutteita verkon perusrakenteissa.

Perinteisestä IP-verkosta puuttuvat täysin menetelmät sovellusten tunnistamiseksi ja suorituskyvyn varmistamiseksi. Verkko kohtelee kaikkia paketteja niiden kohteesta, lähteestä tai sisällöstä riippumatta samanarvoisesti, ja täten hyvinkin erityyppiset sovellukset joutuvat kilpailemaan samoista verkkoresursseista. Jos jossakin verkon osassa on ruuhkaa, pakettien toimitus voi viivästyä, tai ne eivät pääse lainkaan perille asti. Uudemmat interaktiiviset sovellukset ovat erittäin herkkiä siirtoviiveelle ja pakettihävikille, kun taas toisaalta perinteisemmät sovellukset, kuten tiedostojen siirto, kestävät näitä melko hyvinkin [Wang, 2001].

3. Resurssien varaus ja palvelunlaadun varmistaminen

3.1. QoS

Internetin kaupallistuessa ja käytön monipuolistuessa liikennemäärät alkoivat kasvaa verkossa nopeasti. Verkkotekniikan ammattilaiset olettivat 1990-luvun alkupuolella, että lähitulevaisuudessa puhe- ja videoneuvottelut muodostuisivat Internetin "killer applicationiksi" tuoden sen yhä useampien ulottuville (todellisuudessa tämän kuitenkin toteutti huomattavasti yksinkertaisempi sovellus, World Wide Web). Videoneuvottelua testattaessa huomattiin kuitenkin pian, että perinteisessä tasa-arvoisessa IP-verkossa puhelut ja muut reaaliaikaiset sovellukset kärsivät erittäin helposti verkon kuormituksesta tai viiveestä. Samalle linkille osuva ei-reaaliaikainen sovellus, kuten tiedostojen siirto, voi viedä usein huomattavasti enemmän kapasiteettia ja ruuhkauttaa yhteyden, jolloin häiriöille herkempi reaaliaikainen sovellus kärsii ensimmäisenä. Tämän vuoksi alettiinkin pohtia menetelmiä verkkoresurssien varaamiseksi ja erilaisten sovellusten liikenteen tunnistamiseksi sekä erottamiseksi toisistaan. Yleisesti näiden toteutusta kutsutaan nykyään nimellä *Quality of Service* (QoS) [Wang, 2001], josta voidaan myös käyttää suomeksi ilmausta *palvelunlaatu*.

3.2. IntServ ja DiffServ

QoS-toteutukset jaetaan yleisesti kahteen eri lähestymistapaan. Ensimmäinen näistä on vuonna 1994 IETF:n RFC-dokumentissa 1663 määritelty *IntServ* (Integrated Services) -malli. Siinä jokainen verkkoelementti luokittelee erikseen paketit omien sääntöjensä perusteella. Tämä noudattaa alkuperäistä IP:n pakettikytkentäistä ajattelumallia. Toteutus on kuitenkin melko raskas: jokaisen verkkoelementin on toistettava sama luokitteluprosessi pakettien matkan varrella jokaiselle läpi kulkevalle paketille, ja luonnollisesti kaikkiin laitteisiin on myös määriteltävä samat luokittelusäännöt. IntServiin kuuluu olennaisena osana sovelluksen etukäteen tekemä verkkoresurssien varauspyyntö RSVP-protokollalla. Toteutuksen monimutkaisuuden ja väärinkäytösten riskien vuok-

si IntServ- ja RSVP-toteutuksia ei juuri ole käytössä julkisessa Internetissä, mutta sen sijaan organisaatioiden sisäisissä verkoissa niitä hyödynnetään jonkin verran.

Vaihtoehtoiseksi ratkaisuksi onkin myöhemmin kehitetty *DiffServ* (Differentiated Services) –teknologia, jossa liikennettä käsitellään yksittäisten pakettien sijaan suurempina kokonaisuuksina, kuten liikennevirtoina tai lähde/kohdeverkon perusteella aggregoituna kokonaisuutena. IETF:n RFC 2475:ssä vuonna 1998 määritellyn DiffServ-mallin merkittävin etu on kuitenkin se, että siinä paketit tarvitsee luokitella tarkkojen sääntöjen perusteella vain kerran verkon reunalla. Tässä yhteydessä niiden otsikkotietoihin leimataan kuusibittinen DSCP-tunniste, jonka avulla seuraavat verkkoelementit voivat suoraan erottaa jo valmiiksi luokitellut paketit muun liikenteen joukosta. Yleensä DiffServ-merkinnät pätevät vain yhden operaattorin verkon sisällä, mutta niitä voidaan IntServiä joustavammin käyttää myös operaattoreiden välisessä liikenteessä.

Kummassakin menetelmässä kuitenkin keskeinen haaste on pakettien luokittelu. Vaikka DiffServ-tekniikkaa käyttäen se tarvitsee tehdä vain kerran verkon reunalla, saattaa tämäkin nykypäivän verkoissa merkitä valtavia pakettimääriä. Esimerkiksi Suomen kaupallisen Internet-liikenteen FICIX-yhdysliikennepisteessä suurimmilla operaattoreilla pakettimäärät ovat ruuhka-aikana 200 000 pakettia sekunnissa tai enemmän [FICIX, 2007]. Lisäksi pakettien tunnistamisen ja luokittelun merkitys korostuu *hajautettuja palvelunestohyökkäyksiä* (DDoS, Distributed Denial of Service) tai matojen leviämisepidemiaa torjuttaessa, joiden kohdalla luokiteltavat pakettimäärät saattavat kohota moninkertaisiksi normaaliin liikenteeseen verrattuna. Menetelmät eivät myöskään poista sitä perusongelmaa, että liikenne saattaa kulkea useiden operaattoreiden verkkojen kautta, eikä ole takeita siitä, että kaikissa pätisivät samat luokittelusäännöt.

3.3. Haittaliikenne

Internet-verkko rakennettiin alun perin suljetuksi verkoksi, eikä sitä suunniteltaessa osattu varautua kaikille avoimen käytön aiheuttamiin ongelmiin. Koska verkko on hyvin lyhyessä ajassa laajentunut maailmanlaajuisesti satojen miljoonien käyttäjien verkoksi, ei sen pohjana olevien tekniikoiden puutteita voida helposti korjata, sillä ne vaatisivat muutoksia kaikkiin verkkoon kytkeytyviin laitteisiin tai niiden ohjelmistoihin. Käytännössä siis tekniikoiden ytimessä olevien ongelmien kanssa on elettävä ja niitä on pyrittävä paikkaamaan muilla tavoilla.

Verkossa liikkuvan liikenteen sisällöstä on käyty kovasti keskustelua, sillä valtioiden rajat ylittävä verkko tarjoaa helposti mahdollisuudet rikolliseen toi-

mintaan anonyymisti kiinnijäämisen pelon ollessa pieni. Tätä olennaisempi asia on kuitenkin liikenteen sisällöstä riippumaton, verkon infrastruktuurin kannalta haitallinen liikenne, joka nykyään on valitettavasti Internetin arkipäivää eri muodoissaan. Haittaliikenne voidaan karkeasti luokitella kolmeen eri ryhmään: *tahallinen* haittaliikenne, jota ovat tyypillisesti hajautetut palvelunestohyökkäykset (DDoS) sekä roskaposti, *automaattinen* haittaliikenne, joka syntyy matojen levitessä, sekä viimeisenä *tahaton* haitallinen liikenne, jota voi olla esimerkiksi *peer-to-peer* (P2P) –tiedostonjako-ohjelmien aggressiivinen verkkoresurssien hyödyntäminen odottamattomalla tavalla. Vaikka P2P:iä ei aina haittaliikenteeksi lasketakaan, sille on kuitenkin yhteistä muiden tapaan se, että liikenne on operaattorin näkökulmasta sellaista, mitä verkon ei ole välttämättä alun perin suunniteltu välittävän, ja se saattaa siten vaarantaa muun liikenteen sujuvuuden.

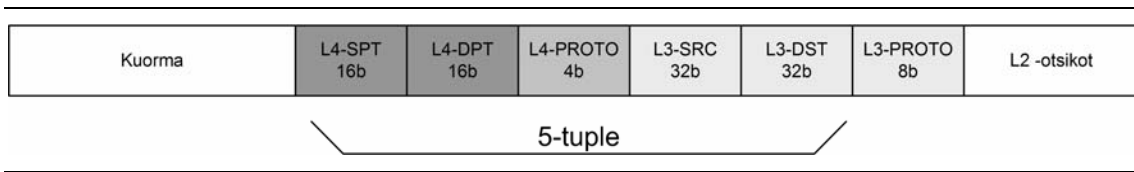
4. Liikenteen luokittelu

4.1. Pakettien luokittelu otsikkokenttien perusteella

Jokainen IP-paketti pitää sisällään samanlaiset otsikkotiedot, joiden avulla pakettien luokittelua ensisijaisesti tehdään. Pakettien luokittelija määrittää yleisesti järjestelmäksi, joka pystyy tunnistamaan paketit vertaamalla paketin otsikkokentän tietoja [Wang, 2001]. Luokittelussa paketti kulkee yhden tai useamman luokittelusäännön läpi, joissa paketin otsikkokenttiä verrataan ennalta määriteltuihin arvoihin. Vertaaminen voi olla tyypiltään eksaktia, pisimpään osumaan (yleensä prefiksistä) perustuvaa tai raja-arvojen puitteissa tapahtuvaa. Gupta ja McKeown [1999] määrittelevät myös pelkän reitityksen pakettien luokitteluksi. Reitityksessä keskitytään kuitenkin tavallisesti vain paketin kohdeosoitekenttään, joten kyseessä on yksinkertainen, yhden komponentin luokittelutapaus.

Varsinaisessa liikenteen luokittelussa pyritään useimmiten luokittelemaan paketit liikennevirtoihin (flow) niin kutsutun viitosmonikon (fivetuple) tai sen alitapausten perusteella. Kuvassa 2 on esitetty IP-paketin otsikkokentät OSI-mallin tasoilta kolme ja neljä. Liikennevirta on yksilöllinen istunto lähteen ja kohteen välillä, ja perinteisesti sen pakettien otsikkotiedoista on pystytty myös päättämään sovellus lähde- tai kohdeportin perusteella. IntServ-tyyppiset QoS-menetelmät perustuvat juuri liikennevirtojen tunnistamiseen. Paketin otsikkotietoihin ei kuitenkaan voida enää pelkästään luottaa luokittelussa, pääasiassa haittaliikenteen ja väärennettyjen otsikkokenttien vuoksi, mutta myös uudentyyppisten sovellusten (esimerkiksi P2P) yleistymisen seurauk-

senä. Jos tavoitteena on tunnistaa varmasti tietyn sovelluksen liikenne, tarvitaan monimutkaisempia luokittelumenetelmiä.



Kuva 2. IP-paketin otsikkokentät.

Pakettien luokittelusääntö voidaan määritellä yleisellä tasolla seuraavasti: jokaisessa luokittelijan säännössä R on d komponenttia. $R[i]$ on säännön i :s komponentti sekä vertailulauseke paketin otsikon kentälle i . Paketti P tunnistuu sääntöön R , jos kaikille otsikkokentille i pätee vertailulauseke $R[i]$ [Gupta and McKeown, 2001]. Koska vertailulausekkeet useimmiten ovat tyypiltään jotakin muita kuin eksakteja vertailuja (esimerkiksi TCP:n portti väliltä 1024–65535 tai IP-prefiksi 192.168.1.0/24), pakettien luokittelu voidaan nähdä moniulotteisena vaihteluvälien tunnistusongelmana (multidimensional range match) [Wang, 2001].

Esimerkiksi luokittelusääntö, jolla tunnistetaan SSH-yhteydet Tampereen yliopiston UNIX-palvelimelle kielo.uta.fi yliopiston omasta verkosta, muodostuu neljästä vertailukomponentista, jotka on kuvattu oheisessa taulukossa 2. Sääntö voidaan myös nähdä viitosmonikkoon perustuvana liikennevirran tunnistamisena, jossa viidettä komponenttia, lähdeporttia, verrataan implisiittisesti sen kaikkiin mahdollisiin arvoihin (0-65535).

i	Kenttä	Vertailutyyppi	Verrattava arvo
1	Protokolla	Eksakti	6 (TCP)
2	Kohdeosoite	Eksakti	153.1.1.10 (kielo.uta.fi:n IPv4 -osoite)
3	Kohdeportti	Eksakti	22 (SSH)
4	Lähdeosoite	Pisin prefiksi	153.1.0.0/16 (Yliopiston IPv4 -verkko CIDR-muodossa)

Taulukko 2. Esimerkki nelikomponenttisesta luokittelusäännöstä.

4.2. Luokittelumenetelmien arviointi

Vaikka pakettien luokittelu onkin jo arkipäivää runkoverkoissa, haasteita uusien algoritmien kehittämiseksi riittää, sillä luokittelutarpeet muuttuvat yhä monimutkaisemmiksi. Toisaalta verkon nopeuden jatkuvasti kasvaessa myös suorituskykyvaatimukset kasvavat samassa suhteessa. Guptan ja McKeownin [2001] mukaan luokittelumenetelmiä ja -algoritmeja tulee arvioida ainakin seuraavien seikkojen suhteen:

- *Aikakompleksisuus.* Suurilla nopeuksilla toimivat runkolinkit vaativat erittäin tehokkaita algoritmeja. Esimeriksi 10 Gbit/s nopeudella joudutaan pahimmassa tapauksessa käsittelemään 31,25 miljoonaa pakettia sekunnissa (TCP-pakettien minimikoon ollessa 40 tavua).
- *Tilakompleksisuus.* Algoritmin tulee toimia vähäisillä muistimäärillä, jotta se voidaan toteuttaa helposti myös laitteistotasolla nopeaa väli-muistia hyödyntäen.
- *Kyky käsitellä suurikokoisia käytännön luokittelusääntöjä.* Oikeissa verkko-sovelluksissa luokittelusääntöjä voi olla tuhansia ja ne voivat muodostaa monimutkaisia rakenteita. Jotkut teoreettiselta pohjaltaan hyvät algoritmit eivät sovellu juuri lainkaan todelliseen käyttöön.
- *Nopeat päivitykset.* Luokittelusäännöt elävät jatkuvasti, ja luokittelijan tulisi pystyä nopeasti ottamaan käyttöön sääntöihin tehdyt muutokset.
- *Skaalautuvuus otsikkokenttien määrän mukaan.* Luokittelijan tulisi toimia tehokkaasti riippumatta siitä, kuinka monen kentän perusteella luokittelu tehdään.
- *Joustava määrittely.* Luokittelusäännöt tulisi pystyä määrittelemään siten, että niissä voidaan käyttää jokerimerkkejä, loogisia operaattoreita ja erilaisia vertailutapoja.

Luvussa viisi kuvataan lyhyesti joitakin luokittelumenetelmiä ja pyritään arvioimaan niiden hyviä ja huonoja puolia osittain näihin kriteereihin pohjautuen.

4.3. Tilastollinen analyysi

Liikennettä voidaan myös pyrkiä karkeasti luokittelemaan ja tunnistamaan tilastollisella analyysillä havaittujen normaalista poikkeavien muutosten ja trendien perusteella. Tällainen analyysi ei koskaan ole täysin reaaliaikaista, mutta kuten Dübendorfer ja muut [2005] osoittavat, se on erittäin hyödyllinen menetelmä verkon tarkkailuun ja etenkin haittaliikenteen havaitsemiseen. Tilastollista analyysia voidaan käyttää apuna suunniteltaessa luokittelusääntöjä, tai sen pohjalta voidaan jopa automaattisesti luoda uusia sääntöjä, jos analyysillä tunnistetaan ennalta asetetut raja-arvot ylittävää liikennettä.

Dübendorfer ja muut esittelevät avoimen lähdekoodin UPFrame-sovelluksen, jossa analyysin pohjana käytetään reitittimien automaattisesti liikenteestä keräämää NetFlow-dataa. Tällöin luokittelua ei siis tehdä varsinaisesta liikenteestä, vaan kyse on verkkoelementin liikenteestä ottamista *näytteistä*, jotka voivat olla jo valmiiksi tiivistettyjä. Tämä vähentää analyysijärjestelmän suorituskykytarpeita, ja toisaalta yhteen analysoijaan voidaan johtaa useiden verkkoelementtien liikenne ja täten muodostaa kokonaiskuva verkon läpi

kulkevasta liikenteestä. Valmiiksi tiivistettyä näytedataa on myös helppo säilyttää pidemmältä ajalta, jolloin arkistoidusta liikennehistoriasta voidaan tehdä monipuolisia koosteita tai soveltaa siihen erilaisia tiedonlouhintamenetelmiä.

Automaattisia tilastolliseen analyysiin perustuvia menetelmiä on esitetty ratkaisuksi DDoS-hyökkäysten, matoepidemioiden tai roskapostin torjumiseksi. Yhdessä Dübendorfin ja muiden kanssa Sveitsin valtiollisen tutkimusverkon liikennettä tutkineiden Wagnerin ja Platterin [2005] mukaan matoepidemia voidaan tunnistaa säännönmukaisten liikennekuvioiden kasvusta, jotka poikkeavat normaalista Internet-liikenteen entropiasta, joka on Kolmogorov-kompleksista. Koska Kolmogorov-kompleksisuus vaikuttaa siihen, miten hyvin bittijono voidaan pakata [Valmari, 2000], testasivat Wagner ja Platter entropian muutoksia tavallisilla pakkausalgoritmeilla. He havaitsivatkin selvän muutoksen näytedatan pakkautuvuudessa, mikä korreloi matojen leviämisen kanssa.

4.4. Syväluokittelu

Uusien sovellusten ja etenkin haittaliikenteen myötä on noussut tarvetta myös luokittelumenetelmille, jotka kykenisivät tarkempaan analyysiin kuin perinteiset pakettien otsikkokenttien tarkasteluun keskittyvät menetelmät. Usein puhutaan OSI-malliin vedoten *Layer 7* -luokittelusta tai yleisemmin *sisältöperustaisista* luokittelumenetelmistä, joille myös käytetään synonyymina *syväluokittelua*. Käytännössä tämä tarkoittaa, että pyritään aidosti tunnistamaan sovellus, jonka liikenteeseen paketti kuuluu, riippumatta paketin otsikkotiedoista.

Syväluokittelussa otetaan yleensä tarkasteluun yhden tai useamman paketin sisältö liikennevirrasta, tavallisesti ensimmäiset paketit kumpaankin suuntaan kulkevasta liikenteestä. Näin ollen voidaan melko helposti olettaa, että jos ensimmäisen lähteestä kohteeseen kulkevan paketin sisältö alkaa "GET /" ja ensimmäisen vastauspaketin sisältö "HTTP/1.1 200 OK" tai "HTTP /1.1 404 Not Found", kyseessä on HTTP-protokollan mukainen WWW-liikenne, riippumatta siitä, mitä portteja on käytössä. Vaikka esimerkki vaikuttaakin yksinkertaiselta, luokittelusääntö on kuitenkin heti huomattavasti monimutkaisempi kuin paketin otsikkotietoihin perustuva. Sisällöllä ei ole yhtä tarkasti määrättyä muotoa, joten vertailuissa on usein käytettävä paljon enemmän jokerimerkkejä. Lisäksi vertailtavat merkkijonot ovat yleensä pidempiä kuin otsikkokentissä. Sisällöstä tapahtuvaa protokollan tunnistamista voidaan harvoin tehdä luotettavasti vain yhdestä paketista, joten luokittelijan on käytävä läpi useita paketteja ennen kuin voidaan katsoa pakettivirran täsmällisen sääntöön.

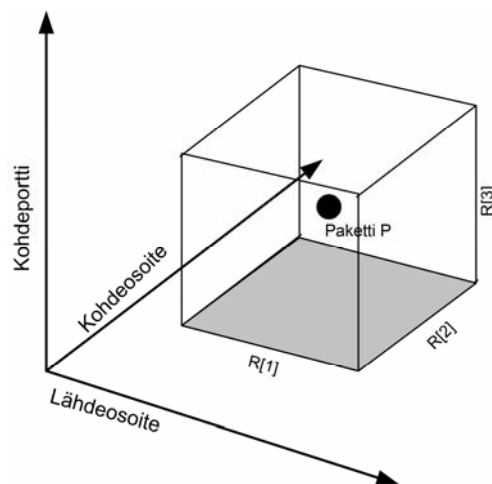
Syväluokittelulla pyritään yleisesti tunnistamaan etenkin haittaliikennettä. Ensimmäisiä sovelluksia olikin palomuuureihin yhdistetty virustorjunta, jossa paketeista etsittiin viruksille tunnusomaisia merkkijonoja. Valitettavasti haittaliikenteen sovelluksia kehittävät pyrkivät aina kiertämään laitevalmistajien ja

verkko-operaattoreiden asettamat suodattimet, joten haittaliikenne naamioituu yhä tehokkaammin. Jos liikenne tunneloidaan salattuna VPN-verkossa, ei operaattorin verkkoelementissä oleva luokittelujärjestelmä pysty tunnistamaan mitään tunnelin sisällä kulkevasta liikenteestä. Myös yhä suurempi osa oikeasta verkkoliikenteestä kulkee salattuna (verkkokaupat, etätyöyhteydet, viranomaisasiointi), joten sisältöön perustuvista tunnistusmenetelmistä on hyötyä jatkosakin lähinnä vain virusten ja matojen tunnistamisessa. Muun tyyppistä haittaliikennettä voidaan paremmin pyrkiä tunnistamaan tilastollisin ja heuristisin menetelmin liikennevirroista [Mirkovic *et al.*, 2005].

5. Erilaisia menetelmiä pakettien luokitteluun

5.1. Geometrinen laskenta

Useaa prefiksiä tai arvoaluetta vertaava luokittelusääntö voidaan nähdä d -ulotteisena kuutiona. Yhden otsikkokentän arvot muodostaa suoran, jolta erotetulla pisteellä, janalla tai puolisuoralla verrattava arvoalue sijaitsee. Kahden otsikkokentän vertaaminen puolestaan muodostaa suorakaiteen (tai vastaavan kaksiulotteisen tason osan), kolmesta syntyy kuutio ja niin edelleen. Jos paketti P , jota geometrisessa mallissa edustaa d -ulotteisen avaruuden piste, on täysin luokittelusäännön komponenttien muodostaman avaruuskappaleen sisässä, pätee luokittelusääntö kyseiseen pakettiin, kuten kuvasta 3 voidaan havaita. Mikäli tämä ehto ei toteudu jokaisen komponentin osalta, ei paketti osu luokittelusääntöön.



Kuva 3. Kohde- ja lähdeosoitteesta sekä kohdeportista muodostettu sääntö esitettyä kuutiona, jonka sisään paketti osuu.

Vastaavia geometrisesti esitettäviä ongelmia löytyy lukuisista muistakin yhteysistä verkkotekniikan ulkopuolelta, joten ongelmaa onkin tutkittu varsin laajasti jo matematiikassa ja tietojenkäsittelyssä. Jos oletetaan, että alueet tai kap-

paleet eivät ole päällekkäisiä, niin voidaan osoittaa, että N :lle kappaleelle pätee $O(\log N)$ aika- ja $O(N^d)$ tilakompleksisuus, kun ulottuvuuksia d on alle kolme, muussa tapauksessa kompleksisuus on $O((\log N)^{d-1})$ ajalle ja $O(N)$ tilalle. Pakettien luokittelussa kuitenkin alueet tai kappaleet yleensä osuvat päällekkäin, jolloin edellä määritelty ei suoraan päde. Lisäksi alueita on yhdessä ulottuvuuksien kanssa niin paljon, että menetelmästä tulee joka tapauksessa liian hidaskäyttö tai tilaa vievä [Gupta and McKeown, 2001].

5.2. Luokittelun tietorakenteet

Yksinkertaisimmillaan luokittelusäännöt muodostavat linkitetyn listan, jonka solmuissa säännöt ovat yleisimmästä tarkimpaan. Pakettia verrataan jokaisen solmun sääntöön, kunnes löytyy sellainen, jonka kaikki komponentit täsmäävät. Linkitetty lista ei kuitenkaan ole kovinkaan tehokas tietorakenne tähän tarkoitukseen, ja luokittelu-aika kasvaa lineaarisesti sääntömäärän lisääntyessä [Gupta and McKeown, 2001].

Hierarkkiset puurakenteet ovat tavallisesti käytössä etuliitteisiin perustuvissa hakumenetelmissä, joissa etsitään pisimmän hakuavaimen mukaan osu-maa. Trie-puut (prefiksipuut) ja radix-puut (PATRICIA-puut) soveltuvatkin erittäin hyvin IP-osoitteiden luokitteluun, sillä IP-verkkoihin perustuvissa hauissa on yleensä kyse juuri pisimmän prefiksin löytämisestä [Opperman, 2005]. Radix-puun aikakompleksisuus on $O(W)$, kun W on etsittävän IP-osoitteen pituus (bitteinä) ja tilakompleksisuus $O(\log^2(N))$, kun N on puussa olevien prefiksien määrä. Runkoverkkotason reitityksessä osoiteprefiksejä voi olla nykypäivänä kuitenkin jo yli 200 000 [Huston, 2006], eikä algoritmia tällöin voida enää pitää kovinkaan tilatehokkaana. Uusilla LC-trie-algoritmeilla voidaan parantaa puun tilakompleksisuutta hyvinkin tehokkaasti, mutta aikakompleksisuus on silti ongelmana, etenkin jos puuta on tarvetta muuttaa tiuhaan [Srinivasan and Varghese, 1999]. Opperman hylkääkin LC-trie-puut reitityskäytössä, koska Internetin runkotasolla reititystaulussa prefiksit muuttuvat jatkuvasti tiheään tahtiin.

Useita muitakin algoritmeja on tutkittu pakettiluokittelun näkökulmasta, mutta useimmissa ongelmaksi muodostuu muistinkulutuksen kasvaminen liian suureksi todellisissa sovelluksissa. Srinivasan ja muut [1998] esittävät ristituloon perustuvan menetelmän, joka skaalautuu helposti mihin tahansa määrään ulottuvuuksia (vrt. geometrinen laskenta), algoritmin aikakompleksisuuden oleessa $O(dt_{RL})$, jossa d on jälleen ulottuvuuksien määrä ja t_{RL} yhden ulottuvuuden vertailun aikakompleksisuus. Valitettavasti ratkaisun tilakompleksisuus on $O(N^d)$, mikä johtaa suureen muistinkulutukseen jo pienelläkin sääntömäärällä.

5.3. Algoritmien yhdistäminen ja heuristiset menetelmät

Todellisessa verkkoliikenteessä on harvoin tilanteita, joissa jokainen perättäinen paketti olisi erilainen ja pitäisi luokitella erikseen. Gupta ja McKeown [2001] osoittavat, että käytännön sovelluksissa tarvitsee siis harvoin mennä huonoimman mahdollisen aika- tai tilakompleksisuuden mukaan, vaan voidaan olettaa, että oikeissa verkoissa käytettävissä luokittelijoissa on tunnistettavissa sellaisia rakenteita ja redundanssia, joita voidaan hyödyntää heuristisesti. Toisaalta on myös tärkeää huomata, että verkon reunoilla ja keskellä olevissa reitittimissä ja palomureissa on aivan erilaiset luokittelusäännöt, mitä ei aina huomioida algoritmeja teoreettisesti tarkasteltaessa [Singh *et al.*, 2003].

Guptan ja McKeownin alun perin vuonna 2000 esittämä HiCuts-algoritmi osoittaa moniulotteisen hakualueen luokittelijan rakennetta hyödyntävän heuristiikan avulla. Jokainen haku johtaa puolestaan HiCuts-puun lehtisolmuun, jossa on pieni määrä sääntöjä, jotka voidaan käydä jo lineaarisesti läpi. HiCuts-algoritmi rakentaa päätöspuun (decision tree) käyttämällä jokaisessa solmussa paikallista optimointia päätöksiin valitessaan seuraavan testattavan ulottuvuuden, ja kuinka moneen osaan ulottuvuus pitää jakaa [Singh *et al.*, 2003]. HiCuts-menetelmää voidaankin pitää päätöspuihin perustuvien luokittelualgoritmien lähtökohtana. Siitä edelleen parannettu HyperCuts-algoritmi nopeuttaa toteutusta sallimalla päätöspuiden solmussa useammankin ulottuvuuden, ja vastaavasti myös useampia lapsisolmuja. Tällä tavalla puun korkeus saadaan pienemmäksi. Niin kauan, kun ositukset tehdään tasakokoisina, voidaan solmussa nopeasti hakea haluttu lapsisolmu pelkästään yhdellä muistinkäsittelyoperaatiolla. Singh ja muut osoittavat empiirisesti, että HyperCuts-algoritmi on kahdesta kymmeneen kertaa tehokkaampi tilakompleksisesti ja puolestatoista viiteen kertaa tehokkaampi aikakompleksisesti HiCutsiin verrattuna.

Cohen ja Lund [2005] osoittavat myös HyperCutsin tehokkuuden todellisessa ympäristössä. He myös tekevät tärkeän havainnon oikeista luokittelusäännöistä: pieni määrä sääntöjä käsittelee valtaosan liikenteestä, ja suurin osa säännöistä on käytössä vain hyvin harvoin tai ei koskaan. Tämä on hyvin lohdullista luokittelujärjestelmien toteutuksen kannalta, sillä kun voidaan huomioida luokittelijan käyttöaste, voidaan keskittyä optimoimaan eniten käytössä olevia sääntöjä esimerkiksi laitteistopohjaisilla toteutuksilla, ja toisaalta harvoin käytetyille riittää keskimääräistä heikompikin suorituskyky. Cohen ja Lund myös uskovat, että vastaava jakauma pätee tulevaisuudessa uusien sovelustenkin kohdalla.

5.4. Laitteistopohjaiset menetelmät

Luonnollisesti verkkolaitteiden valmistajat ovat pyrkineet kehittämään tuotteisiinsa ominaisuuksia, jotka toteuttaisivat luokittelua mahdollisimman tehokkaasti. Reitittimet ja kytkimet pystyvätkin nykypäivänä hämmästyttäviin nopeuksiin. IPv4-liikennettä voidaan reitittää lähes poikkeuksetta linjanopeudella ja suurten reitittimien reittitauluissa voi olla satoja tuhansia prefiksejä. Perinteinen reitin valinta perustuu kuitenkin varsin yksinkertaisiin algoritmeihin (luokittelusäännössä on vain yksi komponentti, kohdeosoite) ja lisäksi laitteissa on nopeat puskurimuistit muistamassa viimeksi tehtyjä reitityksiä, jolloin pake-tin kohde saadaan selville jo hyvin varhaisessa vaiheessa, eikä jokaisen kohdal-la tarvitse tehdä erikseen monimutkaista reitinhakua reititystaulusta.

Vastaavia menetelmiä on toki pyritty soveltamaan myös monimutkaisem-paan luokitteluun. Avainasemassa ovat ASIC-piirit (Application Specific Inte-grated Circuit), jotka nimensä mukaisesti on suunniteltu juuri tiettyä tarkoitu-sta varten. ASIC-piirissä on suoraan piirin rakenteeseen toteutettuna sellaisia perustoimintoja, joita tarvitaan jatkuvasti, ja joiden suoritusajan halutaan ole-van mahdollisimman pieni. Esimerkiksi edellä mainittu reititystiedon puskuro-inti on helppo toteuttaa täysin laitteistotasolla. Näitä toimintoja ASICissa tukee ohjelmoitava prosessoriosuus, joka muistuttaa normaalia tietokoneen yleiskäyt-töistä suorittinta, joka voidaan ohjelmoida suorittamaan mitä tahansa haluttuja tehtäviä [Sardella *et al.*, 2006].

Laitteistopohjaiset ratkaisut ovat kuitenkin kaksiteräinen miekka. Niiden etuna ovat suoraan mikropiireihin toteutetut luokittelualgoritmit, mutta vas-taavasti ne pystyvät toteuttamaan vain näitä ennalta määriteltyjä menetelmiä. Uudet luokittelutavat on toteutettava joko lisäämällä laitteeseen uuden toimin-nallisuuden toteuttavia komponentteja, tai huomattavasti hitaammin ohjel-moimalla toiminnallisuus yleiskäyttöisillä prosessoreilla suoritettavaksi. Mer-kittävän käytännön rajoituksen muodostaa myös tiettyä tarkoitusta varten suunniteltujen erikoislaitteiden hinta yleiskäyttöisiin verrattuna.

6. Yhteenveto

Suurimpien operaattoreiden verkoissa linkkinopeudet runkoverkkotasolla nou-sevat parasta aikaa seuraavaan nopeusluokkaan, joka on 40 Gbit/s. Vaikka lii-kenteen ja pakettien määrä vain nelinkertaistuu edellisestä 10 Gbit/s nopeus-luokasta, on liikenne sisällöllisesti muuttunut huomattavasti monimuotoisem-maksi ja vaikeammin luokiteltavaksi. Haasteet luokittelulle ovat siis moninker-taiset verrattuna muutaman vuoden takaiseen tilanteeseen, jolloin kymmenen gigabitin yhteydet tulivat käyttöön. Toisaalta käytännössä liikennemäärät har-voin ovat linkkinopeuksilla, vaan operaattorit turvaavat yksinkertaisesti palve-

lunlaatua ja verkon suorituskykyä liikenteen luokittelun lisäksi varaamalla huomattavasti enemmän kuin tarpeeksi kapasiteettia tärkeille yhteyksille.

Runkotasolla tehtävää luokittelua liikenteen priorisoimiseksi voidaan myös pitää liian myöhäisenä ratkaisuna, sillä tästä on hyötyä vain tilanteissa, joissa linkin kapasiteetti on jo täynnä tai hyvin lähellä sitä. Tällaisessa tilanteessa siis liikennemäärän yhä kasvaminen johtaa automaattisesti siihen, että vaikka luokittelun avulla voitaisiinkin edelleen välittää haluttua liikennettä ongelmitta, joudutaan jotakin liikennettä kuitenkin pudottamaan. Operaattorit ja ennen kaikkea asiakkaat mieltävät asian usein niin, että jos luokittelua joudutaan tekemään suorituskyvyn vuoksi, verkossa *on jo ongelma*, ja luokittelu ei poista sitä, vaan on vain *tilapäinen* ratkaisu kapasiteetin kasvattamista odoteltaessa.

On kuitenkin ilmeistä, että luokittelu- ja suodatusvaatimusten monimutkaistuesssa niiden toteuttaminen sijoittuu entistä enemmän verkon reunoille, jossa yhteysnopeudet ovat pienempiä ja toisaalta riski linkkien ruuhkautumiselle suurempi. Luokittelualgoritmien tutkimuksessa ei ole myöskään sanottu vielä viimeistä sanaa, vaan työtä tehdään jatkuvasti ympäri maailmaa useiden tutkimusryhmien toimesta. Rinnakkaislaskennan yleistyessä ja muistin määrän ja hakunopeuksien kasvaessa voidaan soveltaa uudenlaisia algoritmeja perinteisten ongelmien ratkaisuun. IPv6:n laajempi osoiteavaruus asettaa myös aivan omat haasteensa luokittelun toteuttamiseen.

Perinteisen WWW:n, sähköpostin ja tiedostojen siirron rinnalle IP-liikenteen sekaan näyttävät lopulta vuosien odottelun jälkeen siirtyvän myös puhelut ja liikkuva kuva. Koteihin tulevien, niin sanottujen viimeisen mailin yhteyksien nopeudet nousevat jatkuvasti uusien laajakaistatekniikoiden myötä, kuten myös mobiililaitteisiin saatavien langattomien pakettidatayhteyksien. Ei ole mitään syytä olettaa, etteikö vaivihkaa nopeasti kaikkialle levittäytyneen ja siten suoraan valmiina olevan IP-verkon päälle siirtyisi jatkossa entistä enemmän palveluita ja muodostuisi uusia, kuten tähänkin asti on tapahtunut. Informaatioyhteiskunnan arkipäiväistyessä yhä suurempi osa elämästä on sidoksissa miljoonia paketteja sekunnissa välittäviin IP-verkkoihin. Ilman luokittelua ja sen perusteella tehtäviä suodatuksia tai palvelunlaatusääntöjä tärkeät työ- tai yksityisasioiden palaset hukkuisivat liian helposti tähän valtavaan virtaan.

Viiteluettelo

[Cohen and Lund, 2005] Edith Cohen and Carsten Lund, Packet classification in large ISPs: Design and evaluation of decision tree classifiers. In: *Proceedings of the 2005 ACM SIGMETRICS*, 73-84.

[Dübendorfer *et al.* 2005] Thomas Dübendorfer, Arno Wagner and Bernharnd Plattner, A framework for real-time worm attack detection and backbone

- monitoring. In: *Proceedings of First IEEE International Workshop on Critical Infrastructure Protection (IWCIP 2005)*, 10.
- [FICIX, 2007] Finnish Communication and Internet Exchange ry., <http://stats.ficix.fi/> (16.5.2007).
- [Gupta and McKeown, 2001] Pankaj Gupta and Nick McKeown, Algorithms for packet classification. *IEEE Network* **2** (15), 24-32, 2001.
- [Gupta and McKeown, 1999] Pankaj Gupta and Nick McKeown, Packet classification on multiple fields. In: *Proceedings of ACM SIGCOMM '99*, 147-160.
- [Huston, 2006] Geoff Huston, The BGP Report for 2005. Internet Society Publications, ISP Column, 2006. Available as <http://ispcolumn.isoc.org/2006-06/bgpupds.html> (18.5.2007).
- [Mirkovic *et al.*, 2005] Jelena Mirkovic, Max Robinson, Peter Reiher, George Oikonomou, Distributed defence against DDoS attacks. University of Delaware CIS Department Technical Report CIS-TR-2005-02, 2005. Available as http://www.nsl.cis.udel.edu/Project%20Pages/DefCOM/Publications/udel_tech_report_2005-02.pdf (15.5.2007).
- [Opperman, 2005] André Opperman, Optimizing the FreeBSD IP and TCP stack. In: *EuroBSDCon 2005*. Available as [http://people.freebsd.org/~andre/Optimizing the FreeBSD IP and TCP Stack.pdf](http://people.freebsd.org/~andre/Optimizing%20the%20FreeBSD%20IP%20and%20TCP%20Stack.pdf) (17.5.2007).
- [Sardella *et al.*, 2006] Alan Sardella, Muralidhar Devarasetty, Julian Eccli and Daniel Kharitov, Efficient scaling for multiservice networks. Juniper Networks, Inc. 2006. Available as http://www.juniper.net/solutions/literature/white_papers/200207.pdf (14.5.2007).
- [Singh *et al.*, 2003] Sumeet Singh, Florin Baboescu, Georg Varghese and Jia Wang, Packet classification using multidimensional cutting. In: *Proceedings of ACM SIGCOMM '03*, 213-224.
- [Srinivasan and Varghese, 1999] V. Srinivasan and George Varghese, Fast address lookups using controlled prefix expansion. *ACM TOCS* **17**, 1 (1999), 1-40.
- [Valmari, 2000] Antti Valmari, *Mitä on Kolmogorov-kompleksisuus?* Tampereen teknillinen yliopisto, ohjelmistotekniikan laitos, 2000. <http://www.cs.tut.fi/~ava/kirjoitelmia/Kolmogorov.html> (18.5.2007).
- [Wagner and Platter, 2005] Arno Wagner and Bernhard Plattner, Entropy Based Worm and Anomaly Detection in Fast IP Networks. In: *14th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises (WET ICE 2005)*, 127-177.

[Wang, 2001] Zheng Wang, *Internet QoS: Architectures and Mechanisms for Quality of Service*. Morgan Kaufmann Publishers, 2001.

[Wikström, 2006] Mika Wikström, OSI-malli, protokollat ja TCP/IP. <http://www.mit.jyu.fi/wikstrom/opetus/itkp104/luennot/itkp104-8.pdf> (15.5. 2007).

Teknologian turhauttavuus

Santtu Mansikkamaa

Tiivistelmä.

Tämä tutkielma on kirjallisuuskatsaus tekijöihin, jotka aiheuttavat turhautumista tietokoneiden käyttäjissä. Turhautuminen vaikuttaa meihin kaikkiin psykologisesti eikä sen vaikutusta aina edes huomaa. Turhautumisesta on käyttäjälle vain haittaa, ja pidemmällä aikavälillä se voi jopa aiheuttaa psyykkisiä ongelmia. Tässä tutkielmassa selvitetään, mitkä asiat aiheuttavat turhautumista, miten sitä tutkitaan ja mitataan, kuinka se vaikuttaa käyttäjään sekä kuinka sen syntymistä voitaisiin ennalta ehkäistä.

Avainsanat: Turhautuminen, emootiot, vuorovaikutteinen teknologia.

CR-luokat: A.2, I.m, J.4

1. Johdanto

Lazar ja muut (2005) määrittelevät turhautumisen siten, että tietokone tekee jotain odottamatonta sellaisella tavalla, joka ärsyttää käyttäjää ja estää häntä saavuttamasta tavoittelevansa päämäärää. Useat asiat vuorovaikutteisessa teknologiassa ärsyttävät käyttäjää aiheuttaen ylikuumenemista. Näitä ovat muun muassa epämiellyttävät viiveet, yhteensopimattomat tiedostotyypit sekä virheilmoituksia muistuttavat ponnausikkunassa olevat mainokset. Nämä ovat usein sekä harhaanjohtavia että turhauttavia. Dialogit jotka ovat epäselvästi tai monimutkaisesti kirjoitettuja, voivat myös johtaa tehdyn työn menettämiseen. (Lazar et al., 2005)

Reevesin ja Nassin (1996) mukaan ihmisillä on taipumus olla vuorovaikutuksessa eri medioiden kanssa sellaisilla tavoilla, jotka ovat meille luonnollisia ja sosiaalisia. Reagoimme siis samaamme tietoon samoin kuin ollessamme vuorovaikutuksessa toisen ihmisen kanssa. Nykyisellään tietokoneiden kyky kommunikoida käyttäjiensä kanssa on hyvin rajallista: se rajautuu pelkästään siihen, kuinka ohjelmissa on suunnitteluvaiheessa oletettu niihin suhtauduttavan, eikä käyttäjän todellisia emootioita voida ottaa huomioon kuin harvoissa tapauksissa. Tietokone, joka pystyisi tunnistamaan ja tuottamaan affektiivisia toimintoja, lisäisi koneen vuorovaikutteisia kykyjä suunnattomasti. (Scheirer et al., 2002)

Monia erilaisia menetelmiä parempaan vuorovaikutukseen on jo kehitteillä, mutta kaikilla on yhteisenä ongelmana tunnistaa käyttäjän emootiaalinen tila monista yhtä aikaa tapahtuvista muuttujista. Koneen täytyy tunnistaa, mitä

käyttäjä on tekemässä ja kuinka hän reagoi saatuun palautteeseen ja yhdistää näitä tietoja kontekstin antamaan tilanteeseen. Hyvä käyttöliittymäsuunnittelu tekisi laitteiden käytön niin yksinkertaiseksi, että normaali käyttäjäkin osaisi niitä käyttää ilman suuria ponnisteluja. (Scheirer et al., 2002; Lazar et al., 2005)

2. Emootioiden tunnistaminen

Emootioiden tunnistaminen on tärkeää tutkittaessa käyttäjän vuorovaikutusta tietokoneen kanssa. Monissa tutkimuksissa mitataan koehenkilön emootioita seuraamalla tämän fysiologista tilaa. Esimerkiksi Scheirer ja muut (2002) vertailivat käyttäjän reaktioita viallisesti ja oikein toimivaan tietokoneen hiireen. He esittelivät menetelmän, jossa käyttäjän pitää suorittaa erilaisia visuaalisia ongelmia tietokoneella olevassa pelissä. Käyttäjän työskentelyä vaikeutetaan siten, että hiiri ei joka kerta joko vastaa sille annettuun syötteeseen ollenkaan tai vastaa viiveellä. Käyttäjää siis yritetään turhauttaa tahallaan, heille sitä kuitenkin kertomatta. Koehenkilön fysiologista tilaa mitataan samalla kun tietoa kontekstista, pelin edistymisestä ja hiiren käytöstä seurataan ja tallennetaan. Fysiologinen tila puolestaan kertoo käyttäjän emootioista.

Ihmisen emootioita voidaan tunnistaa ihon sähkön johtavuudesta, sydämen sykkeestä, lihasten jännittyneisyydestä, verenpaineesta, hengityksestä, aivosähkökäyrästä (EEG), pupillien koosta, kasvojen ilmeistä (ml. lihasjännitteet), haastatteluista, käyttäjän eleistä sekä puheesta. Erilaisia laitteita on kehitetty näiden tunnistamiseen. Esimerkiksi normaaleja lääketieteen käyttämiä sensoreita on asennettu koruihin, kenkiin, vaatetukseen, silmälaseihin ja jopa tietokoneen hiiriin. (Ark et al., 1999; Marrin and Picard, 1998; Picard and Healey, 1997; Scheirer et al., 1999). Yleensä näitä saatuja tuloksia kategorioidaan kaksiulotteisella koordinaatistolla, jossa toisena akselina on ärsykkeen voimakkuus ja toisena sen miellyttävyys (positiivinen/negatiivinen). (Scheirer et al., 2002)

Wilsonin ja Eggemeierin (1991) mukaan on kannattavaa käyttää kahta tai useampaa emootioiden tunnistusmenetelmää yhtä aikaa, jotta saataisiin tarkempia ja laadukkaampia tuloksia. Jokainen menetelmä on eri osa-alueella herkempi kuin toinen ja näiden yhdistäminen useammalla mittausmenetelmällä luo yhdistettynä vahvan metodin emootioiden tunnistamiselle. (Scheirer et al., 2002)

3. Turhautuminen

Tehdyn työn menettäminen ohjelman kaatuessa lienee yksi tunnetuimmista turhautumista aiheuttavista tekijöistä vuorovaikutteisessa mediassa. Turhautuminen vaikuttaa vahvasti tapamme saavuttaa päämäärämme. Lawsonin (1965)

mallissa turhautuminen lisää koe-eläinten taipumusta reagoida aggressiivisesti, esimerkiksi rotta lisää voimankäyttöään jos jokin esine on sen ja (ruoka)palkinnon välissä. Riseberg ja muut (1998) kertovat testihenkilöiden turhautuneen, kun heitä estettiin saamasta hyvää tulosta tietokonepelissä. (Scheirer et al., 2002)

Ihmiset voivat reagoivat emotionaalisesti kykenemättömyyteensä saavuttaa päämääräänsä. Yksi reaktio näistä on turhautuminen. Käyttäjät tuntevat usein jopa raivoa turhoutuessaan, jolloin he haluavat hajottaa tietokoneen ja heittää sen osia ikkunasta (Norman, 2004). Useimmat käyttäjät kertovat kiroilleensa ääneen tietokoneelle turhoutuessaan. Turhautuminen voi johtaa huonompaa työmuokavuuteen, kohonneeseen verenpaineeseen sekä lihasten jännitteisyyteen (Murrell and Sprinkle, 1993). Turhautumisen aiheuttama aggressio pohjautuu ihmisluonteeseen, ulkoiseen ärsykkeeseen sekä kulttuurisidonnaiseen opittuun aggressiomalliin sekä mallin hallinta.

Lazarin ja muiden (2005) tekemässä tutkimuksessa turhauttavia tapauksia koettaessa ne koettiin useimmiten vahvasti turhauttaviksi ja vain harva raportoitu tapaus oli ilmoitettu lievästi turhauttavaksi. Noin 75% raportoiduista tapauksista oli ilmoitettu turhauttaviksi ja lähes puolet tapauksista oli ilmoitettu olevan kahdeksan tai yhdeksän asteikolla yhdestä yhdeksään, yhdeksikön ollessa todella turhauttava ja ykkösen ollessa ei juurikaan turhauttava. (Fortman, 2005; Lazar et al., 2005; Scheirer et al., 2002)

3.1. Turhautumiseen vaikuttavat tekijät

Amselin (1992) mukaan behavioristiset periaatteet esittävät palkinnon saamisen viivästyttämistä erääksi turhautumisen syyksi. Perinteisessä koemallissa tätä voitaisiin toteuttaa ruokapalkinnon viivästyttämisenä opetetun eläimen painaessa oikeaa nappia. Tätä mallia on käytetty analogiana Scheirerin ja muiden (2002) tutkimuksessa siten, että tietokoneen hiiren napin painamista käytetään eteenpäin pääsemiseen (napin painaminen palkinnon saamiseksi), mutta välillä hiirellä annettujen komentojen ja itse tapahtuman välillä on viive (esteitä matkalla palkinnolle), jonka uskotaan aiheuttavan turhautumista. Palkinnoksi oltiin päätetty antaa nopeimmalle tehtävien suorittajalle 100 dollaria. Tällä pyrittiin jäljittelemään tosielämän johonkin päämäärään pääsemisen saavuttamista. Näitä kuvataan usein välittömäksi palautteeksi ja käyttäjän kontrolliksi vuorovaiikutteisessa teknologiassa. (Scheirer et al., 2002)

Turhautuminen esiintyy monina erilaisina ilmiöinä käyttäjässä. Scheirerin ja muiden (2002) mukaan osa koehenkilöistä odotti hiiressä olleen viiveen ja vain harvoin koettivat toistaa toimintiaan, ja täten eivät teoillaan osoittaneet minäänlaista reaktiota heille annettuun ärsykkeeseen. Kuitenkin 19 henkilöä 24:stä

toisti toimintonsa useamman kerran viiveen aikana. Eräs henkilö toisti hiiren klikkauksen jopa kuusitoista kertaa, jotta vihdoinkin sai haluamansa tuloksen, vaikka tulokseen pääsemiseen tarvittiin vain ensimmäinen klikkaus ja viiveen ylitse odottaminen. Scheirer ja muut (2002) huomasivat myös, että jos ensimmäinen hiiren painallus ei heti näkynyt näytöllä, koehenkilöt saattoivat kokeilla muita painikkeita nähdäkseen aiheutuuko niitä painamalla mitään toimintoa.

Locken (1996) mukaan tutkimus päämäärän saavuttamisesta osoittaa, että päämääräsidonnaisuus korreloi vahvasti suoritukseen ja liittyy kahteen tekijään. Näistä toinen on tehtävän tai tuloksen tärkeys ja toinen on usko päämäärän saavutettavuudesta. Henkilö omistautuu päämäärälle, kun se on hänelle tärkeä ja hän uskoo sen olevan mahdollista saavuttaa. Lazarin ja muiden (2005) tutkimuksesta käy ilmi, että vain kolme tekijää korreloi vahvasti turhautumisen kanssa: menetetty aika, menetetyn työn korjaaminen ja tehtävän tärkeys. Korrelaatio tuli esille kummassakin tutkitussa ympäristössä, opiskelijoilla ja työpaikoilla. Lazar ja muut (2005) huomasivat myös, että käyttäjäryhmästä huolimatta suurimmat turhautumista aiheuttavat tekijät ovat tehdyn työn tärkeys, menetetyn työn tekemiseen mennyt aika sekä virheen aiheuttaman menetyksen korjaamiseen mennyt aika. Tutkimuksen mukaan tietokoneen käyttökokemus, koulutus tai ahdistuneisuus eivät vaikuta subjektiiviseen turhautumisen tunteeseen.

Teoria turhautumisesta ennustaa tehtävän keskeytymisen aiheuttavan yksilölle turhautumista. Turhautumisen tasoon vaikuttavat tapahtuma ja yksilöllinen tekijä. Käyttäjän päämäärään omistautuneisuus sekä keskeytyksen vakavuus vaikuttavat tapahtuman vakavuuteen. Yksilöllisen tekijän vahvuus taas koostuu tietokoneen käyttökokemuksesta, mielialasta ja muista psykologisista tekijöistä, sekä kulttuurisista ja sosiologisista vaikutuksista henkilöön. (Lazar et al., 2005)

3.2. Turhautumisen ennaltaehkäisy

Lazarin ja muiden (2005) tutkimuksessa 42% tutkimukseen osallistuneista ilmoitti olleensa vihainen tietokoneelle, kun vain 4% ilmoitti olleensa vihaisia itselleen. Kontrollin menetystä tunsivat 12% käyttäjistä joissain tilanteissa. Vain 23% ilmoitti uskovansa pystyvän korjaamaan ongelman. (Lazar et al., 2005)

Seligmanin *Learned Helplessness* -teorian periaate on, että mitä enemmän henkilö tuntee hallitsevansa työskentely-ympäristöään, sitä iloisemmaksi hän tuntee itsensä. Kun taas henkilö tuntee itsensä turhautuneeksi tai vihaiseksi se luultavasti johtuu asioista joita hän ei ole voinut hallita. (Spolsky, 2000)

Seligmanin teoriasta ja Lazarin ja muiden (2005) tutkimuksesta käy selvästi ilmi, että käyttäjä kokee turhautumisen olevan vahvasti riippuvainen tietoko-

neesta (käyttöliittymästä) ja muista käyttäjästä riippumattomista seikoista. Tämän vuoksi myös turhautumisen ennaltaehkäisemisessä avainasemassa on toimivan ja käyttäjäystävällisen käyttöliittymän suunnittelu.

Käyttäjän emotionaaliseen tilaan on mahdollista vaikuttaa monenlaisella vuorovaikutteisella teknologialla. On osoitettu, että tietokoneen käyttäjät ovat varsin avoimia ottamaan vastaan imartelua tietokoneelta. Huolimatta siitä että käyttäjille on kerrottu tietokoneen olevan huono arvioimaan heidän suoritus- taan, on imartelulla ollut tutkimuksissa positiivisia vaikutuksia käyttäjien emootioihin ja suoritukseen (Fogg and Nass, 1997). Positiivisella palautteella voidaan normalisoida käyttäjän emotionaalista tilaa; tätä voidaan käyttää esimerkiksi erilaisten testien välillä emootioita tutkittaessa. (Aula and Surakka, 2002; Sheirer et al., 2002)

4. Lopuksi

Vaikka koemmekin itsemme kovin kehittyneenä lajina, ei meidän psyykinen maailmamme ole paljoa kehittynyt kivikaudelta, vaikka käyttämämme työkalut ovat moderneja. Monet osoittavat meidän edelleen reagoivan kovin primitiivisillä tavoilla ärsykkeisiin. Yritämme alitajuisen ärsykkeen johdosta kurittaa tottelematonta konetta inhimillisin keinoin; henkisin (huutaminen, kiroileminen, haukkuminen) ja fyysisin painostuskeinoin (lyöminen, heittäminen) jotta kone tekisi vihdoin niin kuin siltä odotamme. Halutun toiminnon tapahtuessa koneeseen taasen reagoidaan kuten työkaluun, tunteettomaan materiaan. Tämän johdosta pitäisi mielestäni keskittyä enemmän työkalujen käytettävyyteen, tehdä enemmän tutkimusta ihmisen psykologisesta olemuksesta ja sen kautta pyrkiä tekemään käyttöliittymistä sellaisia, että ne toimivat odotuksiemme mukaan.

Tottelemattomuus ärsyttää ja siksi vuorovaikutteisessa teknologiassa pitäisi mielestäni enemmän keskittyä kehittämään ja suunnittelemaan ohjelmia ihmisille, eikä käyttötarkoitukselle. Ohjelmia ja laitteita tulisi kehittää siten, että ennen toteutusta suunnitellaan, ennakoidaan, toteutetaan ja testataan niitä käyttäjillä, jotka oikeastikin tulisivat ohjelmaa tai laitetta käyttämään. Analyysi ja haastattelut käyttökokemuksista sitten valjastettaisiin uuden suunnittelu- ja toteutusiteroinnin pohjaksi, ja näin tuotetta kehitettäisiin loppukäyttäjien kanssa yhteistyössä sellaiseksi, että muutkin kuin kehittäjät sitä osaisivat käyttää intuitiivisesti. Tämä olisi mielestäni hyvin tärkeää käytettävyyden kannalta. Modulaarisesti kehittäen ei tämä mielestäni lisäisi kehityskustannuksia, mutta lisäisi käyttönoton helpoutta sekä vähentäisi virheidensyntymisen riskiä ja tätä kautta lisäisi tuloksen kasvua. Lazarin ja muiden (2005) tutkimuksen mukaan tehdystä työstä kolmannes tai jopa puolet on hukkaan heitettyä aikaa

liian vaikeiden käyttöliittymien tai menetetyin työn johdosta. Käyttöliittymien pitäisi siis olla käyttäjille intuitiivisesti selkeitä.

Itsekin olen menettänyt tekemääni työtä huonosti suunnitellun käyttöliittymän johdosta. Ollessani Egyptissä matkalla digitaalisen kamerani muistikortilta loppui tila juuri mielenkiintoisella retkellä ja halusin lyhentää aiemmin ottamaani pitkää videota. Vaikeaselkoinen kysymys valitun alueen poistamisesta osoittautuikin valitun alueen säästämiseksi ja valitsemattoman poistamiseksi, jolloin kaikki parhaat palat oli menetetty. Tällaiset tapaukset turhauttavat käyttäjää paljon. Työoloissa monet tällaiset pienetkin asiat kertautuvat (Spolsky, 2000) ja kasautuessaan aiheuttavat turhautumista ja masentuneisuutta. Nämä taasen laskevat työpanosta ja tuottavuus kärsii.

Lopuksi haluaisin tuoda esille Spolskyn (2000) esittämän aksiooman: “*A user interface is well-designed when the program behaves exactly how the user thought it would.*”

Viiteluettelo

- Aula, A. & Surakka, V. (2002) Auditory Emotional Feedback Facilitates Human-Computer Interaction. In Faulkner, X., Finlay, J., & Détienne, F. (Eds.) *People and Computers XVI: Memorable Yet Invisible, Proceedings of HCI 2002*, 337-349. London, Springer-Verlag.
- Bessière K, Newhagen J. E., Robinson J. P. & Shneiderman B (2006). A model for computer frustration: the role of instrumental and dispositional factors on incident, session, and post-session frustration and mood. *Computers in Human Behavior* 22, Issue 6 , November, 941-961.
- Fogg, B. J. & Nass, C. (1997), Silicon Sycophants: The Effects of Computers that Flatter, *International Journal of Human-Computer Studies* [Electronic version] 46(5), 551-61.
- Fortman, Bas de Gaay (2005). Violence among peoples in the light of human frustration and aggression. Studie-en Informatiecentrum Mensenrechten (SIM), Faculty of Law, Universiteit Utrecht, The Netherlands. *European Journal of Pharmacology*. 526, 1-3, 2-8
- Hone, K. (2006) Empathic agents to reduce user frustration: The effects of varying agent characteristic. *Interacting with Computers*. 18, 2 , 227-245.
- Lazar, J., Jones, A., Hachley, M. & Shneiderman, B., 2005. Severity and impact of computer user frustration: A comparison of student and workplace users. *Interacting with Computers*, 18, 2, 187-207.
- Partala, T. & Surakka, V. (2004) The effects of affective interventions in human-computer interaction. *Interacting with Computers*, 16, 295-309.

- Scheirer, J. Fernandez, R. Klein, J. & Picard, R. (2002) Frustrating the user on purpose: a step toward building an affective computer. *Interacting with Computers*. 14, 2 , 93-118.
- Spolsky, J. (2000) User Interface Design for Programmers. Chapter 1: Controlling Your Environment Makes You Happy. <http://www.joelonsoftware.com/uibook/chapters/fog0000000057.html>. Checked 15.5.2007.

Kuvadatan pakkausalgoritmeista

Iikka Mattila

Tiivistelmä.

Kuvadatan pakkausalgoritmeja on sekä informaatiota hävittäviä että sen säilyttäviä. Informaatiota hävittävät käyttävät usein hyväksi *ihmisen näköjärjestelmän* heikkouksia ja ominaisuuksia. Tässä tutkielmassa esittelen muutamia esimerkkejä tällaisista ominaisuuksista ja niiden hyödyntämistä eri algoritmeissa. Esittelen myös yleistä informaatioteoriaa ja pakkausalgoritmiikkaa siltä osin kun se kuvadatan pakkaukseen liittyy. Lisäksi käyn läpi eri algoritmien yhdistelyä olemassa olevissa formaateissa.

Avainsanat ja -sanonnat: Kuva, pakkaus, tiivistys, entropiakoodaus, häviöllinen pakkaus, häviötön pakkaus, ihmisen näköjärjestelmä

CR-luokat: E.4, H.1.1, I.4.2

1. Johdanto

Kuvallisen datan määrä ympärillämme kasvaa ja siten myös tarve tallettaa ja siirtää kuvadataa kasvaa. Pitkään, noin 1980-luvulle, kuvanpakkauksessa käytettiin perinteisen informaatioteorian oppeja ja pakattaessa poistettiin toisteisuutta ja redundanssia [Reid et al., 1997]. Näiden niin sanottujen ensimmäisen sukupolven pakkausalgoritmien tilalle nousi uusia pakkausalgoritmeja, jotka käyttävät hyväksi tietoa ihmisen visuaalisesta hahmottamiskyvystä. Toisen sukupolven pakkausalgoritmit hävittävät informaatiota, mutta erottamalla hahmottamiselle olennaisen informaation epäolennaisesta ne voivat käyttää pakkausalgoritmeja, jotka keskittyvät nimenomaan olennaisen informaation laadukkaaseen pakkaamiseen.

Tämä kahtiajako on kenties perustavimmanlaatuinen erottelu pakkausalgoritmeissa – häviöllinen ja häviötön pakkaus. Näiden välillä ei ole varsinaista parempi – huonompi -suhdetta, vaan kutakin tarkoitusta varten toinen lähestymistavoista on toista parempi. Häviötöntä pakkausta tarvitaan ja käytetään usein esimerkiksi teknillisissä piirustuksissa, stilisoiduissa logoissa ja sarjakuvisa – kuvadatatassa, joka ei ole lähellä ihmisen luonnossa näkemiä asioita. Häviöllinen pakkaus soveltuu taasen esimerkiksi valokuviin, näissä häviöstä syntyvät pakkausvirheet eivät ihmisen silmään näy yhtä helposti.

Tässä artikkelissa esittelen muutamia algoritmisia lähestymistapoja kuvadataan pakkaukseen, kerron mihin ne pohjautuvat ja mihin ne soveltuvat.

2. Peruskäsitteistöä kuvadatasta

Termillä kuvadata viitataan dataan, joka on tarkoitettu tietokoneen näytön tai vastaavan laitteen näyttämäksi kaksiulotteiseksi kuvaksi. Kuva koostuu kuvapisteistä eli *pikseleistä*, joista jokaisella on oma värinsä. Näytöt muodostavat kuvan käyttämällä additiivista värinmuodostusta, jossa väri muodostuu kolmesta väriavalosta, punaisesta, sinisestä ja vihreästä. Näitä komponentteja kutsutaan myös *kanaviksi*. Näiden kolmen värin muodostama niin sanottu *rgb-väriavaruus* on yleinen nimenomaan kuvaa näytölle tuottaessa, mutta esimerkiksi tiedostoja tallettaessa väri-informaatiota voidaan käsitellä myös toisissa väriavaruuksissa, kuten esimerkiksi myöhemmin esiteltävässä JPEG-formaatissa tehdään. Väriavaruusmuunnoksista saattaa kuitenkin seurata informaation katoamista.

Tämän hetken tekniikassa yleisintä on käyttää 256-portaista asteikkoa jokaiselle puna-, viher-, sinivärikomponentille, jolloin syntyy $256^3 = 16\,777\,216$ värisävyä. Tällaisten kuvien *bittisyvyys* on 24 bittiä, eli jokaisen pikselin värin määrittelee 24 bittiä. Joissain käyttöyhteyksissä on tarkoituksenmukaista tallentaa kuvaan läpinäkyvyyskanava, niin sanottu *alfakanava*, vastaavasti 8 bitin tarkkuudella. Tällöin kuvien bittisyvyys olisi 32 bittiä.

Eri kuvissa voi luonnollisesti olla vaihteleva määrä pikseleitä ja kuvat voivat olla siis erikokoisia. Tätä pikselileveyden ja -korkeuden tuloa kutsutaan *resoluutioksi*. Resoluutiolle ei vaihtelevista käyttötarkoituksista johtuen ole mielekäästä asettaa standardeja.

Tallennettaessa kuvia tietokoneen tiedostoihin täytyy siis tallettaa jokainen pikseli ja sen väri. Kaksiulotteinen kuvadata muutetaan yksiulotteiseksi bittivirraksi, siis muun datan kaltaisiksi tavuiksi, eli merkeiksi.

3. Häviöttömistä algoritmeista

Koska kuvadata on muutettavissa yksiulotteiseksi tavujen virraksi, voidaan sitä käsitellä kuin mitä tahansa dataa ja pakata sitä geneerisillä pakkausalgoritmeilla. Tämä ei kuitenkaan ole tehokasta, sillä kuvalliseksi tarkoitettussa datassa on tyypillisiä piirteitä ja ominaisuuksia, joita voitaisiin pakata kuviin erikoistetuilla algoritmeilla.

3.1. Entropiakoodauksesta

Pakattavaan tietoon liittyy kiinteästi käsite siitä, kuinka satunnaista tai ennalta arvattavaa tieto on. Tätä tiedon ominaisuutta kutsutaan *entropiaksi*. Tiedossa, jossa on vain yhtä tiettyä merkkiä, on alhainen informaation sisältö ja siten sen entropia on nolla. Täysin satunnaisella tiedolla, joka koostuu n erilaisesta merkistä $\{s_1, s_2, \dots\}$, entropia on $\log_2 n$. Yleensä kuitenkin pakattavan tiedon satunnaisuus sijoittuu näiden kahden ääripään väliin, jolloin sen entropia on

$$H = \sum_{i=1}^n p_i \log_r(1/p_i),$$

missä p_i on merkin s_i esiintymistodennäköisyys. Pakattavan tiedon entropia määrää myös teoreettisen rajan pakkaustehokkuudelle, eikä täysin satunnaista tietoa voikaan pakata pienempään tilaan. [Aho et al., 2000]. Kuvailen seuraavaksi joitain algoritmeja, joilla tiedosta voidaan etsiä toistuvuutta ja säännön mukaisuutta, eli entropiakoodata datan.

3.1.1 Juovakoodaus

Juovakoodaus (run-length encoding, RLE) on tapa pakata perättäisenä toistuvaa merkkiä kertomalla (i) montako kappaletta merkkiä tulee ja (ii) mitä merkkiä. Tätä yksinkertaista algoritmia käyttää esimerkiksi PCX-formaatti. Algoritmi ei ole erityisen tehokas.

3.1.2 Huffman-koodaus

Huffman-koodauksen periaatteena on laskea merkkien esiintymistiheydet ja muodostaa näiden lukumäärien avulla binääripuu, jossa useimmin esiintyvät merkit on lähellä puun juurta ja harvaan esiintyvät merkit kaukana siitä. Kun alkuperäisestä datasta korvataan merkkien esiintymät lyhyemmällä bittijonolla, jotka kertovat, kuinka puussa edetään, saadaan lopputuloksena pienempi tiedosto. Koodauksen kehitti David A. Huffman jo vuonna 1952, mutta se on iästään huolimatta edelleen sangen yleisessä käytössä. Huomattavaa on, että perusmuodossaan Huffman-koodauksen tulee lukea koko syöte, ennen kuin se voi laskea merkkien esiintymistiheyksiä. On kuitenkin kehitetty myös dynaamisia versioita Huffman-koodauksesta, jossa syötettä voidaan pakata sitä mukaa, kun sitä luetaan.

3.1.3 Lempel-Ziv -koodaus

Abraham Lempel ja Jacob Ziv kehittivät 1977 pakkausalgoritmin, joka tunnetaan LZ77-algoritmina. Algoritmi on niin sanottu *sanakirjapakkaus*. Se siis etsii datasta toistuvia merkkijonoja eli sanoja, antaa näille sanoille lyhyemmän koo-

din ja korvaa sanat koodeilla. LZ77-algoritmin toimintaperiaate on seuraava [Aho et al., 2000]:

1. Lue syötteestä merkki x .
2. Jos merkkiä ei ole aiemmin luettu, lisää koodisanaluetteloon koodisana $(0, x)$. Jatka kohdasta yksi.
3. Koska merkki on jo luettu, lue seuraava merkki. Jos edellisen merkin ja juuri luetun merkin yhdessä muodostama sana on koodisanaluettelossa, luetaan seuraavakin merkki. Näin jatketaan muodostettavaa sanaa pidentäen kunnes muodostuu sana, jota ei koodisanaluettelossa vielä ole.
4. Lisätään luetteloon koodisana (i, y) , missä i on sen koodisanan järjestysnumero luettelossa, joka oli viimeinen luettu koodisana ja y viimeksi luettu merkki.
5. Jatka kohdasta yksi, kunnes syöte on luettu läpi.
6. Tallenna koodisanaluettelo ja koodisanoilla korvattu tieto.

Seuraavana vuonna Lempel ja Ziv julkaisivat parannetun LZ78-algoritmin. Vuonna 1984 Terry Welch julkaisi tunnetun LZW-algoritmin, joka sisältää joitain parannuksia alkuperäisiin LZ-algoritmeihin. Muun muassa GIF-formaatti käyttää LZW-algoritmia. LZ-algoritmeista on olemassa lukuisia muunnelmia.

3.2. Tiedon ennakointi

Vaikkei häviöttömissä algoritmeissa voidakaan jättää informaatiota pois, voidaan niissäkin tehdä tiettyjä yleistyksiä tavallisesti pakattavana olevien kuvien sisällöistä.

Häviötön PNG-formaatti käyttää LZ77- ja huffman-koodausta yhdistävää DEFLATE-algoritmia pakkaamiseen, mutta pakkaamista edeltää suodatinvaihe, jossa esitetään eräänlaisia arvauksia tulevasta kuvadatasta. Tämän jälkeen arvaus ja arvauksen poikkeama todellisuudesta voidaan pakata yhdessä, ja lopputulos vie usein vähemmän tilaa kuin ilman suodatinvaihetta. [Duce, 2003]

Koska PNG-formaatin suodattimet ovat malliesimerkki tiedon ennakoinnista, esittelen ne tarkemmin. Formaatti käyttää viittä vaihtoehtoista suodatinta, joista valitaan jokaiselle kaksiulotteisen kuvan riville yksi, jolla käsitellään jokainen rivin pikseli. Suodattimet ovat [Duce, 2003]:

1. Ei suodatinta (None). Rivin pikseliarvoja ei käsitellä.
2. Ero (Sub). Jokaisen pikselin arvoksi tulee tämän pikselin ja sen vasemmalla puolella olevan pikselin arvojen erotus.
3. Ylempi (Up). Kuin ero-suodatin, mutta arvoksi tulee tämän pikselin ja sen yläpuolella olevan pikselin erotus.

4. Keskiarvo (Average). Suodin palauttaa pikselin vasemmalla puolella ja yläpuolella olevien pikseleiden arvojen keskiarvon erotuksen pikselistä itseltään.
5. Paeth (Paeth). Suodin laskee pikselin vasemmalla puolella, yläpuolella ja ylävasemmalla olevien pikseleiden erään lineaarisen funktion ja vähentää käsiteltävänä olevasta pikselistä sen pikselin arvon, joka on lineaarisen funktion tulosta lähinnä.

Suodattimet siis käyttävät hyväkseen tietoa jo purettujen pikseleiden arvoista ja mukautuvat dataan valitsemalla sopivimman suodattimen. Huomattavaa on myös se, että suodatinvaihtoehdot ovat selkeästi suunnattuja kaksiulotteiselle datalle, jollaista kuvadata juuri onkin.

4. Häviöllisistä algoritmeista

Sitä, kuinka ja mitä ihminen näkee, on luonnollisesti tutkittu pitkään. Saamalla selville, mitkä informaatio-osat kuvasta ovat ihmiselle tärkeintä kuvan hahmottamisen kannalta, voidaan pakkausalgoritmeissa käyttää tilaa ja aikaa juuri näiden informaatio-osien pakkaamiseen. *Ihmisen näköjärjestelmästä* (human visual system, HVS) onkin saatu selville muun muassa, että olennaista kuvan tunnistamisen kannalta on informaatio hahmojen reunoista. Siksi onkin nähty paljon vaivaa, jotta osattaisiin koneellisesti erottaa pinta- ja reunainformaatio toisistaan ja pakata näitä erillisillä algoritmeilla. [Reid et al., 1997] Samoin on saatu selville, että ihmissilmälle on väri-informaatiota oleellisempaa kirkkausinformaatio [emt].

Kun käytetään häviöllistä pakkausalgoritmia, on oleellista, että purettu kuva muistuttaa kyllin paljon alkuperäistä pakkaamatonta kuvaa. Määritelmällisesti kaksi kuvaa ovat samankaltaisia, jos niiden jokainen pikseli, kuvayksikkö, on samankaltainen [Kosheleva, 2004]. Tämä tarkoittaa sitä, että ero kahden kuvan vastaavien pikselien välillä on pieni. Tämä lienee myös intuitiivisesti selvää. Usein ei kuitenkaan ole selvää, mitä tarkoittaa ”pieni ero” pikselien informaatioissa. Onkin esitetty joitain vaihtoehtoisia tapoja laskea tätä informaatioeroa [emt].

4.1 Diskreetti kosinimuunnos

Yksi mullistavimmista pakkausteknologian keksinnöistä 1980-luvulla oli hyödyntää diskreettiä kosinimuunnosta kuvadatan esittämiseen. Diskreetti kosinimuunnos on erään tyyppinen Fourier-muunnos, joka muuntaa signaalin taajuusavaruuteen, jossa voidaan tarkastella eri taajuuskomponenttien osuutta

signaalissa. Tekniikkaa käytetään pakkauksessa, koska ihmissilmälle yleisesti tarkoitettu informaatio yleensä muuttuu hitaasti, eli korkeataajuiset komponentit ovat vähäisiä. Muunnoksen jälkeen nämä korkeataajuiset komponentit voidaan hävittää kvantisoimalla. Ihmissilmälle tämä hävitetty informaatio ei ole olennaista. [Judin, 2004]

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

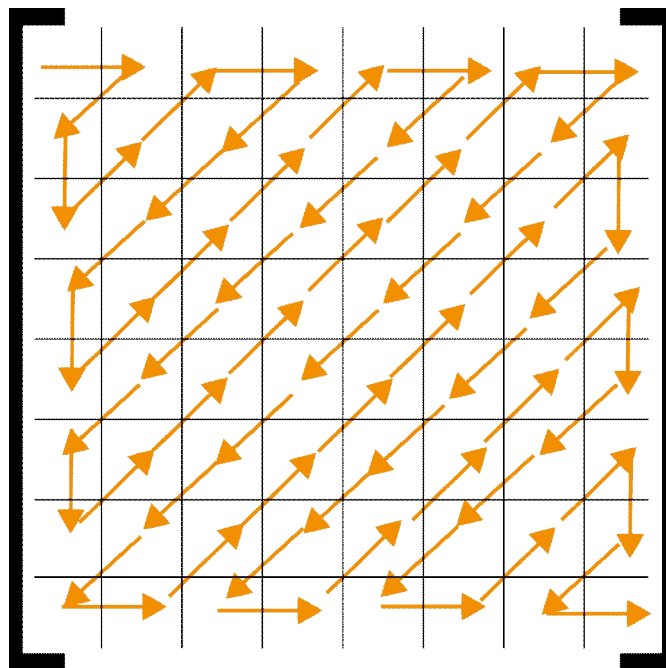
Kaavio 1. JPEG-kvantisointimatriisit

Joint Photographic Experts Group -komitea antoi oman nimensä JPEG-kuvaformaatille, joka on yksi yleisimmin käytettyjä valokuvapakkausformaatteja. Vuonna 1992 julkaistu formaatti yhdistää monia visuaalista järjestelmää hyödyntäviä elementtejä saadakseen hyvälaatuisen ja pienikokoisen pakkauksen. Formaatti toimii seuraavasti [Wallace, 1991]:

1. Värikanavamuunnos. Puna-viher-sininen rgb-data muunnetaan YCbCr-muotoon, jossa Y on pikselin kirkkaus ja Cb ja Cr ovat kromaattisia värikanavia.
2. Osiointi. Koko kuva hajotetaan 8x8 -kokoisiin alueisiin.
3. Värikanavien tarkkuuden alentaminen. Tässä vaiheessa käytetään hyödyksi tietoa ihmissilmän suuremmasta herkkyydestä kirkkausinformaatiolle verrattuna väri-informaatioon [Reid et al., 1997]. Värikanavien Cb ja Cr tarkkuutta alennetaan suhteessa 4:4:4 (ei lainkaan), 4:2:2 (puolella vaakasuuntaan) tai 4:2:0 (puolella vaaka- ja pystysuuntaan). 4:2:2 pakkaus pienentää lopullisen kuvan kokoa noin 33 %, 4:2:0 noin 50 %.
4. Diskreetti kosinimuunnos. 8x8-kokoisille alueille, tarkemmin sanottuna alueiden jokaiselle kanavalle, suoritetaan kaksiulotteinen muunnos. Muunnoksen tuloksena on 8x8-kokoinen matriisi, jossa eniten merkitsevä arvo on vasemmassa yläkulmassa ja oikeaa alanurkkaa lähestyessä arvojen merkitys muuttuu vähäisemmäksi, kuvan pieniksi detaljeiksi.
5. Kvantisointi. 8x8-kokoisen alueen edellisestä vaiheesta jääneet arvot jaetaan kvantisointimatriisiin arvoilla ja pyöristetään kokonaisluvuksi. Tämä on tär-

kein häviöllinen vaihe pakkauksessa. Yleisimmin käytetty kvantisointimatriisi on kaaviossa 1. Siinä on huomattavaa lukujen kasvaminen oikeaa alanurkkaa lähestyessä. Tämä tarkoittaa käytännössä sitä, että yksityiskohtia, korkean taajuuden arvoja, pakataan karkeammin. Käyttämällä esitettyä matriisia suurempia arvoja saadaan aikaan aggressiivisempi, enemmän tietoa hukkaava pakkaus.

6. Entropiakoodaus. Aluksi kaksiulotteiset osiot kootaan zigzag-kuviolla (ks. kaavio 2) yksiulotteiseksi dataksi. Koska kvantisoidessa käytetään matriisin oikeassa alanurkassa verrattain suuria arvoja, keskittyy tuohon kulmaan usein suuri joukko nollia, jotka zigzag-kuviolla kootessa järjestyvät perättäisiksi. Tämä yksiulotteiseksi järjestetty data juovakoodataan, jolloin perättäiset nollat menevät pieneen tilaan. Sen jälkeen kaikkien osioiden data Huffman-koodataan yhdessä. Formaatti sallii Huffman-koodauksen tilalla myös aritmeettisen koodauksen, mutta sitä ei patenttisyyistä yleensä käytetä.



Kaavio 2: Zigzag-koontijärjestys

4.2 Aaloke-muunnos

Aaloke-muunnos (wavelet transform) on verrattain uusi tulokas kuvanpakkauksen kentässä. Sitä verrataan usein diskreettiin kosinimuunnokseen, sillä molempien toimintaperiaatteena on muuntaa kuvan pikseleiden väri-informaatio helpommin pakattavaksi, toisteisemmaksi, informaatioksi. Lisäksi tunnetuimman diskreettiä kosinimuunnosta käyttävän formaatin, JPEGin, edelleen kehitetty versio JPEG2000 käyttää aaloke-muunnosta. [Skodras et al., 2001].

Tekniikoissa on luonnollisesti kuitenkin eroja. Siinä missä diskreettiä kosini-muunnosta yleensä käytetään jakamalla kuva lohkoihin, toimii aallocke-muunnos koko kuvan alalla. Tästä seuraa muun muassa se, että jo informaationsiirron alkupuolella voidaan luoda karkea kuva lopullisesta kuvasta. [Au et al., 2007].

4.3 Fraktaalit

Fraktaalipakkausta ehdotettiin ensikerran 1980-luvun lopussa. Tekniikka perustuu tosielämästä otettujen kuvien sisäiseen samankaltaisuuteen, *itse-similaarisuuteen*. Fraktaalipakkauksen ongelmaksi on koettu, että vaikka pakattu tiedosto onkin yksiselitteisen helppo purkaa, on pakkaaminen monimutkaista ja siten hidasta sopivien parametrien etsintää.

Vuonna 1992 ehdotettiin tekniikkaa, jossa kuva jaetaan pienemmiksi epäällekkäisiksi lohkoiksi, ja jokaiselle lohkolle etsitään eniten sitä vastaava lohko. Tämän jälkeen jokainen lohko voidaan ilmaista muunnoksena sitä eniten vastaavasta lohkoista. Lopuksi löydetään muunnosten joukko, jonka avulla saadaan luotua koko kuva. [Wua et al., 2006]. Nämä muunnosten joukot ovat iteroituvia funktiosysteemejä (IFS) [Käenmäki, 2003]. Tekniikasta ei vielä ole onnistunut toteutusta.

6. Lopuksi

Pitkään kuvienkin pakkaukseen käytetyt geneeriset pakkausalgoritmit, siis käytännössä entropiakoodausalgoritmit, ovat väistyneet kuvadataspesifimpien pakkausalgoritmien tieltä. Hyödyntämällä tietoa ihmisen visuaalisesta järjestelmästä voimme nykyään keskittyä pakkaamaan sitä dataa, joka on hahmotuksen ja kuvantunnistuksen kannalta olennaisinta. Lisäksi havainnoimalla yleisesti toistuvia piirteitä kuvadatassa voimme tehdä algoritmeja, jotka toimivat erityisen hyvin tietyntyyppisen kuvandatan pakkauksessa. Voimme esimerkiksi ennakoita tietoa, kuten PNG-formaatin suodatinvaiheessa tehdään, tai etsiä kuvasta itsesimilaarisuutta, kuten fraktaalipakkauksessa tehdään. Algoritmien kehitys myös jatkuu ja uusia pakkausmetodeja kehitetään jatkuvasti. Tällä hetkellä kiinnostavaa on esimerkiksi neuroverkkojen käyttö pakkauskeinona. Tutkimusta tehdään muun muassa Teuvo Kohosen kehittämän itseorganisoidun kartan ja ART1-verkon tiimoilta. [Soliman et al., 2006].

Viiteluettelo

- [Aho et al., 2000] Isto Aho, Erkki Mäkinen ja Timo Poranen, Algoritmien suunnittelu ja analyysi, Tampereen yliopisto, Tietojenkäsittelytieteiden laitos, 2000.
- [Au et al., 2007] K.M. Au, N.F. Law & W.C. Siu, Unified feature analysis in JPEG and JPEG 2000-compressed domains, *Pattern Recognition* 40, (July 2007), 2049-2062
- [Duce, 2003] David Duce (Ed.). Portable Network Graphics (PNG) Specification (Second Edition) Available <http://www.w3.org/TR/PNG/#F-Relationship>, 2003. Checked 26.02.2007.
- [Jiang, 1999] J. Jiang, Image compression with neural networks – A survey, *Signal Processing: Image Communication* 14, (June 1999), 737-760
- [Judin, 2004] Judin Pekka, Fourier-transformaatiot ja kuvien pakkaus, <http://mark.math.helsinki.fi/Symbolinen%20laskenta/Seminar%20Talks/Judin/Slides.pdf>, 2004. Luettu 26.03.2007
- [Kosheleva, 2004] O. Kosheleva, On the optimal choice of quality metric in image compression: a soft computing approach. *Soft Computing* 8, 4, 2004, 268-273.
- [Käenmäki, 2003] Antti Käenmäki, Iterated function systems: Natural measure and local structure. Rep. Univ. Jyväskylä Dept. Math. Stat. 93, <http://www.math.jyu.fi/~antakae/publications/preprints/thesis.pdf> November, 2003. Checked 26.04.2007.
- [Reid et al., 1997] M. M. Reid, R. J. Millar and N. D. Black, Second-generation image coding: An Overview, *ACM Computing Surveys* 29, 1 (March 1997), 3-29.
- [Saupe and Hamzaoui, 1994] Dietmar Saupe and Raouf Hamzaoui, A review of the fractal image compression literature, *ACM SIGGRAPH Computer Graphics* 28, (1994), 268-176.
- [Skodras et al., 2001] A. N. Skodras, C. A. Christopoulos and T. Ebrahimi, JPEG2000: The upcoming still image compression standard. *Pattern Recognition Letters* 22, (2001), 1337-1345.
- [Soliman et al., 2006] Hamdy S. Soliman and Mohammed Omari, A neural networks approach to image data compression, *Applied Soft Computing*, 6, (March 2006), 258-271.
- [Wallace, 1991] Gregory K. Wallace, The JPEG still picture compression standard, *Communications of the ACM* 34, 4, (April 1991), 30-44.
- [Wua et al., 2006] Ming-Sheng Wua, Jyh-Horng Jengb and Jer-Guang Hsieha, Schema genetic algorithm for fractal image compression. *Engineering Applications of Artificial Intelligence* 20, (June 2007), 531-538.

Mobile Social Media

Sari Miettinen

Abstract.

This paper studies the characteristics of social media and the motivation for using social media applications. The paper presents the limitations and advantages mobile phone can bring to social media, studies the current situation on mobile social media and presents few ideas where the area of mobile social media could be headed in the future.

Keywords and terms: social media, community, mobile.

CR-classification: H.4.3, J.4,

1. Introduction

Social media has become the new buzzword of IT-business. This is mainly due to the fact that several social media applications and services have gotten masses of users. Advertisers are getting interested in the vast amount of users these services have. More traditional services are figuring out how to keep their current customers and are turning their eyes on social media. [Kujanen, 2007]

Social media application can be defined as an online service, where a group of people – a community - can share and consume content. The content can be text, images, videos, animation or sound. The evolution of Web programming techniques along with technical evolution has created a new term called Web 2.0, which will be defined more detailed in Chapter 2. Social software and social media are sometimes used as synonyms for term Web 2.0, but they can be seen as two different phenomena.

Social media itself is not a new phenomenon, the internet has always included social communication: irc, newsgroups, chats and forums can all be seen as social media [Melakoski et al., 2007]. New programming techniques like AJAX (Asynchronous JavaScript and XML) have only made it easier and faster to create appealing Web applications and combination of services. Traditional media houses like TV, magazines and radio publishers are now waking up to the rising of social media and user-generated content.

Accessing the internet with mobile phones have already outnumbered internet accesses via PC in some countries, and the growth of the mobile internet does not show any signs of decreasing [Wright, 2006]. Fast mobile broadband is coming to Europe with HSDPA and mobile phones are becoming more like small computers than simple calling devices. Most mobile users already send more SMS's than they make phone calls. The next revolution of internet,

Web 3.0, is thought to appear when mobile phones and other mobile device usage are combined to the new social media, that Web 2.0 brought along. [Koskinen, 2006]

In this paper the social media is studied as a phenomenon and few, especially mobile, social media applications are considered. At the end of this paper few ideas about the direction of mobile social media development are presented.

2. Definitions

Before I further study what social media actually is, I'll define few of the key terms I will be using in this paper.

The term Web 2.0 does not refer to any distinct application or a technology. Instead, it is a concept that gathers together a group of new technologies, phenomena and behaviour. O'Reilly [2005] lists some typical characteristics of Web 2.0 services: rich user experience, collective intelligence, importance of data and lightweight programming models. The term is most typically used to refer to new agile programming methods like AJAX and social-networking or online communities. Typically Web 2.0 applications use networks as a platform, the users own the data in the application, and the applications are participatory, provide rich end-user experience, include social networking aspects and enhanced graphics. [Hintikka, 2007]

Social software is an application that creates a technological premise for social media. The software should support, for example, easy content creation and sharing, and community formation [Li, 2005]. Many of the social software are community created and administrated. Social software is typically created with Web 2.0 tools like AJAX and has open API's (application programming interface).

Social media application is an interactive online service, where a community can share their opinions and other content. The term social media is used to describe the phenomena of these new media types. Community created content is content that has been published at the service. Content can be anything from text to videos and it is free of charge and usually free of any copyright claims.

The most important aspect of social media is its interactivity: one can comment, modify or other ways interact with the media. [Hintikka, 2007, Kangas et al., 2007] Social media does not have any time, place or quantity limitations like traditional media, for example, magazines and TV. Social media is not published at any particular time: it is accessible at any time. Social media is not restricted to any geographical area: it is accessible from anywhere in the world, where one is able to use the internet. Quantity limitations like circula-

tion do not apply to social media either: viewing or copying the electronic content does not cause extra costs to the producer of the media.

The debate over how to define community has been around longer than Web 2.0 [Bruckman, 2006], and the term can be defined in many different ways. I use the definition given by Schlichter et al. [1998], which defines communities as a group of people, who have something in common, but do not necessarily know each other or interact at a personal basis. Schlichter et al. says that the usefulness of communities lies in the fact that people can share experiences and perhaps get some help in solving their problems. The online communities usually follow the same convention.

Term virtual reality is often used to refer to a technology that allows user to interact with computer-simulated world. Virtual realities have been used long for flight training or other situations where virtual reality is cheaper or safer to use than the corresponding real-life situations. Virtual reality tries to mimic the real world in order to create similar environment. [Brit1]

With the rise of Web 2.0, virtual reality has got a new meaning. People often use it to refer to an application that creates virtual space where people can interact with each other. These virtual communities do not even try to mimic the real world, but can be fantasy worlds, like multiplayer online game World of Warcraft [WoW] or to use very bulk graphics like virtual community Habbo Hotel [HabboHotel]. I use the term virtual community to refer to online communities, and the term virtual reality to refer to mimicking of real world with computer-mediated devices.

The terms mobile and mobility refer to location free, to something that is not tied to any physical location. Creating and using media can be location free, not tied to any physical limitations such as LAN or PC.

3. Characteristics of social media applications

So what are the typical characteristics of social media applications? Social media applications can be seen as a set of services that enable people to communicate with each other better.

Zhang and Weiss [2004] list some characteristics of virtual communities. The members of the community need to be aware of other members in the community, but this must be balanced with privacy in the social media application. People do not want to post all their personal details for the entire community, but usually pose under a nickname, or alias.

Social media applications encourage people to interact and take an active role in the service. Users are not typically just passive consumers, but also active producers of content and participants of communication. The social media application is usually built upon an open source solutions and includes open

API's for users to build their own extensions and features for the application, like personalizing their profile page.

Social media applications almost always have some kind of communication channel, such as forum or chat. Social media applications allow people to create communities quickly and to communicate effectively with each other about their common interest. Social media applications often use combinations of different media-types and services and can be called as a mash-up service.

Moore and Serva [2007] divide virtual community-generated content systems, which can also be seen as social media applications, into four categories: wikis, blogs, internet forums and knowledge bases. They define wikis as open Website environments that allow rapid creation and modification of Web pages by anyone. Blogs are defined as frequently-updated Webpages, whose newest content is positioned at the top of the page. Internet forums are described as electronic bulleting boards, newsgroup or message board. Knowledge base is defined as an online knowledge management system or information repository. Knowledge base supports codification, storage and transfer of knowledge.

4. Examples of Web-based social media applications

4.1. IRC-galleria

IRC-galleria [IRC-galleria] is a good example of social media application. Service was originally built for IRC (internet relay chat)-users to get to know each other by face rather than by just their nickname. The service got huge popularity also among non irc-users and nowadays it is the biggest Finnish internet service with over 300000 daily users [Dynamoid]. Irc-galleria is especially popular among 13-17 year-olds, who use it as well as a communication channel rather than as a simple picture service, which it was originally designed to be. People have long dialogs by commenting each other's pictures.

The service has typical characteristics of social media, like commenting, user involvement, semi-open API and community creation. Users can create their own profiles and post their own pictures to the service. Moreover, users can comment on other user's pictures and view their profiles.

Community creation is built-in to the service, as originally people had to be part of at least one community, their Irc-channel. As the service has become more popular among other users, one can now create and join communities without any restrictions. Real Irc-users can identify themselves in the service and block non Irc-users from commenting and seeing them when logged in. The service includes few really huge communities with over 25000 members, but most of the communities are small with less than 100 people. Service is free, but for charge, one can buy extra features, like modified profile pages or private messages.

4.2. Wikipedia

Wikipedia [Wikipedia] is the world's largest online dictionary. Anyone reading an article in Wikipedia can edit or add more information to it. Wikipedia administrators are community chosen people that ensure some order while editing pages. Wikipedia has become the number one reference site in the internet.

Wikipedia is most widely known example of wikis, Websites, that allow everyone in the community to create, remove or edit pages in it. Wikis are widely used for group work and document-management in projects and companies.

4.3. Digg

The Digg [Digg] is "a user driven social content Website" [Digg]. It includes topics about news, videos or podcasts, that other people of the community have though as interesting, they "digg" it. Community members, all Digg users, can give their opinion about the topic, whether they think it is interesting or not. Digg can be seen as a sort of blog. Blog is usually thought as an internet diary, but Moore and Sherva [2007] defined blog as a frequently updated Web service, where the newest content is at the top. As Digg fulfils these requirements from most parts, it can be seen as a blog.

The content at the page changes constantly, as the newest popular topics are always put to the top of the page. People can search for the most popular topics from different time periods: what has been the most popular news today, or within a year. The topics at the site are direct links to internet magazines, blogs or other sites where the topic was picked from.

The site can be seen as receiving community-filtered information: what does the community consider is worth knowing. People are interested to know what other people think about some news or podcasts and post them to the Digg to get some reviews.

4.4. Habbo hotel

Habbo hotel [HabboHotel] is a good example of virtual community. The service metaphor is a hotel, in which each of the members has their own room. Each Habbo member can create their own character, Habbo, and modify their own site, Habbo Room.

Members, or guests of the hotel, can talk to each other, they can modify their appearances at any time, and with Habbo-money, they can decorate their own room. Habbo-money can be bought with real money. The service includes both individual and group minigames. For some games the players need Habbo-money to be able to play them.

Habbo hotel shows that a social service does not need to use elaborate graphics in order to succeed. Service uses very bulky graphics, which also en-

sure that the service is usable in various environments. The service has gotten masses of young users and it has been expanded to several different lingual releases.

5. Why people create communities and use social media

Nardi et al. [2004] studied why people use social media like blogs. Their finding was that bloggers had five main reasons why they wanted to share their lives with others: people wanted to update others of their whereabouts and activities, express their opinions to influence others, they wanted to seek others' opinions and feedback, and they wanted to "think by writing", and to release emotional tension.

Moore and Serva [2007] found out that the motivation to contribute to the community varies across the type of community in question, and that the motivation to use different communities depends on the persons needs. Wiki users seem to prize collaboration. The contribution of information gives wiki users a sense of belonging to something bigger, and can lead to improved self-esteem. Bloggers on the other hand, enjoy expressing themselves, and noticing that their expressions have potential influence on others.

The most common motivations behind a person contributing to a community are altruism, collaboration, egoism, knowledge and reputation. The motivation varies a lot between different types of community services. The users of internet forums do not consider collaboration as main motivation unlike bloggers, and wiki- and knowledge base -users.

Lyman [1998] has used gift giving as an analogy of virtual community. People exchange gifts and expect to receive equal information, or social status in the community. To sum it up, it can be said that the sense of belonging, sharing one's own opinion and gaining social status are the main motivators behind virtual communities.

6. What technical limitations and advantages mobile devices create?

This chapter concentrates to study what limitations and advantages mobile devices, especially mobile phones brings to social media creation and consumption.

The mobile phone was originally developed for calling. Today the devices are used to perform a variety of activities, such as internet browsing and digital imaging, but the primary function is still communication. The mobile phone software development has quite many technical and social limitations compared to traditional computing. [Kangas et al., 2007] Especially in the field of social media, where usability, easy access and convenience are keywords, the

mobile phone sets limitations for the service. On the other hand, mobile phone has characteristics that can bring new opportunities for social media services.

A lot of the social media content is produced with mobile devices, such as digital cameras, mobile phones and PDA's. Though, hardly any of the services support consuming of the content with mobile device. This is mainly due to the limitations of mobile devices, like small screen and difficult navigation.

The mobile devices rely on wireless network for internet access. These networks are usually limited by capacity and price. As many of the social media applications are Web-based applications, their transference to limited network devices can be challenging. Lots of pictures and other graphics do not sit well into the 20kb/s GPRS network.

Mobile phones have a small screen that limits the amount of data a person can view at a time. Textual data is the best form to present information for mobile device, but it restricts the user experience.

Mobile phone's input devices, even with extra accessories, are more limited than PC's input devices. The mobile phone has a small keyboard, usually with less than 20 keys, and the keys are used for a different purpose in different situations. Navigation with mobile phone is bulky and not as easy as with mouse and keypad.

The memory capacity of mobile phones is limited, even though memory card extensions can be used. The performance capacity of mobile phones is still far behind that of PC's. [Li, 2005]

The issues of privacy and security concerns mobile phone users as well. Can one truly anonymously use the service if one is using her own phone number? What if someone gets a hold of your mobile phone number and tracks you down? These are some of the security and privacy concerns that mobile phone users have. [Li, 2005]

The mobile phone does not only bring limitations though, but it also opens new doors with features that a traditional PC does not offer by default. Mobile phone usage differs from the PC usage, as the mobile phone is always on and it is always brought along. The mobile phone is more easily accessible: it is more convenient device to access services while out and about.

Most important feature that a mobile phone can bring to social media is the context-awareness. The media can be connected to the social or physical context. The physical location can be used as a filter when presenting services for a person. The people that the user is with can be used as a filter as well.

The current mobile phones are starting to have even more sophisticated features that can be utilized in social media application. High-resolution digital cameras, GPS receivers, music and media players can enable even richer applications. Unfortunately, currently the mobile phones are mostly used only as an

input-device for Web-applications. Truly mobile social media applications are still waiting.

7. Examples of current mobile social media

7.1. IRC-galleria WAP

IRC-galleria [IRC-Galleria] has brought a WAP-service, which enables members to check if they got any new comments, add new pictures to their profile, and for extra charge, view other users' pictures and comment on them. Not all abilities of the Web-service are available to the mobile users, features like community creation or searching members by the community are not available.

7.2. PocketPal

PocketPal [PocketPal] is a bit similar virtual community application as Habbo Hotel. User creates a character of their own to the service and decorates their own room. The pal that created pal, can be then transferred to mobile phone with a Java application. The character needs to be fed, entertained and taken care of. The Pal is a sort of a virtual pet that needs to be nurtured. The character can be sent to the PalCity, where one can communicate with other pals.

The service includes a friend-list and an internal messaging system. Communities can be joined and formed by anyone. The actual communication happens at the PalCity chat rooms, in a similar manner as in the HabboHotel. The difference is that user can enter to the PalCity even from their mobile application.

In a user study from the application developers [IronStar, 2007], it was noticed, that some of the users used the application instead of SMS's to communicate with each other. The application was most popular among 13-15 years old females.

7.3. Jumbuck

Jumbuck [Jumbuck] company offers a variety of mobile social software for mobile operators. The applications are not accessible directly by common mobile phone users, only through a partner operator or other service provider.

Jumbuck offers different types of chat and community services. The Jumbuck Island is bit similar service to the PocketPal. Users can create their own character and talk with other community members at the service. The service is implemented as an installable Java application.

Power Chat is a WAP-based service, where people can chat with other users in chat rooms. The rooms can be created by the users. Users can create and maintain their own profiles. They can upload images to the service via email or MMS, they can create friendlist, and use instant messaging with the members

on their friendlist. Both applications are simple communication applications that do not support community creation or management.

7.4. Rabble

Rabble [Rabble] is a WAP-based social networking application. At their Web page Rabble proclaims to be a location based application that is downloaded to user's phone. In reality it is purely a WAP-based application that is used with mobile phone's browser. Service does not have any main driver or methodology behind it. Instead it tries to offer all known social media features to the user. User can write their own blog, create friendlist, present their own pictures, comment other users, create communities - groups, and create events and places.

The application tries to offer everything at the same time, and fails to be superior at anything. The company has not focused to any feature of the service, which makes the service difficult to comprehend and to get a clear view of the system.

7.5. Dodgeball

Dodgeball [Dodgeball] is an SMS-based service that allows people to meet up friends and friends-of-friends while out and about. A user sends a text message to the service saying where the user is located at the moment, and the service then notifies all the user's friends within 10 blocks, informing them where the user is at the moment. User will be notified if any of their friends-friends are nearby. Users can also send broadcast messages to their friends. They simply send one message to the service and it is then broadcasted to all the people on user's friendlist. This is a quick and easy way to reach all your friends at the same time, if you, for example want to find someone to go to lunch with. The service allows users to get connected there and then. The service is entirely SMS-based and no WAP-connections or installable software is needed. User has to register to the service via Web though.

7.6. Summary

As we can see from the service examples, the mobile social media is still in its early stages and still relies heavily on pre-existing Web-services. None of the presented applications offers truly mobile social media experience, they are all tied to the Web in one way or another. Either you had to create your profile first in the Web or the application is just an add-on to the pre-existing service. The limitations of these solutions are further discussed in the next chapter.

8. Why mobile social media?

Why should we have social media applications that are especially designed for mobile phone usage? Why can't we just transfer the ones in the Web directly to mobile?

Only two of the presented mobile social media applications are proprietary applications that need to be installed to the phone. All the others are either SMS or browser-based services. From my experience, the usability of mobile phone browser-based applications versus installed Java or Symbian application is significantly worse. The response time, navigation method and the way the information is presented varies a lot between these two basic access methods.

An installable application, even if it uses data access, offers better navigation possibilities, more structured way to present the information than browser based services, as information does not have to be presented as links. The user interface is richer as it can include more pictures because they do not have to be downloaded each time. If all information is downloaded each time the user uses the application, the response time of the program is significantly slower.

Mobile social media applications should be easy to use, agile and always connected to ensure rich user experience. The mobile social media should be context-aware in order for it to bring the added value to the customer. Current Web-based solutions aren't context-aware, which is one of the key benefits of mobile devices, especially mobile phone. It is always on, always with and the phone's profile follows the situation person is in. The phone is turned on a silent-mode when person cannot be reached and on general-mode when she is available.

The current Web-based solutions cannot be transferred as such to mobile phones. The implementation technique of current services is not the main obstacle as mobile phones are starting to support the new Web-programming techniques like AJAX. Instead, the PC-minded implementation solution, which relies heavily on mouse and keyboard, creates the biggest obstacle. These services are not simply usable in mobile environment.

Rantanen et al. [2004] present four needs of urban people when out and about. First one is the need to make new acquaintances. People would like to make new friends when spending time in public or semi-public places, but they do not know how to approach other people.

Second, people want to be aware of co-located friends. They wanted to know if some of their friends were nearby. Third need is to understand places. People want to be informed about the background and history of the place they are visiting and about general information like opening hours and such. Tourist guides might satisfy the need, but a discussion with people more familiar with the area would be more satisfying. The fourth was the need to share one's opinion about the surroundings.

The needs presented above can be easily satisfied with social media application, but the context (location, time, people) cannot be easily implemented into existing Web-applications. Nor are the existing social media applications easily accessible when out in the city. This is why we need context-aware applications that can be used right then and there. Mobile social media application can be the answer to this problem. When people want to connect with other people in the same area, mobile social media application is an easy way of doing that.

Mobile social media applications might have a demand on the professional field as well. People working with the same target can share information concerning the target without having to use proprietary database or even have to be aware of each other. [Kangas et al., 2007] Professionals could share knowledge, opinions and tips in a mobile internet forum or wiki.

9. The future of mobile social media

Mobile phones are used more and more for accessing the existing social media applications and proprietary mobile access portals are starting to emerge, like WAP-access to IRC-galleria. [Kangas et al., 2007]

The current solutions are lacking truly mobile communication and administration. The mobile access to social media applications is not as rich as the Web-application. The photo sharing service Flickr [Flickr] supports picture sending from mobile phone, but browsing through the pictures is very difficult from a tiny mobile screen.

Mobile applications also lack the context awareness that mobile phones can bring to social media applications. One might want to answer to a quick poll on their community site even when one is out in the city. Or one might want to find a buddy to go to lunch with when they are strolling around the city.

There is a demand for a truly mobile social software that would give all the abilities and features to mobile phone users as compared to internet users. As the issues with bandwidth are solved, some truly mobile social media application can start to emerge. Mobile applications that are easily accessible, agile and have live content, are the next generation of social media applications.

10. Conclusions

The social media is the next revolution of the internet. Traditional media companies have already started to make their own social media applications, in order to keep their current customers. [Kujanen, 2007] The number of mobile phones is soon exceeding the number of PC's and developing countries are often relying to mobile devices as their only access to the internet. [Parikh and Lazowska, 2006] The social media applications have their appeal and are surely going to be around for a long time.

The number of mobile devices in the world is rapidly increasing, and the features of these devices are getting more versatile. People carry their mobile phones with them all the time and context awareness can be utilized in many ways in social media applications.

New, truly mobile social media applications are needed as the current solutions support only Web-based community sites, and none of the studied applications supported truly mobile community creation and administration.

References

- [Brit1] Encyclopædia Britannica - virtual reality, [online] Available at <http://search.eb.com/eb/article-9001382>. Checked 8.5.2007.
- [Bruckman, 2006] Amy Bruckman, A new perspective on "community" and its implications for computer-mediated communication systems. In: *CHI '06 Extended Abstracts on Human Factors in Computing Systems*, 2006, ACM Press, 616 – 621.
- [Burak and Sharon, 2004] Asaf Burak, Taly Sharon, Usage patterns of Friend-Zone: mobile location-based community services. In: *Proceedings of the 3rd International Conference on Mobile and Ubiquitous Multimedia MUM '04*, October 2004, ACM Press, 93-100.
- [Digg] Digg-community [online] Available at <http://digg.com/>. Checked: 7.5.2007
- [DodgeBall] Dodgeball-community [online] Available at <http://www.dodgeball.com/>. Checked: 6.5.2007.
- [Dynamoid] Dynamoid-yrityksen markkinointimateriaali [online] Saatavilla <http://www.dynamoid.com/mainostus/>. Luettu: 7.5.2006.
- [Flickr] Flickr photo service [online] Available at <http://www.flickr.com/>. Checked: 8.5.2007
- [Gross and Kleppe, 2005] Tom Gross and Martin Kleppe, Decision-making and communication: FrameDrops: a mobile VideoBlog for workgroups and virtual communities. In: *Proceedings of the 2005 International ACM SIGGROUP Conference on Supporting Group Work GROUP '05*, November 2005, ACM Press, 128-131.
- [HabboHotel] Habbo Hotel [online] Available at <http://www.habbo.fi/>. Checked: 6.5.2007.
- [Hietanen, et al., 2007] Hietanen Herkko, Oksanen Ville and Välimäki Mikko, *Community Created Content. Law, Business and Policy*. Turre Publishing, 2007.
- [Hintikka, 2007] Kari A. Hintikka, Johdatus internetin uusiin liiketoimintamahdollisuuksiin, Tietoyhteiskunnan kehittämiskeskus ry, Report 28,

- Tammikuu 2007. Saatavilla http://www.tieke.fi/mp/db/file_library/x/IMG/20815/file/julkaisu_28.pdf.
- [Irc-galleria] Irc-galleria -yhteisö [online] Saatavilla <http://irc-galleria.net/>. Luettu: 6.5.2007
- [IronStar, 2007] IronStar Helsinki Ltd., PocketPal Pilot Report, March 2007.
- [JumBuck] Jumbuck-service [online] Available at <http://www.jumbuck.com/home/default.php>. Checked: 6.5.2007.
- [Kangas et al., 2007] Petteri Kangas, Santtu Toivonen ja Asta Bäck, Googlen mainokset ja muita sosiaalisen median liiketoimintamalleja, 2007. Saatavilla <http://www.vtt.fi/inf/pdf/tiedotteet/2007/T2369.pdf>.
- [Koskinen, 2006] Petri Koskinen, Web 3.0 paljastaa sinusta kaiken, Talouselämä, Joulukuu 18, 2006. Saatavilla http://www.talouselama.fi/docview.do?f_id=1082577.
- [Kujanen, 2007] Janne Kujanen, Yhteisöllinen media ja sen kaupallinen hyödyntäminen, Sosiaalisen median seminaari, Maaliskuu 3, 2007. Saatavilla <http://www.professia.fi/@Bin/1552702/janne%20kujanen%202903%202007.pdf>.
- [Li, 2005] Xiaoqing Li, Buddy-finding in the mobile environment, *Technovation* 25, 9 (Sept.2005),1017-1023.
- [Lyman, 1998] Peter Lyman, The Poetics of the Future: Information Highways, Virtual Communities and Digital Libraries. Draft, November 8, 1998. Available as <http://www.ischool.berkeley.edu/~plyman/articles/Lazerow.pdf>.
- [Melakoski et al., 2007] Cai Melakoski, Sohvi Sirkesalo and Helena Tirronen, "Himottaa mutta pelottaa", Tampereen ammattikorkeakoulu, Sarja B-19, Maaliskuu 2007. Saatavilla <http://www.somelab.fi/files/Himottaamuttapelottaa.pdf>.
- [Moore and Serva, 2007] Trevor Moore and Mark Serva, Understanding member motivation for contributing to different types of virtual communities: a proposed framework, In: *Proceedings of the 2007 ACM SIGMIS CPR conference on 2007 Computer Personnel Doctoral Consortium and Research Conference: The Global Information Technology Workforce, 2007*, ACM Press, 153-158.
- [Parikh and Lazowska, 2006] Tapan Parikh and Edward Lazowska, Designing an architecture for delivering mobile information services to the rural developing world, In: *Proceedings of the 15th International Conference on World Wide Web, 2006*, ACM Press 798-800.
- [PocketPal] PocketPal, Mobile community [online] Available at <http://www.pocketpal.fi/index.jsp>. Checked: 6.5.2007.
- [Rabble] Rabble [online] Available at <http://www.rabble.com/View/About/>. Checked: 6.5.2007

- [Rantanen et al, 2004] Matti Rantanen, Antti Oulasvirta, Jan Blom, Sauli Tiitta and Martti Mäntylä, InfoRadar: group and public messaging in the mobile context. In: *Proceedings of the Third Nordic Conference on Human-Computer Interaction NordiCHI '04*, October 2004, ACM Press, 131-140.
- [Schlichter et al., 1998] Johann Schlichter, Michael Koch and Chengmao Xu, Awareness - The Common Link Between Groupware and Community Support Systems, *Community Computing and Support Systems: Social Interaction in Networked Communities*, Springer, Berlin, 1998, 77.
- [Wikipedia] Wikipedia, Online dictionary [online] Available at <http://www.wikipedia.org/>. Checked: 5.5.2007.
- [WoW] World of Warcraft -online game [online] Available at <http://www.worldofwarcraft.com/index.xml>. Checked: 6.5.2007.
- [Wright, 2006] Adam Wright, The Face of The Web, 2006, Ipsos Insight. Also available at <http://www.ipsos-na.com/news/pressrelease.cfm?id=3049>.
- [YouTube] YouTube [online] Available at <http://www.youtube.com/>. Checked: 6.5.2007.
- [Zhang and Weiss, 2003] Yanru Zahn and Michael Weiss, Virtual Communities and Team Formation, *Crossroads* **10.1**, 2003.

Kolmiulotteisen geometrian topologisesta esittämisestä monikulmioverkoilla

Matias Muhonen

Tiivistelmä.

Monikulmioverkkojen topologisen informaation esittämistä voidaan tehostaa b-rep-rakenteilla ja piilopintojen poistamisella. Vierekkäisyys- ja kytkentärelaatioiden eksplisiittisellä esittämisellä b-rep-rakenne suoriutuu tehokkaammin topologisista kyselyistä, sillä ne ovat vastaus monia mallia koskettaviin kysymyksiin.

Avainsanat ja -sanonnat: Monikulmioverkot, b-rep, topologia, piilopintojen poistaminen

CR-luokat: E.1, I.3.7

1. Johdanto

Tietokonegrafiikassa kolmiulotteisten kappaleiden geometria esitetään yleensä monikulmioverkoilla. Monikulmioverkot voidaan esittää yksinkertaisesti listana verkon kulmapisteitä ja monikulmioita [Mäntylä, 1988]. Esitystapa on yksinkertainen ja tehokas moneen tarkoitukseen. Esimerkiksi OpenGL-grafiikkarajapinta tukee monikulmioverkon määrittämistä kulmapistelistoina (vertex array) [Shreiner et al., 2004]. On kuitenkin olemassa sovellusalueita, joissa monikulmioverkko on tehotonta esittää yksinkertaisena kulmapistelistana.

Usein monikulmioverkkoa halutaan käsitellä eri tarkkuustasoilla (level of detail, LOD). Lähtökohtana on tarkka monikulmioverkko, josta muodostetaan eri tarkkuustasolle approksimoitu versio. Tällöin on usein tarpeen poistaa verkosta särmiä ja supistaa ne yksittäiseksi kulmapisteeksi. Supistaminen tehdään kaksivaiheisesti. Ensin haetaan särmän rajaavat monikulmiot ja poistetaan ne. Tämän jälkeen päivitetään uusi monikulmio niistä särmistä, jotka jäävät poiston jälkeen jäljelle. Toimenpide vaatii erityistä topologista informaatiota verkon rakenteesta, kuten tiedon siitä, mitkä monikulmiot ja särmit ovat vierekkäisiä. Topologisen informaation selvittäminen kulmapistelistasta on mahdollista, mutta tietojenkäsittelyllisesti kallista, sillä esimerkiksi vierekkäisyyden selvittäminen vaatii pahimmassa tapauksessa koko monikulmio-listan läpikäynnin.

Verkon topologinen tieto liittyy vierekkäisyyteen ja kytkentöihin. Usein halutaan esittää ainakin seuraavan tyyppisiä kyselyitä [McGuire, 2000]:

- Mitkä monikulmiot ovat vierekkäisiä yksittäiselle monikulmiolle?
- Mitkä monikulmiot on kytketty yksittäiseen kulmapisteeseen?
- Mitkä monikulmiot rajaavat yksittäisen särmän?
- Mitkä särmät on kytketty yksittäiseen kulmapisteeseen?
- Mitkä särmät rajaavat yksittäisen monikulmion?

Monikulmioverkkojen tapauksessa on olemassa rajaavia tietorakenteita (b-rep), joilla tämän tyyppisiin kyselyihin saadaan tehokkaasti vastauksia. Esimerkiksi half-edge-tietorakenne [Mäntylä, 1988] kykenee vastaamaan useimpiin topologiaan kyselyihin vakioajassa. Lisäksi on olemassa useita erityistarpeisiin kehitettyjä tietorakenteita, kuten tarkkuustasojen käsittelyyn tarkoitettu progressive mesh [Hoppe, 1998]. AIF [Silva and Gomes, 2004] määrittelee täsmällisesti kaikki verkon topologiset relaatiot. Topologisen informaation eksplisiittisen esittämisen haittapuoli on lisääntyvä tilankäyttö.

Monikulmioverkon topologian esittämisen lisäksi halutaan yleensä rajata vain pieni osa mallia renderöitäväksi. Mallin näkyvä osa rajataan katselutilavuudella. Yksinkertaisimmillaan monikulmioverkon koko on pieni, joten käytännössä renderöinti koskee kaikkia verkon monikulmioita. Monimutkaisemmat mallit, kuten tietokonepelien pelikentät, saattavat koostua tuhansista yksittäisistä kappaleista, joista kerrallaan katselutilavuudessa on näkyvissä vain pieni osa.

Katselutilavuuden rajaaminen on osa renderöinnissä tapahtuvaa piilopintojen poistoa. Katselutilavuuteen kuuluvat monikulmiot voidaan rajata rajauslaatikkotekniikoilla [Samet, 1989], joissa malli on jaettu hierarkkisesti pienempiin kokonaisuuksiin, joista kukin vastaa tiettyä avaruuden osaa. Valitsemalla optimaaliset kokonaisuudet renderöintiin voidaan pienentää renderöinnistä aiheutuvaa kuormaa piilopintojen poistossa. Octree [Picco, 2003] on yksi tähän menettelyyn pohjautuva rakenne.

2. 3D-mallien esittäminen

3D-malli on matemaattinen esitys kolmiulotteisesta kappaleesta. Malli voidaan esittää esimerkiksi *NURBS-pintana* [Rogers and Earnshaw, 1991] tai *aliositusmallina* (subdivision model). Yleisimmin käytetään monikulmioverkkoja, sillä niiden etuna on yksinkertaisuus sekä laaja tuki rajapinta- ja laitteistotasolla. Monimutkaisemmat menetelmät pohjautuvat usein mallin redusoimiseen monikulmioverkoksi.

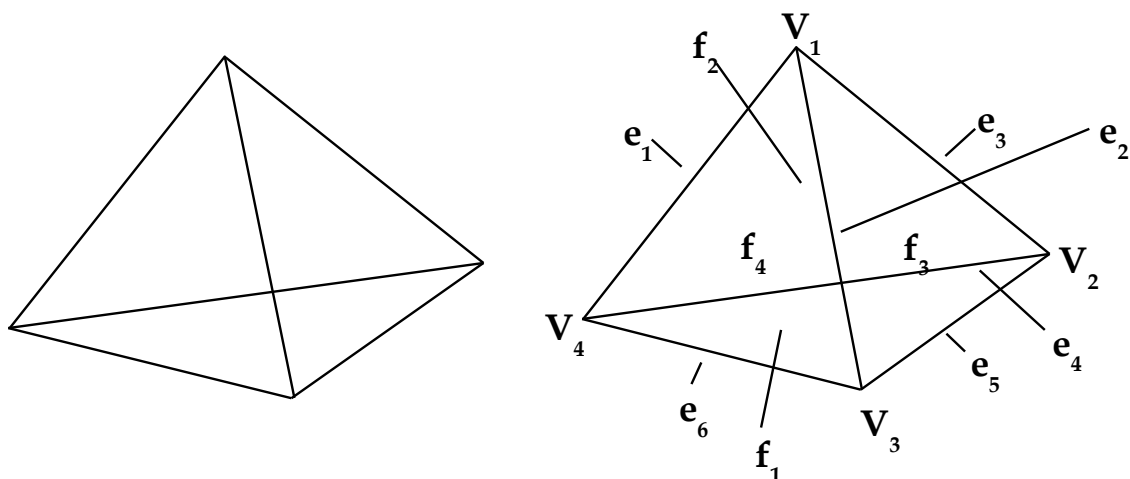
2.1. Monikulmioverkkomallit

Monikulmioverkko (polygon mesh) on joukko *solmuja* (vertex), *särmiä* (edge) ja *monikulmioita* (face), sekä niiden välisiä topologisia relaatioita [Botsch et al., 2002]. Solmut ovat yksittäisiä pisteitä avaruudessa. Särmä on jana, joka

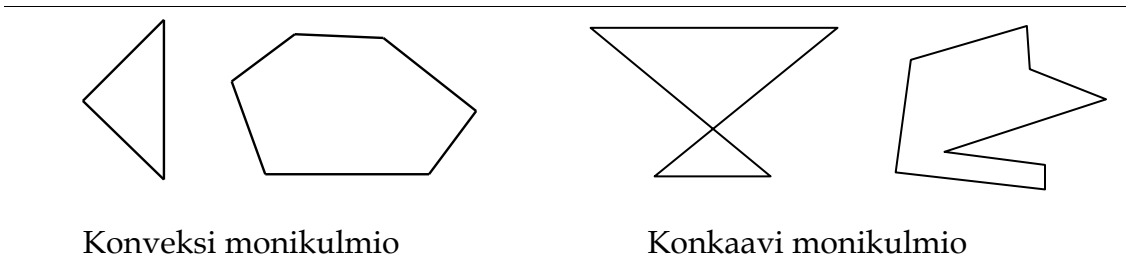
yhdistää kaksi solmua. Särmiä on osa yhtä tai useampaa monikulmiota. Monikulmiot muodostuvat suljetuista särmäketjuista. Monikulmion muodostama pinta on aina suljettu, joten ketjun peräkkäisten särmien on jaettava sama solmu. Monikulmioverkkojen erikoistapaus on *kolmioverkko* (triangular mesh), jossa verkon kaikki monikulmiot ovat kolmikulmaisia. Mikä tahansa monikulmioverkko voidaan redusoida kolmioverkoksi. 3D-renderöintilaitteistot käsittelevät usein vain kolmen särmän monikulmioita, joten on käytännöllistä rajoittaa verkko kolmioverkoksi.

Kuvassa 1 on esitetty nelitahokas, ja sitä vastaava monikulmioverkko, joka on myös kolmioverkko. Nelitahokkaassa on neljä solmua $\{V_1, V_2, V_3, V_4\}$, joiden välillä on joukko särmäjä $\{e_1, e_2, e_3, e_4, e_5, e_6\}$. Särmäjoukosta muodostuu neljä monikulmiota $\{f_1, f_2, f_3, f_4\}$, joissa kussakin on kolme särmää.

Monikulmio on *konvekksi* (convex), jos se täyttää seuraavat kaksi ehtoa. Konveksin monikulmion jokainen *sisäkulma* (internal angle) on korkeintaan 180° . Sisäkulma on määritetty monikulmion jokaiselle solmulle saman päätepisteen jakavan särmän välisenä kulmana. Toisena ehtona konveksille monikulmiolle määritetään joukko E , joka sisältää jokaisen mahdollisen janan monikulmion kahden solmun välillä. Tällöin E sisältää vain sellaisia janoja, jotka pysyvät monikulmion särmien muodostaman ulkorajan sisällä tai täsmälleen seuraavat ulkorajaa. *Konkaavi* (nonconvex) monikulmio ei täytä konveksin monikulmion ehtoja, vaan esimerkiksi jokin sen särmä leikkaa toisen särmän. Tällaista leikkausta on havainnollistettu kuvan 2 vasemmassa konkaavissa monikulmiossa. Kuvan 2 oikeanpuoleinen konkaavi monikulmio sisältää janoja, jotka ylittävät särmien muodostaman ulkorajan. Käytännön sovelluksissa konkaavit monikulmiot voivat aiheuttaa ongelmia. OpenGL-grafiikkarajapinta ei esimerkiksi tarkasta konkaaveja monikulmoita, eikä renderöi niitä oikein [Shreiner et al., 2004]. Tästä syystä monikulmioiden tulisi



Kuva 1. Nelitahokkaan monikulmioverkkoesitys [Patrikalakis, 2003]

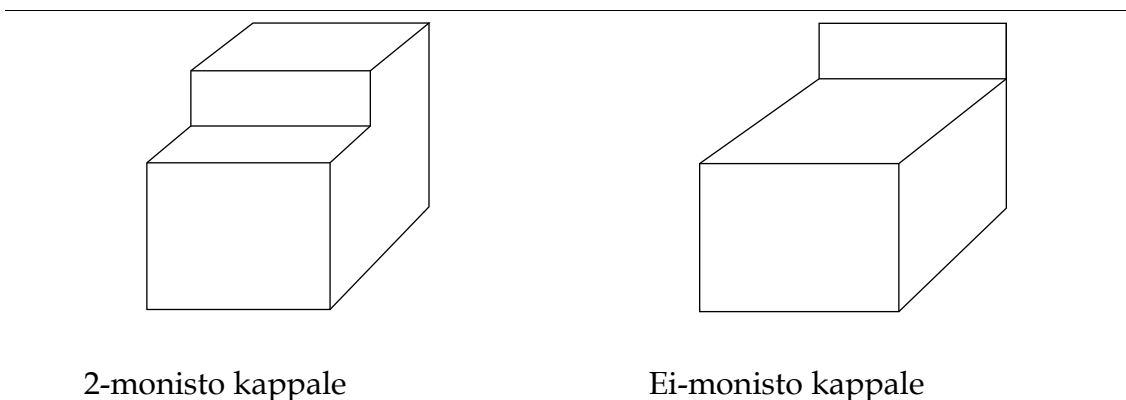


Kuva 2. Konvekksi ja konkaavi monikulmio [Shreiner et al., 2004]

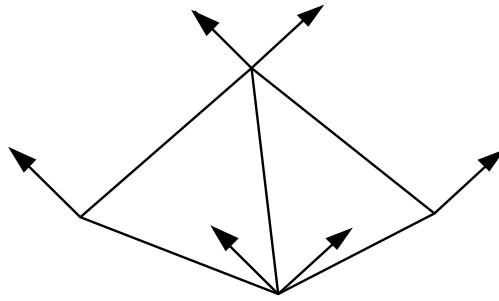
olla konvekseja. Konkaavi monikulmio voidaan aina redusoida konveksiksi pilkkomalla monikulmio useammaksi monikulmioksi. Kolmioverkon tapauksessa monikulmiot ovat aina konvekseja.

Monikulmioverkot kategorisoidaan topologian mukaan *2-monistoihin* (two-manifold) ja *ei-monistoihin* (non-manifold). 2-moniston pinta käyttäytyy paikallisesti kaksiulotteisen tason kaltaisesti jokaisessa pintaan kuuluvassa pisteessä. Tätä voi havainnollistaa ajattelemalla paperiarkkia, josta taittelemalla saadaan kolmiulotteinen kappale aikaiseksi. Täsmällisesti määriteltynä jokaisessa pinnan lokaalissa pisteessä on mahdollisuus liikkua maksimissaan kahteen vierekkäiseen suuntaan. Kuution sivulla voidaan esimerkiksi liikkua ylös tai alas ja vasemmalle tai oikealle – toisin sanoen maksimissaan kahteen suuntaan. Ei-monistossa kappaleessa liikkumasuuntia on enemmän kuin kaksi. Esimerkiksi kuutio, johon on liittyneenä yksi ulospäin suuntautunut sivu on ei-monisto, sillä tietyissä lokaaleissa pisteissä liikkumasuuntia on enemmän kuin kaksi. 2-monistoista ja ei-monistoista kappaleista on esimerkki kuvassa 3. Ei-monisto kappale voidaan redusoida 2-monistoksi pilkkomalla kappale useampaan kappaleeseen, jotka esitetään eri monikulmioverkolla.

Kolmioverkkojen tapauksessa 2-monistaisuudesta seuraa, että verkon jokainen särmä saa olla korkeintaan kahden kolmion osana. Pinnoille on rajoitteena, että mikään kärkipiste ei saa jakaa erillistä pintaa. Tämä tarkoittaa esimerkiksi pyramidin tapauksessa, että muoto, jossa kaksi pyramidia jakaa yhdessä huippukärkensä, ei ole sallittu. Sama rajoite koskee monikulmioita. Esimerkiksi kaksi kuutiota, jotka jakavat yhden sivuistaan ei ole sallittu.



Kuva 3. 2-monisto ja ei-monisto kappale



Kuva 4. Nelitahokas ja pinnan orientaatio

Moniston asteluku on tietorakenteen valinnan kannalta merkittävä asia, sillä vain osa monikulmioverkkojen tietorakenteista soveltuu myös ei-monistojen kappaleiden esittämiseen (taulukko 2).

Monikulmiolla on etu- ja kääntöpuoli, jotka ovat tulkinnallisia monikulmion *orientaatiosta* (orientation). Etu- ja kääntöpuoli mahdollistavat kappaleen sisä- ja ulkopinnan määrittämisen. Orientaatio on tulkinnallinen normaalivektorista, joka osoittaa pinnasta kohtisuorasti 90° kulmassa. Orientaatiota on havainnollistettu kuvassa 4, jossa nelitahokkaan normaalivektorit ovat näkyvissä. Monikulmion orientaatio voidaan määrittää eksplisiittisesti asettamalla normaalivektori monikulmiolle. Toisena vaihtoehtona käytössä oleva tietorakenne asettaa omat käytäntönsä normaalien muodostamiselle implisiittisesti. Esimerkiksi OpenGL tulkitsee monikulmiot, joiden kärjet näyttäytyvät järjestyksessä vastapäivään ruudulla etupuolisiksi [Shreiner et al., 2004]. Laskenta tapahtuu monikulmion ruudulta viemän pinta-alan mukaan, joka tulkitaan positiiviseksi tai negatiiviseksi kärkien järjestyksen mukaan.

Orientaation on syytä olla yhdenmukainen koko monikulmioverkon tasolla, sillä grafiikkarajapinnat käyttävät yleisesti orientaatiotietoa hyväkseen teksturoinnissa ja valaistuslaskennassa. Osa tietorakenteista, kuten half-edge [Mäntylä, 1988], on orientoituja, mutta toiset, kuten AIF [Silva and Gomes, 2004], vaativat erityisiä toimenpiteitä orientaation laskemiseksi.

Yksi triviaaleimmista tietorakenteista monikulmioverkon esittämiseen on särmiin pohjautuva malli [Mäntylä, 1988]. Monikulmioverkosta muodostetaan

Särmä	Solmut	Solmu	Koordinaatit	Monikulmio	Särmät
e_1	$V_1 V_2$	V_1	$x_1 y_1 z_1$	f_1	$e_6 e_5 e_4$
e_2	$V_2 V_3$	V_2	$x_2 y_2 z_2$	f_2	$e_6 e_2 e_1$
e_3	$V_3 V_4$	V_3	$x_3 y_3 z_3$	f_3	$e_5 e_3 e_2$
e_4	$V_4 V_1$	V_4	$x_4 y_4 z_4$	f_4	$e_4 e_3 e_1$
e_5	$V_1 V_5$				
e_6	$V_2 V_6$				

Taulukko 1. Kuvan 1 nelitahokkaan särmäpohjainen malli [Mäntylä, 1988]

kolme listaa: solmu-, särmä- ja monikulmiolista. Solmulistassa on lueteltuna verkkoon kuuluvat solmut $\{V_1, V_2, \dots, V_n\}$. Jokaiselle solmulle on määritetty sijainnin avaruudessa määrittävä koordinaatti (x_n, y_n, z_n) . Särmälistassa on lueteltu yksittäiset särmät $\{e_1, e_2, \dots, e_n\}$. Särmätietueella on osoitin lähtö- ja tulokärkeen. Monikulmiolista luettelee monikulmiot $\{f_1, f_2, \dots, f_n\}$. Monikulmiotietueessa puolestaan on osoittimet solmuihin, jotka muodostavat monikulmion. Särmäpohjainen tietorakenne on yksinkertainen ja tilankäytön kannalta tehokas [Silva and Gomes, 2004], mutta sen heikkoutena on vaikeus selvittää topologia. Pahimmillaan vierekkäisen särmän selvittämiseen verkosta kuluu aikaa $O(n)$, sillä verkko ei sisällä eksplisiittistä topologiainformaatiota.

Tässä kohdassa esitettiin perusteoria monikulmioverkkoista ja yksinkertainen rakenne niiden esittämiseen. Seuraavaksi käsitellään topologian määrittelyä tarkemmin ja kehittyneempiä tietorakenteita monikulmioverkkojen esittämiseen. Monikulmioverkon roolina on vastata 3D-mallin rakenteeseen liittyviin kysymyksiin, joten pelkkä yksinkertainen esitys ei aina käytännössä riitä. Myös monikulmioverkkoa eri *tarkkuustasoilla* (level of detail, LOD) käsittelevät algoritmit tarvitsevat tehokkaan tavan käsitellä verkkoa topologisesti.

2.2. Monikulmioverkkojen topologiasta

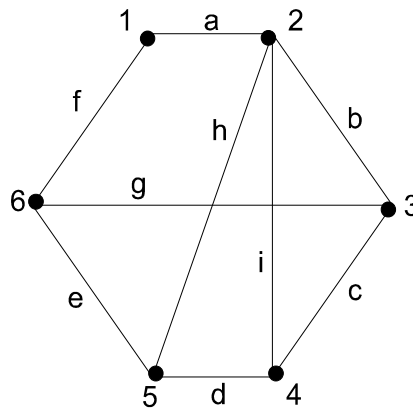
Monikulmioverkko voidaan esittää usealla tavalla. Ideaalinen tapa on esittää monikulmioverkon topologiset ominaisuudet niin, että ominaisuuksien perusteella voidaan suorittaa tehokkaasti topologisia hakuja. Toinen aspekti on topologisen informaation viemä tila, jonka tulisi olla mahdollisimman pieni. Usein nämä kaksi näkökulmaa ovat vastakkaiset, sillä malli, joka on tehokas topologisen informaation esittämiseen, ei välttämättä ole tilankäytöltään kompakti.

Monikulmioverkkojen yhteydessä topologiset ominaisuudet ovat suhteita kärkipisteiden, särmien ja monikulmioiden välillä [Mäntylä, 1988]. Monikulmioverkko onkin joukko topologisia relaatioita verkon elementtien välillä. Topologiset relaatiot eksplisiittisesti tallentavia tietorakenteita kutsutaan *rajaaviksi tietorakenteiksi* (boundary data structures, b-rep). Vastaava informaatio on mahdollista selvittää myös yksinkertaisemmista monikulmioverkkotietorakenteista. Implisiittinen esitys tarkoittaa usein heikentyneitä suorituskykyä.

Vierekkäisyys (adjacency) ja *kytkentä* (incidency) ovat perustavanlaatuisia ominaisuuksia topologisesti määritetyille rakenteille. Aiemman määritelmän mukaisesti särmä on jana, joka yhdistää kaksi solmua. Oletetaan, että särmän päätepisteet ovat solmut x ja y . Kyseiset solmut ovat tällöin *vierekkäisiä* (adjacent). *Vierekkäisyysrelaatio* (adjacency relation) ilmaistaan notaatiolla $x \sim y$, joka tarkoittaa, että x on vierekkäinen solmulle y .

Särmään yhdistetyn solmun sanotaan olevan *kytketty* (incident) kyseiseen särmään. Vastaavasti särmä on kytetty kahteen solmuun. Kytkettyvyyden perusteella voidaan määrittellä *kytkentärelaatio* (incidence relation). Vierekkäisyys- ja kytettärelaatioista voidaan käyttää merkintää $x \Leftrightarrow y$, joka tarkoittaa, että x on relaatiossa y :n kanssa. Kuvassa 5 on havainnollistettu vierekkäisyys- ja kytettämatrisilla tässä kuvattuja relaatioita. Vierekkäisyysmatriisin rivit ja sarakkeet kuvaavat verkon solmuja $\{1, 2, \dots, 6\}$. Matriisin alkioista AM_{12} selviää, että vierekkäisyysrelaatio on tosi solmuille 1 ja 2, sillä $AM_{12}=1$. Arvo 0 tarkoittaa, että relaatio ei ole voimassa. Kytkentämatriisin sarakkeina on verkon särmät $\{a, b, c, \dots, i\}$ ja riveinä solmut $\{1, 2, \dots, 6\}$. Solmuun 6 on kytetty särmät e, f ja g , joten $IM_{65}=1$, $IM_{66}=1$ ja $IM_{67}=1$.

Vierekkäisyysrelaatio on johdettavissa kytettärelaatiosta, mutta sen eksplisiittinen esittäminen johtaa parempaan suorituskykyyn. Kytkentärelaatio on *orientoitu* (orientable), jos se määrittää orientoidun järjestyksen kaikille mallin monikulmioille. Käytännössä kytettärelaation on määritettävä kaikkien monikulmioiden kulmapisteet ja särmät samassa kiertojärjestyksessä. Kuvassa 6 on esitetty nelitahokas ja sen topologia b-rep-esityksenä. Rakenteen alimmalla tasolla on nelitahokkaan kulmapisteet $\{V_1, \dots, V_4\}$. Jokainen kulmapiste on osa täsmälleen kolmea särmää kaikkien särmien joukosta $\{e_1, \dots, e_6\}$. Ylimmällä tasolla on esitetty monikulmiot, jotka taas koostuvat särmistä. Kappaleen topologia voidaankin ajatella hierarkkisenä rakenteena, jossa



$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

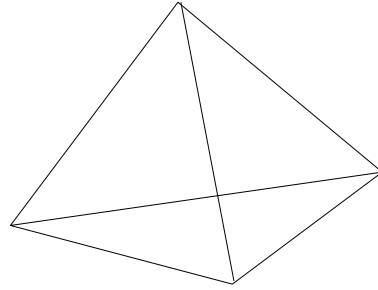
$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{pmatrix}$$

Vierekkäisyysmatriisi **AM**

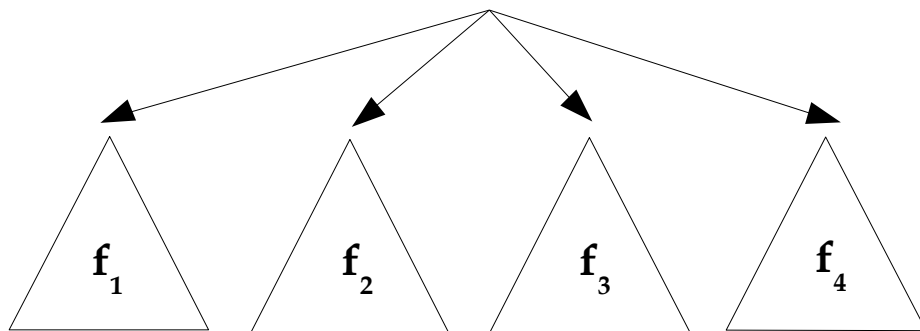
Kytettämatrisi **IM**

Kuva 5. Verkko, ja sen vierekkäisyys- ja kytettärelaatiot kuvattuna matriisilla

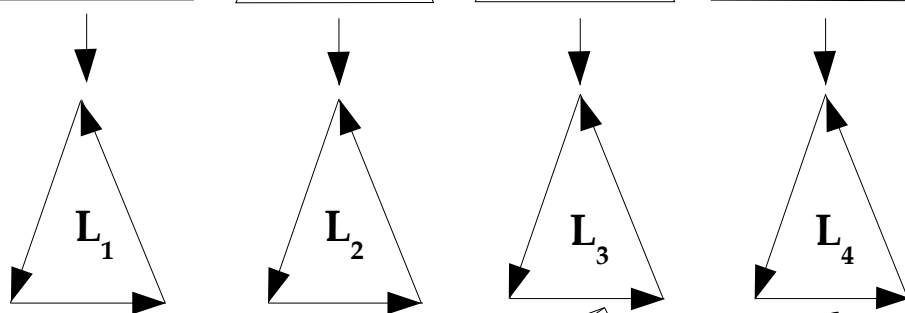
Kappale:



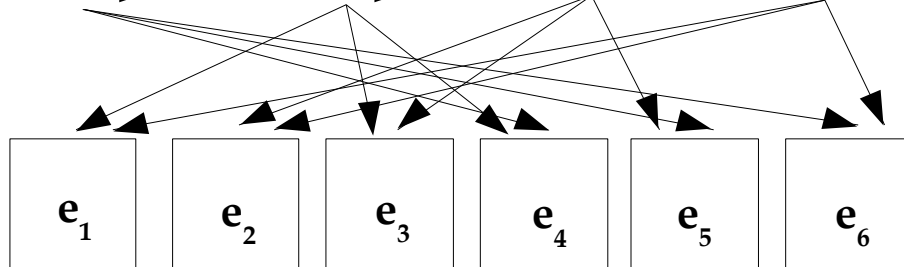
**Moni-
kulmiot:**



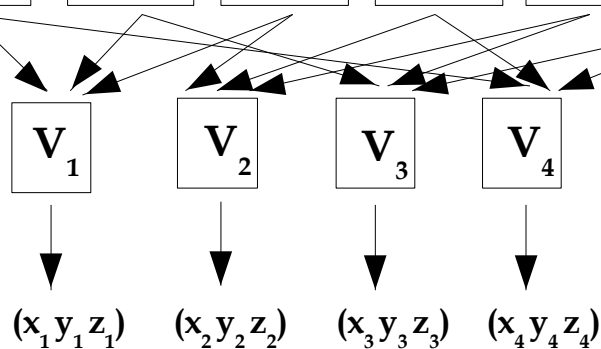
Silmukat:



Särmät:



**Kulma-
pisteet:**



Kuva 6. Nelitahokkaan b-rep-malli [Patrikalakis, 2003]

voidaan toisaalta navigoida ylä- ja alasuuntaan sekä rakenteen samalla tasolla. Vierekkäisyysrelaation ansiosta esimerkiksi särmästä voidaan selvittää sitä naapuroivat särmät. Toisaalta taas särmästä tiedetään monikulmiot, joiden osana särmä on.

2.3. Topogologisten monikulmioverkkojen vertailua

Monikulmioverkkojen topologiseen esittämiseen on kehitetty lukuisia tietorakenteita, joita voidaan vertailla tuettujen ominaisuuksien perusteella. Tietorakenne voi tukea ainoastaan kolmioverkkoja tai kaiken tyyppisiä monikulmioverkkoja. Osa tietorakenteista on rajoittunut ainoastaan 2-monistojen kappaleiden esittämiseen, kun taas osa tukee myös ei-monistojen kappaleita. Ideaalinen tietorakenne kykenee esittämään myös ei-monistot kappaleet, sisältää tuen kaiken tyyppisille monikulmioverkoille ja kuluttaa mahdollisimman vähän muistitilaa.

Yksinkertaisimpia rakenteita monikulmioverkoille on jo aiemmin esitetty monikulmiolista. Rakenteen avulla on vaikeaa hyödyntää vierekkäisyys- ja kytkentärelaatioita, sillä monikulmiolista koostuu yksinkertaisesti joukosta monikulmioita, joihin liittyy joukko solmuja. Topologista informaatiota ei ole esitetty eksplisiittisesti.

Yleinen tapa nopeuttaa topologisia kyselyitä on lisätä tietorakenteeseen erillinen orientoitu saantirakenne, jolla topologinen informaatio on tehokkaampaa selvittää. Tällaisia orientoituja rakenteita ovat winged-edge [Baumgart, 1972], half-edge [Mäntylä, 1988], radial-edge [Weiler, 1988], directed-edges [Campagna, 1998] ja FastMesh [Pajarola, 2001]. Directed-edges-rakenne eroaa muista orientoiduista rakenteista siten, että se tukee vain 2-ulotteisia kolmioverkkoja. Orientoitujen rakenteiden sisältämä saantirakenne on rasite muistinkäytölle, mikä voidaan tulkita haitaksi yksinkertaisempiin rakenteisiin verrattuna. Esimerkiksi radial-edge kuluttaa muistia noin kolme kertaa enemmän kuin monikulmiolista (taulukko 2).

Tri-edge [Loop, 2000] tukee sekin vain 2-ulotteisia kolmioverkkoja. Star-vertex [Kallmann, 2001] ei sisällä eksplisiittistä tietoa särmistä ja monikulmioista, joten vierekkäisyys- ja kytkentärelaatioiden selvittäminen on

Tietorakenne	Verkon tyyppi	Ei-monistot	Tavua / Δ
Monikulmiolista	Δ	Kyllä	18
Star-Vertex	Mikä tahansa	Kyllä	$10 + 4k$ ($22, k=3$)
Progressive Mesh	Δ	Ei	33
Tri-Edge	Δ	Ei	35
AIF	Mikä tahansa	Kyllä	$29 + 2k$ ($35, k=3$)
PSC	Δ	Kyllä	$37 + 2k$ ($43, k=3$)
Directed-Edges	Δ	Kyllä	44
Half-Edge	Mikä tahansa	Ei	46
FastMesh	Δ	Ei	53
Radial Edge	Mikä tahansa	Kyllä	56
Winged-Edge	Mikä tahansa	Ei	60

Taulukko 2. B-rep-rakenteiden kategorisointi tilankäytön ja ominaisuuksien mukaan [Silva and Gomes, 2004] Symboli Δ tarkoittaa kolmioverkkoa.

hidasta, vaikkakin rakenne on kompakti. PSC (Progressive Simplicial Complexes) [Popovic and Hoppe, 1997] ei sisällä eksplisiittisesti orientoituja pintoja eikä normaalivektoreita monikulmoille, joten sen renderöinti on käytännössä hankalaa.

Progressive mesh [Hoppe, 1998] on suunniteltu erityisesti sovellusalueille, joissa verkkoa halutaan käsitellä eri tarkkuustasoilla. Tietorakenne ei tue erityisen hyvin topologian selvittämistä, sillä kytkentärelaatio on määritelty eksplisiittisesti niin, että monikulmiot tietävät niihin kuuluvat solmut, muttei toisinpäin. AIF [Silva and Gomes, 2004] tukee sen sijaan eksplisiittisesti vierekkäisyys- ja kytkentärelaatioita sekä ei-monistoja kappaleita. Sen tilankäyttö on 25-50% pienempi verrattuna orientuihin tietorakenteisiin (kuten winged-edge ja half-edge, taulukko 2).

Taulukkoon 2 on koottu yhteenveto eri monikulmioverkkotietorakenteiden ominaisuuksista ja tilankäytöstä. Muuttuja k ilmaisee verkon tilankäytön ulottuvuudessa k . Monikulmioverkkojen tapauksessa k on kiinnitetty arvoon 3. Taulukossa omiksi ryhmikseen erottuvat kompaktit, mutta topologian kannalta heikot rakenteet (monikulmiolista, star-vertex). Toinen ryhmä on topologian kannalta paremmat, sekä keskikokoiset tietorakenteet (progressive mesh, tri-edge, AIF, PSC). Ääripään koon suhteen muodostavat orientoituneet rakenteet (directed-edges, half-edge, FastMesh, radial edge, winged-edge).

2.4. AIF, b-rep-rakenne monikulmioverkoille

AIF (Adjacency and Incidence Framework) [Silva and Gomes, 2004] on rajaava (b-rep) tietorakenne monikulmioverkkojen esittämiseen. Sillä voidaan esittää monistoja ja ei-monistoja kappaleita sekä kolmioverkkojen lisäksi monikulmioverkkoja. AIF kuluttaa vähemmän tilaa kuin orientoituneet rakenteet (taulukko 2).

Rakenne perustuu topologisten relaatioiden eksplisiittiseen esittämiseen. Oletetaan, että V vastaa solmua, E särmää ja F monikulmiota. Tällöin AIF määrittää vierekkäisyysrelaatiot $V \Leftrightarrow E$ ja $E \Leftrightarrow F$. Tämän lisäksi rakenne sisältää kytkentärelaatiot $E \Leftrightarrow V$ ja $F \Leftrightarrow E$. Näiden relaatioiden perusteella voidaan muodostaa kaikki Weilerin [1988] määrittämät yhdeksän topologista perusrelaatiota [Silva and Gomes, 2004]. AIF on siis topologisena rakenteena kattava.

AIF voidaan esittää tietorakenteellisesti viitenä luokkana: *Point*, *Vertex*, *Edge*, *Face* ja *Mesh*. Mesh on rakenteen ylin luokka, ja säiliö koko monikulmioverkolle. Se sisältää listan mallin kaikista monikulmioista, särmistä ja solmuista. Se on rakenteellisesti verrattavissa yksinkertaiseen kulmapistelistaan. Face kuvaa monikulmiota sisältäen vierekkäisyysrelaation särmiin. Face sisältää myös monikulmion normaalivektorin, jotta verkko on orientoitavissa. Edge eli särmä sisältää vierekkäisyysrelaation solmuihin ja

<pre># Piste class Point { double x; double y; double z; }</pre>	<pre># Solmu class Vertex { int id; # Solmuun # kytketyt särmät List<Edge> edges; Point point; }</pre>	<pre># Särmä class Edge { int id; # Viereiset solmut Vertex v1, v2; # Särmään kytketyt # monikulmiot List<Face> faces; }</pre>
<pre># Monikulmio class Face { int id; # Vierekkäiset särmät List<Edge> edges; # Monikulmion # normaalivektori Point normalVec; }</pre>	<pre># Monikulmioverkko class Mesh { # Mallin kaikki solmut List<Vertex> vertices; # Mallin kaikki särmät List<Edge> edges; # Mallin kaikki # monikulmiot List<Face> faces; }</pre>	

Taulukko 3. AIF-rakenteen luokkahierarkia [Silva and Gomes, 2004]

kytkentärelaation niihin monikulmioihin, joiden osa kyseinen särmä on. Rakenteen alin osa Vertex kuvaa yksittäistä solmua, jolla on kytkentärelaatio särmiin. Point kuvaa yksittäistä pistettä avaruudessa. Point on aina osa tiettyä solmua. Rakenne ei ole implisiittisesti orientoitu, mutta Silva ja Gomes esittävät algoritmin, jolla orientaatio voidaan laskea eksplisiittisesti koko verkolle.

3. Piilopintojen poistaminen

On tavallista, että monikulmioverkon monimutkaisuus kasvaa. Yksinkertainen malli saattaa muodostua yksittäisestä kappaleesta, mutta joissain malleissa voi olla jopa kymmeniätuhansia kappaleita, joita kaikkia käsitellään samassa avaruudessa. Yksinkertaisissa tapauksissa kappaleet voidaan jakaa pienemmiksi kokonaisuuksiksi, mutta se ei ole mahdollista kaikissa sovelluksissa. On siis tarpeen tutkia tekniikoita, joilla kappaleiden renderöimisen aiheuttamaa kuormaa saadaan pienennettyä, jotta renderöinti monimutkaisemmissa tapauksissa on mahdollista.

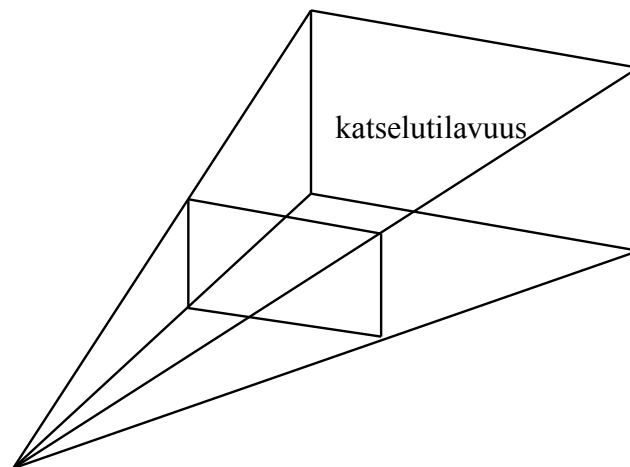
Yleisellä tasolla kyse on niiden pintojen piilottamisesta, jotka eivät katsojan näkökulmasta ole näkyvissä. Tätä voidaan tehdä monella tasolla. Joko renderöintivaiheessa, tai karsia jo ennen renderöintiä näkymättömät osat kappaleesta pois. Siirtämällä piilopintojen poisto jo osaksi tietorakenteen tehtäviä renderöintiprosessia saadaan kevennettyä, sillä monikulmioiden määrä vaikuttaa suoraan renderöinnin nopeuteen. Yleisin tapa poistaa piilopinnat on muodostaa sellainen erillinen saantirakenne, että sen avulla on

mahdollista poistaa katsojan näkökulmasta piilossa olevat monikulmiot. Quadtree- ja octree-variantit [Samet, 1989] ovat yleisimmin käytössä tietorakenteen tasolla tapahtuvassa optimoinnissa. BSP-puilla saadaan taas renderöinnin aikana pinnoille järjestys.

3.1. Katselutilavuuden rajaaminen

Katselutilavuuden rajaaminen (frustum culling) on osa renderöintiä. Ennen kuin 3D-malli voidaan nähdä, se on redusoitava kaksikulotteiseen tasoon. Prosessia kutsutaan perspektiiviprojektitioksi. Perspektiiviprojektiossa 3D-malli projisoidaan tietyn pisteen kautta sisältämään *katselutilavuuteen* (frustum) kuuluvat objektit. Katselutilavuus voidaan intuitiivisesti ajatella pyramidiksi, jonka kärki on katkaistu. Pyramidin mittasuhteet ja sijainti määrittävät lopulta projisoitavan kuvan mittasuhteet. Kuvassa 7 on havainnollistettu katselutilavuuden rajaamista. Katselutilavuuden rajaaminen saattaa tarjota mahdollisuuden geometrian piilottamiseen, sillä rajaamisen seurauksena joukko geometriasta voi jäädä näkymättömiin. Renderöintiprosessia voidaankin nopeuttaa jättämällä jo ennen renderöintiä katselutilavuuden ulkopuolinen geometria pois [Picco, 2003].

Katselutilavuuden ulkopuolisen geometrian rajaaminen voidaan toteuttaa niin, että kappaleiden rajaama avaruus jaetaan pienempiin aliavaruuksiin kappaleiden viemän tilan mukaan. Tällöin on kyse *rajauslaatikkohierarkiasta* (bounding box hierarchy), jossa ylimmän tason rajauslaatikko kattaa koko avaruuden, ja sen alaisuudessa olevat rajauslaatikot pienempiä yksittäisiä alueita. Usein jako suoritetaan niin, että avaruutta ei ole jaettu tasaisen kokoisiin rajauslaatikoihin, vaan rajauslaatikoiden koko vaihtelee geometrian monimutkaisuuden mukaan. Iso tyhjä alue on esimerkiksi kannattavaa esittää yhdellä rajauslaatikolla, koska sillä on merkitystä näkyvyyden rajaamisessa ainoastaan kokonaan näkyvänä tai piilotettuna alueena. Geometrisesti



Kuva 7. Katselutilavuuden rajaaminen [Shreiner et al., 2004]

```

class OctreeNode {
    # Solmun 8 lapsisolmua
    OctreeNode children[8];
    # Lista solmun sisältämistä monikulmioista
    List faces;
    # Solmun sisältämän kuution
    #keskipisteen koordinaatit
    int x, y, z;
    # Solmun sisältämän kuution leveys
    int width;
}

```

```

class Octree {
    # Puun juurisolmu
    OctreeNode root;
}

```

Algoritmi 1. Tietorakenne octree-pohjaisen rajauslaatikkohierarkian esittämiseen

DrawOctree (Node) :

```

# Älä piirrä solmua, jos solmu on tyhjä osoitin
if Node is NIL
    return
end if
# Jos solmu ei ole osa nykyistä katselutilavuutta, älä piirrä
if not Node in Frustum.Current()
    return
end if
if HasChildren(Node)
    DrawOctree(Node.children[TOP_LEFT_FRONT])
    DrawOctree(Node.children[Child.TOP_LEFT_BACK])
    DrawOctree(Node.children[Child.TOP_RIGHT_BACK])
    DrawOctree(Node.children[Child.TOP_RIGHT_FRONT])
    DrawOctree(Node.children[Child.BOTTOM_LEFT_FRONT])
    DrawOctree(Node.children[Child.BOTTOM_LEFT_BACK])
    DrawOctree(Node.children[Child.BOTTOM_RIGHT_BACK])
    DrawOctree(Node.children[Child.BOTTOM_RIGHT_FRONT])
end if
# Piirrä solmun geometria (lista monikulmioista)

```

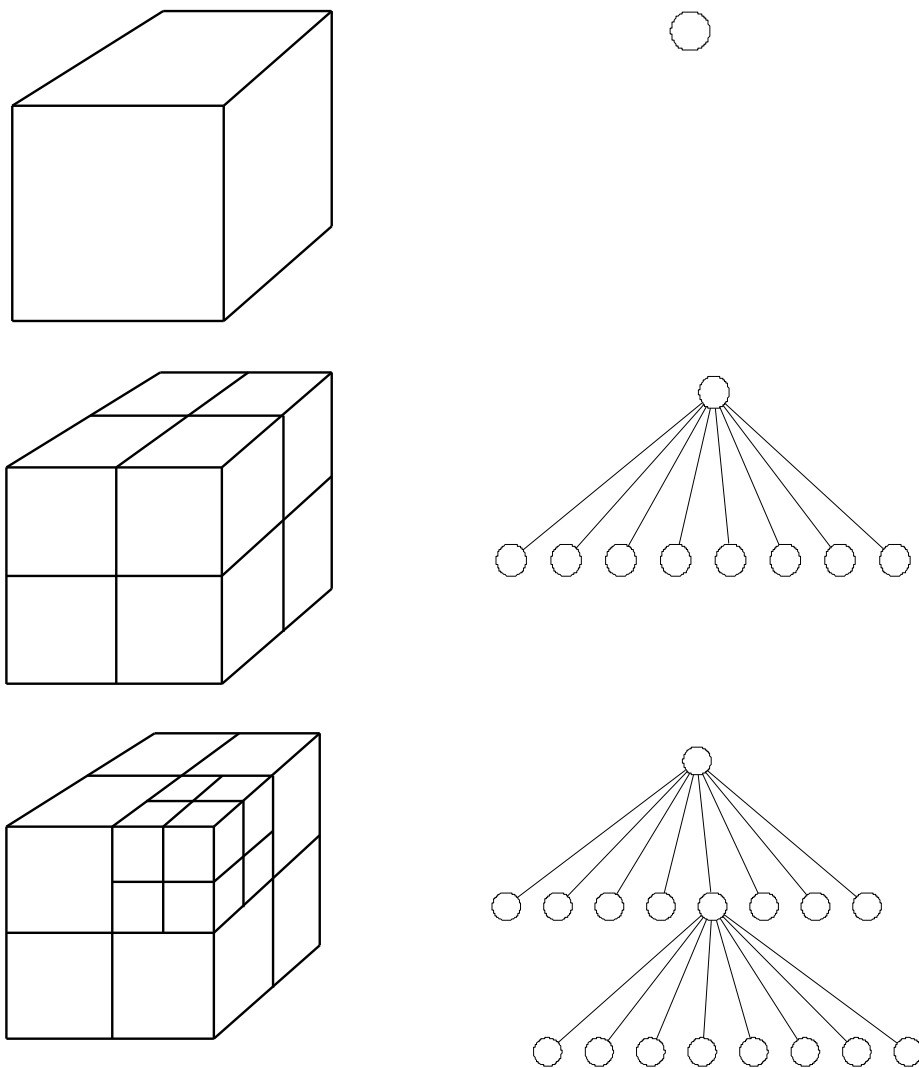
Algoritmi 2. Octree-rakenteen piirtäminen

monimutkaisempi alue voi tarjota useita näkyvyystasoja, jolloin sen pilkkominen useampaan rajauslaatikkoon on järkevää.

Rajauslaatikot esittäväälle tietorakenteelle on eduksi, että sillä on mahdollista selvittää mahdollisimman pienellä rasitteella katselutilavuuden leikkaamat rajauslaatikot. Hierarkkisten rakenteiden etu on, että koko avaruutta ei ole tarpeen käydä läpi leikkauspisteiden selvittämiseksi, vaan kun tietyn aliavaruussolmun tiedetään olevan näkymättömissä, myös sen lapset voidaan jättää näyttämättä. Riippuen puun tasapainoisuudesta menetelmällä päästään alle lineaarisen ajan, joka muuten olisi väistämätön kaikkien rajauslaatikoiden läpikäynnin myötä.

3.2. Rajauslaatikkohierarkia octree-rakenteella

Octree-rakenne on epätasapainoinen puu, joka jakaa avaruuden kuution muotoisiin aliavaruuksiin. Aliavaruudet vastaavat rajauslaatikkohierarkian rajauslaatikoita. Octreen juurisolmu kattaa koko avaruuden, ja sen alaisuudessa



Kuva 8. Kolmen jakotason octree-avaruus [Johnson, 2004]

olevista lapsisolmuista kukin kattaa koko sisemmän avaruuden tai jakaa sen kahdeksaan sisempään kuution muotoiseen aliavaruuteen. Aliavaruuksien koko pienenee jaon täsmentyessä. Kuva 8 havainnollistaa avaruutta, jossa on kolme jakotaso.

Tietorakenteellisesti octree on yksinkertainen esittää tasapainoittamattoman puun variaationa, jossa jokaisella solmulla on varattu kiinteän pituisella taulukolla tilaa 8 lapsisolmulle. Solmun sisältämät monikulmiot on esitetty osoittimia sisältävänä listana. Octreen aliavaruudet ovat kuutioita, joten niiden paikan esittämiseen riittää keskipiste ja leveys. Keskipisteen ja leveyden perusteella on myös yksinkertaista laskea rajaustaatikon tasot.

Octree-rakenteen piirtäminen tapahtuu algoritmien 1 ja 2 mukaisesti. Piirtäminen aloitetaan juurisolmusta, josta edetään puun alemmille solmuille. Rekursio pysäytetään kun solmun rajaustaatikko ei ole näkyvässä. Tämä on yhtäpätevää sen kanssa, että lapsisolmujen geometrian tiedetään olevan näkymättömissä, kun ylemmän tason rajaustaatikko ei näy. Rajaustaatikon ja katselutilavuuden välinen leikkaus voidaan laskea tehokkaasti kaksivaiheisella menetelmällä [Picco, 2003]. Menetelmä pohjautuu katselutilavuutta rajaavien pisteiden arviointiin laskemalla rajausta geometrisesti ensin pallolla ja tarvittaessa kuutiolla. Niiden avulla voidaan tarkistaa geometrinen leikkaus rajaustaatikoita vasten.

Octree-rakenteen heikkous on, että se on muodostettava uusiksi, jos geometria muuttuu. Puu on tasapainotettu geometrian monimutkaisuuden mukaan, joten geometrian muuttuessa myös octree on muutettava vastaamaan tapahtuneita muutoksia. Octreen alustaminen on kallis operaatio, sillä se edellyttää koko geometrian läpikäymistä. Usein octree onkin käytössä tilanteissa, joissa geometriaa ei luomisen jälkeen enää muuteta, vaan puu on laskettu osaksi esimääriteltä dataa, tai puu lasketaan korkeintaan kertaalleen. Tämä on tilanne usein esimerkiksi pelien tapauksessa. Muuttuva geometria on toki myös mahdollista jättää octreen ulkopuolelle ja huolehtia erikseen sen rajaamisesta.

Tämä luku tarjosi menetelmän monimutkaisten mallien näyttämisen nopeuttamiseen hierarkkisilla octree-rakenteella. Rajaustaatikkohierarkiat korostuvat erityisesti, kun mallien koko kasvaa, sillä ilman rajaamista näytettävän geometrian koko kasvaa helposti liian suureksi.

4. Yhteenveto

Kolmiulotteisen geometrian esittämiseen tarvitaan yhä kehittyneempiä rakenteita, sillä 3D-mallien koko ja monimutkaisuus on alati kasvanut. Samalla on myös syntynyt uusia haastavia tarpeita mallien reaaliaikaiseen käsittelyyn.

Tässä tutkielmassa esitetyt b-rep-rakenteet ovat osaltaan vastaus näihin tarpeisiin.

B-rep-malleista klassisen aseman ovat saavuttaneet orientoidut rakenteet winged-edge [Baumgart, 1972] ja half-edge [Mäntylä, 1988]. Niille on esitetty myös uudempia vastineita, kuten progressive mesh [Hoppe, 1998] ja AIF [Silva and Gomes, 2004]. B-rep-rakenteen hyvyttä voidaan mitata sen tukemien ominaisuuksien perusteella (kuten tukeeko malli ei-monistoja kappaleita tai tukeeko malli pelkästään kolmioverkkoja). Usein vastakkainen puoli mallin ominaisuuksille on sen tilankäyttö, joka on varsinkin orientoitujen b-rep-rakenteiden ongelma. B-rep-mallin käyttö on myös suhteutettava käyttöympäristöön, kuten OpenGL-rajapinnan [Shreiner et al., 2004] asettamiin käyttötapauksiin.

Piilopintojen poistamisella voidaan vähentää monikulmioverkon renderöimiseen liittyvää kuormaa, sillä rajaaminen jättää katselutilavuuden ulkopuoliset osat pois renderöinnistä. Piilopintojen ennakoiva poistaminen voidaan toteuttaa esimerkiksi octree-rakenteella [Picco, 2003]. Piilopintojen rajaaminen on hyödyllistä etenkin isoille monikulmioverkoille.

Viiteluettelo

- [Baumgart, 1972] B. G. Baumgard, Winged-edge polyhedron representation. Technical report, STAN-CS-320, Stanford University, 1972.
- [Bikker, 1999] Jacco Bikker, Building a 3D Portal Engine http://www.flipcode.com/articles/portals_issue01.shtml (Checked 23.2.2005).
- [Campagna, 1998] Swen Campagna, Leif Kobbelt and Hans-Peter Seidel. Directed edges - a scalable representation for triangular meshes, *Journal of Graphics Tools* 4, 3 (1998) 1-12.
- [Hoppe, 1998] Hugues Hoppe. View-Dependent Refinement of Progressive Meshes. Technical Report MSR-TR-98-02, Microsoft Research, 1998.
- [Johnson, 2004] Maggie Johnson. CS148, Introductory Computer Graphics (Aut 04-05) <http://www.stanford.edu/class/cs148/handouts/20.HSR-short.pdf> (Checked 9.3.2005)
- [Kallmann, 2001] Marcelo Kallmann and Daniel Thalmann, Star-vertices: a compact representation for planar meshes with adjacency information, *Journal of Graphics Tools* 6, 1 (2001), 7-18.
- [Loop, 2000] Charles Loop. Managing Adjacency in Triangular Meshes. Technical Report MSR-TR-2000-24, Microsoft Research, 2000.
- [McGuire, 2000] Max McGuire. The Half-Edge Data Structure. http://www.flipcode.com/articles/article_halfedge.shtml (Checked 23.1.2007)

- [Mäntylä, 1988] Martti Mäntylä, *An Introduction to Solid Modeling*. Computer Science Press, 1988.
- [Pajarola, 2001] Renato Pajarola. FastMesh: Efficient View-dependent Meshing. In: *Proceedings of the Pacific Graphics 2001*, 22-30.
- [Patrikalakis, 2003] Nicholas M. Patrikalakis, Computational Geometry http://ocw.mit.edu/NR/rdonlyres/Ocean-Engineering/13-472JComputational-GeometrySpring2003/0D6F531E-437F-493F-91B5-6B5452881293/0/lecnotes1_fixed.pdf (Checked 9.3.2005)
- [Picco, 2003] Frustum Culling, Dion Picco http://www.flipcode.com/articles/article_frustumculling.shtml (Checked 5.5.2005).
- [Popovic and Hoppe, 1997] Jovan Popovic and Hugues Hoppe. View-Dependent Refinement of Progressive Simplicial Complexes. *Computer Graphics* 31 (1997), 217-224.
- [Rogers and Earnshaw, 1991] David F. Rogers and Rae A. Earnshaw, State of the Art in Computer Graphics. In: David F. Rogers and Rae A. Earnshaw (eds.), *Visualization and Modeling*. Springer-Verlag, 1991, 225 - 269.
- [Samet, 1989] Hanan Samet, *Applications of Spatial Data Structures*. Addison-Wesley, 1989.
- [Shreiner et al., 2004] Dave Shreiner, Mason Woo, Jackie Neider, and Tom Davis, *OpenGL Programming Guide*. Addison-Wesley, 2004.
- [Silva and Gomes, 2004] Frutuoso Silva and Abel Gomes, A b-rep data structure for polygonal meshes. In: Adérito Marcos and Miguel Salles Dias (eds.), *Revista VIRTual Special edition "Advances in Computer Graphics in Portugal"*, 2004, 1-8.
- [Weiler, 1988] Kevin Weiler, The radial edge structure: a topological representation for non-manifold geometric boundary modeling. In: M. Wozny, H. McLaughlin and J. Encarnacao (eds.), *Geometric Modelling for CAD Applications*. North-Holland, 1988, 3-36.

Yhteisölliset verkkopalvelut

Jarno Ojala

Tiivistelmä.

Tutkielmassani tarkastelen yhteisöllisten WWW-palveluiden roolia osana internetin erilaisten palveluiden kirjoa. Tarkoitus on myös kertoa verkkopalveluiden sijoittumisesta yksittäisen käyttäjän sosiaaliseen elämään verkossa ja myös verkon ulkopuolella.

Avainsanat ja -sanonnat: WWW, sosiaaliset verkkopalvelut, sosiaaliset verkostot, virtuaaliyhteisöt online-yhteisöt MySpace, yhteisöllisyys, IRC-galleria, Last.fm, YouTube.

CR-luokat: J.4., J.5., K.4.

1. Johdanto

Aina internetin alkuajoista lähtien sitä on käytetty myös sosiaaliseen kanssakäymiseen. Yhteisölliset WWW-palvelut tarjoavat uudenlaista sosiaalista vuorovaikutusmahdollisuutta käyttäjilleen. Web 2.0 -palveluiden määritelmän mukaan (O'Reilly, 2005) ne aktivoivat käyttäjiä ja antavat mahdollisuuden oman sisällön aktiiviseen luomiseen passiivisen lukemisen sijaan.

Tutkielmani on kirjallisuuskatsaus aiheesta. Aiheeseen liittyvää tilastotietoa tai tutkimuksia on vielä – aiheen suhteellisesta uutuudesta johtuen – erittäin vähän saatavilla. Lähdemateriaalina on käytetty myös itse palveluiden verkkosivuja ja niiden mahdollisesti tarjoamaa koostettua tilastotietoa.

Luvuissa 3-5 otetaan lähempään tarkasteluun yksittäisiä yhteisöllisiä palveluita. Luvut 6-8 tutkielman lopussa kokoavat havaitut ydinasiat palveluista.

2. Yhteisölliset WWW-palvelut

2.1. Uusi verkko

Internet on muuttumassa uudenväliseen aikaisempaa vuorovaikutteisempaan suuntaan. Web 2.0:n ideologian mukaan käyttäjät luovat aktiivisesti omaa sisältöä sen sijaan, että he passiivisesti ”kuluttaisivat” verkkoon luotua materiaalia. Web 2.0 on uudenlainen ajattelutapa, ei niinkään uutta teknologiaa (O'Reilly, 2005). O'Reilly näkee yleisön suhtautumisen verkkoon muuttuvan niin, että tulevaisuudessa verkko on enemmänkin alusta, jolle käyttäjät itse luovat sisällön. Yhteisölliset WWW-palvelut ovat tässä mielessä hyvä esimerkki Web 2.0 -aikakauden sovelluksista. Palvelun tarjoaja on luonut teknologisen

puitteen, tai alustan, jonka avulla käyttäjät voivat luoda sisältöä, eli tässä tapauksessa yhteisöjä ja verkostoja. Palvelun tarjoajan ei sinänsä tarvitse tuottaa sisältöä itse, vaan se syntyy "itsestään" palvelua käytettäessä. O'Reillyn periaatteiden mukaan palvelu "paranee mitä enemmän sitä käytetään".

2.2. Virtuaaliset yhteisöt

Yhteisö voidaan muodostaa välineeksi jonkin päämäärän tavoittelussa. Ihmisille on luontaista kuulua yhteisöön ja yhteisöllisyyden tunne sitoutuu vahvasti ihmisen sosiaaliseen perusolemukseen. (Mäyrä, 2002). Tämä oletamus toistuu myös verkossa tapahtuvassa kanssakäymisessä.

Yhteisö voi tarjota jäsenelleen eri vahvuisia verkostoja. Verkoston vahvuus koostuu muun muassa sen velvoittavuudesta, normeista, säännöistä ja toimintarakenteesta (Wellman & Gulia, 1999). Virtuaalisissa yhteisöissä nämä verkostot ovat yleensä suhteellisen heikkoja, koska liittyminen ja eroaminen on yleensä yksinkertaista, eikä vaadi erityisiä ponnisteluja. Myös sääntöjen velvoittavuus on usein jäsenen itsensä päätettävissä. Toisaalta virtuaalisten yhteisöjen jäsenet voivat tietyissä tapauksissa olla äärimmäisen omistautuneita asialleen ja noudattaa yhteisön sääntöjä ja normeja tunnollisesti ilman rangaistuksen uhkaa tai palkitsemisen toivoa. Virtuaalisiin yhteisöihin kuuluu usein myös vapaaehtoinen työ yhteisön hyväksi. Tällainen pyyteeton työ voi käytännössä tarkoittaa vaikkapa tietokoneen tekniseen puoleen liittyvää opastusta ja neuvojen antamista keskustelufoorumille vapaa-ajalla.

Internet on määritellyt uudestaan paikan ja paikallisuuden käsitteen. Yhteisöllisten palveluiden kautta oma hengentuote tai idea, olipa se sitten sävellyks, videolle kuvattu puhe tai päiväkirjamerkintä voidaan sekunneissa siirtää palveluun, josta sen voi nähdä ympäri maailman lukematon määrä ihmisiä. Ilmainen tiedotus tai mainonta on myös mahdollista tämän verkoston kautta. Tämä antaa uudenlaisen mahdollisuuden itsensä ilmaisemiseen suurellekin joukolle omin avuin, ilman ulkopuolista jakelijaa tai markkinoijaa.

Uuden yhteisöllisyyden aikana virtuaaliyhteisöistä on tullut myös liiketoimintaa. Heikkoakin virtuaalista yhteisöllisyyttä voidaan myydä kuten mitä tahansa tuotetta. Kuten jo aikaisemmin mainittiin, useiden yhteisöllisten palveluiden ideana on, että yhteisö ja sen jäsenet tuottavat kaiken sisällön itsenäisesti. Tämä idea miellyttää tietenkin myös niitä, jotka haluavat päästä hyötymään ilmiöstä taloudellisesti. Hyvän alustan ja mainostilan toteuttamisen lisäksi palveluntarjoaja voi seurata sivusta, kun palveluun rakentuu sisältö kuin taikavoimasta. Vaikka jäsenyys yhteisössä olisi ilmaista, mainostilasta ollaan valmiita maksamaan sen jälkeen, kun tietty kriittinen massa käyttäjämäärissä on ylitetty.

3. MySpace.com

MySpace on kuukausittaisissa kävijämäärissä mitattuna tämän hetken suurin verkkosivusto maailmassa. MySpace ohitti kävijämäärissä Yagoon marraskuussa 2006, jolloin koko kuun kävijämääräksi mitattiin noin 38,7 miljardia (Holahan, 2006). Tutkimuksessa kävijämäärä laskettiin yksittäisinä sivulatauksina. Kaiken kaikkiaan MySpacen kasvu on ollut räjähdysmäistä aina sen perustamisesta, vuodesta 2003, lähtien. Tällä hetkellä kasvu näyttää jatkuvalta ja tämän lisäksi MySpace-palveluun liitetään jatkuvasti uusia ominaisuuksia. Tästä esimerkkinä voidaan mainita artistiprofiileille lisätty omien, mp3-muotoisten musiikkikappaleiden myymistöiminto.

MySpacen perustivat vuonna 2003 Tom Anderson ja Chris DeWolfe. MySpacen alkuperäinen tarkoitus oli toimia Los Angelesin alueen musiikkiyhteisön verkottumis- ja yhteydenpitovälineenä (Sellers, 2006). MySpace kasvoi kuitenkin nopeasti mittoihin, joita tuskin kukaan osasi odottaa. MySpace siirtyi 2006 Rupert Murdochin mediayhtiön omistukseen 580 miljoonan dollarin kauppahinnalla (Sellers, 2006).

MySpacen alkuperäinen idea oli luoda pohja yhteisölliselle palvelulle, johon käyttävät voivat lisätä vapaasti lähes mitä tahansa materiaalia. Tom Anderssonin mukaan käyttäjien toimintaa pyrittiin rajoittamaan mahdollisimman vähän (Sellers, 2006).

3.1. Yhteisöt ja käyttäjät

Kaikki toiminta MySpacessa alkaa ja loppuu käyttäjien profiileihin. Käyttäjä luo ensimmäiseksi itselleen kohdesivun MySpaceen, jonka jälkeen hänen on mahdollista alkaa luomaan kontakteja muihin käyttäjiin. Kontakteja voidaan luoda pyytämällä toisia käyttäjiä "kavereiksi". Kohdesivun luonti on ilmaista, kuten myös suurin osa toiminnoista MySpacessa.

Profiilit on MySpacessa jaettu kahteen eri luokkaan: artistiprofiilit ja tavalliset käyttäjäprofiilit. Artistiprofiilin erikoisuutena on omien musiikkitiedostojen lisäämismahdollisuus. Omat musiikkitiedostot tarkoittavat tässä tapauksessa artistin itse säveltämiä kappaleita, joihin hänellä itsellään on siis tekijänoikeudet. Artistiprofiilit erotellaan muista profiileista palvelussa myös toimintalogisella tasolla, sillä käyttäjien on mahdollista esimerkiksi estää artistien "friend requestit" eli ystävyyspyynnöt. Konkreettisia yhteisöjä ei voi MySpacessa muodostaa tai nimetä. Yhteisöt ja verkostot syntyvät pelkästään ystävien kautta.

MySpacen käyttäjät ovat tiettyssä määrin uskollisia palvelulle: keskimääräinen käyttäjä viettää kuukauden aikana yhden tunnin ja 45 minuuttia palvelussa (Williams, 2005). Tämä voi kuulostaa melko lyhyeltä ajalta palvelun käyttämi-

selle, mutta keskimääräisenä käyttöaikana tämä luku on omasta mielestäni kohtuullisen suuri.

Kuten aiemmin tutkielmassani kerrottiin, MySpacen yhdeksi perusideaksi on mainittu, että käyttäjien toimintaa rajoitettaisiin mahdollisimman vähän. Tämä aiheuttaa tietysti ongelmia mm. artistien virallisten profiilien ja faniprofiilien erottamisessa. Periaatteessa kuka tahansa voi profiilissaan esiintyä kenen tahansa nimellä ja kuvalla. Palvelun laajuudesta johtuen profiilien todenperäisyyden tarkistaminen on jokseenkin mahdotonta. Ongelma ei yllättäen kuitenkaan ole oman kokemukseni mukaan niin suuri kun voisi kuvitella, vaan viralliset profiilit löytää kohtuullisen helposti.

3.2. MySpacen kaupallisuus vastaan idealismi

MySpacea on kritisoitu myös siitä, että se ei ole todellisuudessa pelkästään vapaa yhteisöllinen palvelu, vaan pohjimmiltaan vain virtuaalinen "mainostustila". Luomalla profiilin MySpaceen käyttäjät tietysti mielessä vastaanottavat lähes kaikenlaista mainostusta kaverien kommenttien lisäksi.

Myös MySpacen alkuperästä ja siihen liittyvästä "tuhkimotarinarasta" on esitetty epäilyksiä. Intermix-yhtiöllä väitetään olevan yhteyksiä vanhempaan eUniverseen liiketoimintaan, jossa harjoitettiin suoramarkkinointia verkossa mm. sähköpostin välityksellä (Lapinski, 2006). Lapinskin perustelut tosin jäävät kovin hatariksi. Näkökanta on silti mielenkiintoinen ja roskapostin saavutettua jokin aika sitten määrässä "hyödyllisen" postin, asia on myös kriittinen ja ajankohtainen. Tom Andersonin roolia MySpacen synnyssä epäillään myös artikkelissa.

Totta tai tarua, MySpace on todellakin saatu kaupallistettua erittäin tehokkaasti. Käyttäjämäärät ovat massiivisia, joten mainostila sivustolla on haluttua ja mainosten pikselihinta nousee tähtitieteellisen korkeaksi. Vaikka MySpace toimiikin tehokkaana markkinointi- ja tiedotuskanavana esimerkiksi aloitteleville yhtyeille ja muunlaisille taiteentekijöille, on myös vähemmän toivottu massamainonta mahdollista sivuilla.

4. IRC-galleria

IRC-galleria on suomalainen versio yhteisöllisestä verkkopalvelusta. IRC-gallerian on alun perin perustanut Tomi Lintelä joulukuussa 2000 *IRC:ssä*, eli *internet relay chatissa* keskustelevien henkilöiden yhteydenpitotarkoituksiin. IRC-gallerian piti kerätä *irkkaa* kuvia yhteen paikkaan, jotta muut voisivat nähdä minkälaisia henkilöjä IRC-lempinimien eli *nickien* takana on (<http://irc-galleria.net/irc-opas.html>). Nykyään palvelu on kuitenkin laajentanut koko

Suomen nuorison yhteiseksi, ja IRC-keskustelijat ovat vain pieni osa käyttäjiä.

IRC-galleria mitattiin Pohjoismaiden vilkasliikenteisimmäksi verkkopalveluksi 2006 viikolla 36. Tutkimuksen suoritti TNS Gallup ja viikon aikana sivuilla mitattiin yhteensä 326 miljoonaa sivulatausta (Karvonen, 2006). IRC-galleria on profiloitunut vahvasti teini-ikäisten ja nuorten aikuisten palveluksi. IRC-gallerian käyttäjien keski-ikä onkin 19,3 vuotta, ja käyttäjien ikäluokista suurin sijoittuu ikävuosille 15-17 (IRC-galleria, 2007). Paikkakunnittain järjestettynä käyttäjiä on eniten Helsingistä ja toiseksi eniten Tampereelta.

IRC-gallerian toiminnasta vastaamaan perustettiin 2003 maaliskuussa Dynamoid Oy. Yhtiön myötä mahdollistui myös maksullisten palveluiden lisääminen sivustolle. Maksullisia palveluita ovat VIP-jäsenyys ja erilaiset taustakuvien ja *ihkusälän* tilausmahdollisuudet. Ihkusälä on erilaisia liikkuvia graafisia tavaroita tai hahmoja, joita on mahdollista lisätä oman profiilinsa kohdesivulle. IRC-galleria on otettu suomalaisena konseptina käyttöön myös muissa maissa. Näistä mainittakoon esimerkkeinä virolainen "minugalerii.ee" ja venäläinen "my-gallery.ru", jotka ovat myös grafiikaltaan ja asetteluiltaan hyvin samannäköisiä kuin IRC-galleria.

4.1. Käyttäjät

IRC-galleriassa käyttäjät luovat itselleen profiilin, jonka tulee sisältää ainakin yksi kuva käyttäjästä itsestään. Palveluun liittymisen yhteydessä suostutaan noudattamaan palvelun sääntöjä, jotka vaativat muun muassa *oletuskuvan*, eli profiilin etusivulla näkyvän kuvan, olevan sellainen, josta käyttäjän kasvot on helposti tunnistettavissa. Mikäli käyttäjällä on käytössään "luotettava sähköpostiosoite" [WWW.irc-galleria.fi], profiilin luominen on ilmaista. IRC-galleria tarjoaa myös maksullisia VIP-ominaisuuksia, joihin kuuluu mm. kattavampi haku, yksityiskommentit ja oman profiilin muokkaaminen eli *modaus*.

Kommunikointi IRC-galleriassa on asynkronista. Toisten käyttäjien profiileiden kohdesivuille on mahdollista jättää kommentteja, mikäli itse on kirjautuneena sisään. Vaikka kommentit näkyvät muille käyttäjille reaaliaikaisesti, ei kommunikaatioita voi pitää synkronisena, koska yhteistä "keskustelutilaa" ei ole. Selain on päivitettävä, jotta näkisi esimerkiksi omalle kohdesivulleen lähetetyn uuden kommentin. Tässä mielessä IRC-galleria ei ole erityisen käytännöllinen palvelu keskusteluun verkossa, hieman harhaanjohtavasta nimestäänkin huolimatta.

Käyttäjillä on mahdollisuus myös kirjoittaa omaa päiväkirjaansa IRC-galleriaan blogityylisesti oman profiilinsa yhteyteen. Päiväkirjamerkinnot voivat olla yksityisiä tai julkisia.

4.2. Yhteisöt

Käyttäjät voivat liittyä IRC-galleriassa erilaisiin yhteisöihin. IRC-galleria tarjoaa kohtuullisen hyvän tilasto-osion, jossa on koottu yhteen tietoja käyttäjämääristä ja yhteisöistä. Yhteisöt voidaan jakaa karkeasti kahteen osaan: IRC-kanavien käyttäjien yhteisöt ja muut. IRC-kanavien yhteisöihin liittyvät siis henkilöt, jotka käyttävät, tai ovat joskus käyttäneet kyseisiä kanavia irkatessaan. Muut yhteisöt on perustettu muun muassa harrastusten, paikkakuntien tai humorististen tai vakavien elämänkatsomuksien ympärille.

Yhteisöt on palvelussa luokiteltu 25 eri kategoriaan. Luokittelu on tehty helpottamaan eri yhteisöjen luoman kokonaisuuden hahmottamista, ja toisaalta myös helpottamaan yhteisöjen perusteella tapahtuvaa hakemista. Kategoriat ovat hyvin vaihtelevia, mutta kategorisointi on erittäin kattava. Suurin yhteisö IRC-galleriassa koostuu yli 27 000 käyttäjästä, kun taas suurin IRC-kanavan yhteisö koostuu alle 700 henkilöstä (IRC-galleria, 2007).

5. Muita yhteisöllisiä palveluita

Last.fm on erikoistunut musiikin ympärille muodostuviin yhteisöihin. Erikoisuutena Last.fm:ssa on ominaisuus, jolla seurataan Winampissa soitettujen kappaleiden tietoja. Soitetut kappaleet kirjataan profiiliin, minkä lisäksi käyttäjän on vielä itse mahdollista lisätä omia suosikkiartistejaan ja kappaleitaan. Last.fm suosittelee soitettujen kappaleiden perusteella käyttäjälle samantyyllisiä artisteja. Palvelu näyttää samantyyllisestä musiikista kiinnostuneet ”naapurit”. Tämän lisäksi myös yhteisöjen muodostaminen artistien, kappaleiden tai vaikka musiikkitapahtumien ympärille on mahdollista.

Youtube tarjoaa käyttäjilleen mahdollisuuden jakaa videoita, joihin heillä on tekijänoikeudet. Videoiden kommentointi, linkittäminen ja jopa videovastauksien lähettäminen ovat olennaisia osia puhuttaessa Youtuben yhteisöllisyydestä. Videovastaus tarkoittaa videomuodossa lähetettyä kommenttia videolle, tai samaan aihepiiriin liittyvän videon tai ns. vastinevideon linkittämistä Youtube tarjoaa siis uudenlaisen kommunikointi- ja ilmaisutavan yhteisölliseen toimintaan ja vuorovaikutukseen verkossa: videomuotoisen viestinnän. Tässä suhteessa tarkasteltuna Youtube täyttää verkkoyhteisön määritelmästä useita kriteerejä (Järvinen et al., 2002). Konkreettisia yhteisöjä myöskään Youtube ei tue, mutta videoiden ympärille syntyvä keskustelu ja yleinen jakamisideologia jo sinänsä antavat palvelulle yhteisöllisen leiman.

6. Palveluiden tarkastelua

6.1. Yhteisöjen vetovoima

Yhteisölliset WWW-palvelut toimivat tavallaan vastavoimana erilaisten keskustelupalstojen, IRC:n ja chattien käyttäjien identiteettittömyydelle. Kuulumalla erilaisiin yhteisöihin, kuten IRC-galleriaan, voi viestittää muille käyttäjille omia kiinnostuksen kohteitaan. Tällä tavoin voi määrittää omaa identiteettiään verkossa ja samalla voi tavata muita samoista asioista kiinnostuneita.

MySpacen ja IRC-gallerian tapaisilla palveluilla on sanottu olevan myös suurta tukea nuorten identiteetin ja minäkuvan rakentamisessa. Sean Rapackin mukaan blogien ja bulletinien kirjoittaminen MySpaceen voi opettaa nuoria ilmaisemaan ja purkamaan omia tuntejaan. Bloggaamisen Rapacki toteaa myös olevan nykyajan vastine ja luonnollinen jatkumo perinteiselle päiväkirjalle (Rapacki, 2007).

Virtuaalisten yhteisöjen jäseneksi liittyminen on helppoa. Toisaalta myös eroaminen yhteisöstä käy yhtä helposti, ja tästä syystä omistautuminen tai yhteisön mahdollisten sääntöjen noudattaminen ei aina välttämättä toteudu.

6.2. Julkisuus ja kuuluisuus verkossa

Yhteisölliset WWW-palvelut ovat luoneet myös uuden kuuluisuuden muodon. ”Nettijulkikkiset” ovat esimerkiksi MySpacea käyttäviä henkilöitä, jotka useat tuntevat käyttäjänimeltä. Tällaisen henkilön tai jopa kuvitteellisen profiilin erikoisuus on siinä, että hänen julkisuutensa perustuu kokonaan verkossa tapahtuvaan toimintaan. Tällainen henkilö saattaa nousta myös nettiyhteisöön kuuluvien henkilöiden tietoisuuteen jostain muuta kautta, vaikkapa esiintymällä tv-ohjelmassa. Samanlainen ilmiö toistuu myös musiikin verkkolevittämisessä. MySpacessa paljon kuunteluja saanut ja huomiota herättänyt ”autotaliityte” saattaa herättää suuren levy-yhtiön huomion, vaikka ei olisi koskaan tehnyt yhtään keikkaa! Tämä kertoo jostain yhteisöllisten palveluiden voimasta promootiokanavana varsinkin pienille ja riippumattomille taiteen tekijöille.

Kuten aiemmin mainittiin, MySpacen tai YouTuben kaltaiset palvelut ovat markkina-arvoltaan erittäin suuria. Silti suurin osa palveluiden ”sisällöstä” tuotetaan vapaaehtoisin voimin. Tästä herääkin kiinnostava kysymys, joka jätettävään käsittelemättä muuta kuin ajatuksen asteella: eli tulisiko käyttäjien, jotka tosiasiallisesti luovat palvelun sisällön, saada jonkinlainen osuus tuottamistaan summista tai muunlaista tunnustusta? Tuotteliaimmat ja eniten katselijoita tai kuuntelijoita keränneet riippumattomat tuottajat voitaisiin palkita vaikkapa vuosittain jollakin palkinnolla, jonka saajat yhteisö saisi äänestää itse.

6.3. Ystävät ja identiteetti verkossa

Vaikka MySpacen kaltaiset palvelut tarjoavat mahdollisuuden maailmanlaajuisen yhteydenpitoon, voidaan kuitenkin sanoa, että osalla käyttäjistä IRC-gallerian yhteisöt sekä MySpace-ystäväverkostot muodostuvat suurelta osin verraten lähellä asuvista ihmisistä. Vaikka nämä ihmiset eivät olisi muulla tavoin tuttuja, on yllättävää, että paikallisuus sitoo yhteen tällaisessa paikasta riippumattomassa palvelussa. Tästä esimerkkinä voidaan mainita vaikka IRC-gallerian yhteisöt, jotka muodostuvat paikkojen tai alueiden ympärille. Myös MySpacessa on mahdollista etsiä ystävikseen ihmisiä samalta alueelta. Tässä mielessä myös yhteisöllisissä palveluissa voidaan nähdä tiettyä paikkakeskittyneisyyttä (Heinonen, 2000). Tämän voidaan sanoa kertovan "oikean elämän" ystävyysuhteiden ja verkon online-suhteiden limittymisestä. Verkossa tavattu ystävä usein halutaan nähdä myös kasvotusten.

Osa käyttäjistä haluaa kuitenkin tehdä selvän erottelun online-ystävien ja muiden ystävien välillä. Online-tuttujen kanssa jutellaan ja vaihdetaan ajatuksia verkossa, mutta asia halutaan pitääkin näin. Muiden ystävien kanssa verkkopalveluita saatetaan käyttää pelkästään kasvokkain tapahtuvien kohtaamisten sopimiseen (Wellman, 1997).

Samaan virtuaaliseen yhteisöön kuuluvat henkilöt ovat usein tietyllä tavoin paljon läheisempiä, kuin samassa kaupunginosassa ja vaikkapa naapurissa asuvat ihmiset. Virtuaaliyhteisöissä ihmisiä yhdistävät samat kiinnostuksen aiheet tai mielipiteet, kun taas naapureita saattaa yhdistää pelkkä asuinpaikka (Wellman, 1997).

Verkkoyhteisöllisyys voi olla huomattavasti avoimempaa ja vapaampaa kuin muu yhteisöllisyys. Sosiaalinen status ei merkitse yleensä verkossa mitään, mielipiteet ja yhteiset kiinnostuksen kohteet yhdistävät ihmisiä. Tässä mielessä verkkoidentiteetti voi erota "arki-identiteetistä" erittäin paljon. Wellmanin artikkelissa mainitaan, että tasa-arvoa lisäävä tekijä on se, että ulkonäkö ei vaikuta netissä tapahtuvassa vuorovaikutuksessa. Tämä on asia, mikä on mielestäni muuttunut netti-identiteetissä. Esimerkiksi IRC-gallerian profiilissa vaaditaan käyttäjältä omakuva, josta käyttäjä on mahdollista tunnistaa, ja huomio profiilia katsottaessa kiinnittyy keskeisellä paikalla olevaan omakuvaan. MySpacessa kuvan ei tarvitse olla käyttäjältä itsestään, mutta useissa henkilökohtaisissa profiileissa omakuva on kuitenkin lisättyä. Tässä mielessä voidaan väittää, että ulkonäkö kuuluu myös yksilön identiteettiin näissä verkkoyhteisöissä.

Wellman (1997) mainitsee myös verkkoyhteisöissä vallitsevan sukupuolijakauman vääristymisen. Wellmanin esittelemien vuodelta 1995 olevien tutkimustulosten mukaan vain yksi viidesosa sen aikaisten palveluiden käyttäjistä

oli naisia. Tilanne on muuttunut vahvasti tasa-arvoisemmaksi, ainakin tutkielmassani tarkasteltujen palveluiden osalta. IRC-galleriassa naisia on jopa hieman enemmän kuin miehiä (IRC-galleria, 2007).

7. Yhteisöjen muodostuminen

Tarkastelluissa palveluissa yhteisöt muodostuivat hyvin erilaisten asioiden ympärille. MySpacessa yhteisöille ei ole yhteisiä nimittäjiä, eikä yhteisöjä ole erikseen muodostettu tai nimetty, vaan verkottuminen tapahtuu pelkästään lisättyjen ”kaverien” kautta. Jonkinlaisena yhteisönä voidaan MySpacessa nähdä vaikkapa tietyn yhtyeen muodostama *ystäväverkosto*, jonka kautta samanlaisesta musiikista pitävät ihmiset voivat löytää toisiaan ja muodostaa omia MySpace-ystävyyssuhteitaan.

Mielenkiintoista olisi tutkia näiden verkostojen muodostumista lähemminkin. Lisäksi mielenkiintoinen tutkimusaihe olisi ”tosielämän ystävyysuhteiden” ja verkkoystävien erotteleminen ja toisaalta näiden limittyminen. Tällaisessa tutkimuksessa voitaisiin tarkastella vaikkapa yksittäisten käyttäjien alun perin verkossa tapaamien ystävien tapaamista kasvokkain ja toisaalta ”tosielämän ystävien” lisäämistä myös WWW-ystäväverkostoon. Pitkällä aikavälillä voitaisiin saada mielenkiintoisia tutkimustuloksia näiden ystävyysuhteiden kestävyydestä!

8. Yhteenveto

Yhdistäviä tekijöitä tutkielmassani esitetyille palveluille on niiden yhteisöllinen luonne ja erilaisten verkostojen luominen. Tilausta tällaisille palveluille on, ja todennäköisesti tulee aina olemaan, koska sosiaalinen toiminta ja kontaktien luominen kuuluu inhimilliseen luonteeseen myös verkossa. Verkon luonteen muuttuessa ja kehittyessä yhteisölliset palvelut tosin voivat muuttua hyvinkin erilaisiksi.

Voidaan kuitenkin sanoa, että yhteisölliset WWW-palvelut tulevat olemaan erittäin suuri vaikuttaja niin sosiaalisesti kuin taloudellisestikin. Käyttäjämäärät ovat jatkuvassa kasvussa, ja palvelujen merkityksen voidaan sanoa kasvavan myös yksittäisten käyttäjien sosiaalisessa elämässä. Yhteisölliset palvelut ovat muuttuneet marginaalisten ryhmien harrastuksesta valtaviihteeksi.

Viiteluettelo

BBC. (5.9.2006). MySpace set to sell music online. Retrieved 19.2.2006. <http://news.bbc.co.uk/2/hi/entertainment/5316000.stm>.

Holahan, Catherine (2006). Did MySpace Really Beat Yahoo?. *Business Week Online*. 12, 15 (December).

- Honan, Mathew, (2006). Network to New Music. *Macworld*. 23, 11 (November), p74, 2p, 3c. http://www.businessweek.com/technology/content/dec2006/tc20061215_471367.htm?site=cbs&campaign_id=cbs.
- IRC-galleria (2007). IRC-gallerian tilastot, tammikuu 2007. Tarkistettu 30.1.2007. <http://irc-galleria.net>.
- Aki Järvinen, Satu Heliö & Frans Mäyrä (2002). *Communication and Community in Digital Entertainment Services*, Preresearch study , Hypermedia Laboratory Net Series, Tampere, 2002, Retrieved 20.2.2007. <http://tampub.uta.fi/tup/951-44-5432-4.pdf>.
- Karvonen, Tuomas, (2006). IRC-Galleria koetteli TNS Gallupin mittaussjärjestelmää (21.9.2006), *Taloussanommat, ITviikko*. Tarkistettu 19.2.2007. http://www.it-viikko.fi/page.php?page_id=15&news_id=200612634.
- Lapinski, Trent (2006). MySpace: The business of spam 2.0. Retrieved 17.2.2007 from <http://www.valleywag.com/tech/myspace/myspace-the-business-of-spam-20-exhaustive-edition-199924.php>.
- Mäyrä, Frans (2002). *Yhteisö*, Mediumi 1.1., 3.4.2002.
- O'Reilly, Tim, (2005). *What is Web 2.0 - Design Patterns and Business Models for the Next Generation of Software*, (30.9.2005) Retrieved 20.2.2007 from <http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>.
- Sean Rapacki (2007). *Places like MySpace*, Young Adult Library Services, Winter 2007.
- Sellers, P., (2006). MySpace cowboys. *Fortune*. (29.8.2006) Retrieved 19.2.2007 from http://money.cnn.com/magazines/fortune/fortune_archive/2006/09/04/8384727/index.htm.
- Wellman, Barry & Milena Gulia (1999). Virtual communities as communities. Net surfers don't ride alone. In Marc A. Smith & Peter Kollock *Communities in Cyberspace*, 167-194. London & New York: Routledge. Retrieved 18.2.2002 from <http://www.chass.utoronto.ca/~wellman/publications/netsurfers/netsurfers.pdf>.
- Williams, Alex, (2005). MySpace: A site with 26 million 'friends'. *New York Times* Thu 1, (Sep 2005). Retrieved 19.2.2007 from <http://www.iht.com/articles/2005/08/31/business/teensite.php>.

Roskapostin torjunta

Harri Pirttinen

Tiivistelmä.

Tämä tutkielma käsittelee erilaisia lähestymistapoja roskapostin torjuntaan. Torjuntakeinoja tutkitaan suodattamisen, toimintatapojen ja protokollien sekä lainsäädännön kannalta. Tutkielma käsittelee torjuntaratkaisuja yleisellä tasolla eikä siten sisällä esimerkiksi torjuntaohjeita kotikäyttäjille.

Avainsanat ja -sanonnat: roskaposti, spam, suodatus.

CR-luokat: H.4.3

1. Johdanto

Internet ja sähköposti ovat mahdollistaneet nopean, helpon tiedonsiirron ja kommunikoinnin, jota nykyajan yhteiskunnat hyödyntävät monipuolisesti. Sähköpostin avulla kuka tahansa voi tavoittaa maailmanlaajuisesti suuren määrän ihmisiä lähettämällä vain yhden viestin. Oikeellisen postin vastaanottamisen lisäksi sähköpostin käyttäjät ovat haluumattaan tutustuneet ilmiöön nimeltä roskaposti.

Roskaposti on todellinen, kansainvälinen ongelma, joka varaa tietoliikenteen resursseja, aiheuttaa ihmisille ja organisaatioille harmia ja taloudellisia menetyksiä. Tällä hetkellä sähköpostiliikenteestä suurin osa on roskapostia.

Ongelman vakavuus on ymmärretty ja ongelman taltuttamiseen kehitetään jatkuvasti vastatoimia useilla rintamilla. Tutkijat ovat huomanneet kolme keskeistä rintamaa, joilla sotaa roskapostia vastaan voidaan käydä. Nämä kolme rintamaa ovat roskapostin suodattaminen, protokollien uudelleenmäärittäminen sekä lainsäädäntö.

2. Mitä roskaposti on

Spamhaus [2007] käyttää roskapostille englanninkielistä määritelmää UBE, *Unsolicited Bulk Email*. MOT-sanakirjan [2007] mukaan *unsolicited* tarkoittaa pyytämätöntä ja *bulk* suurta kokoa, määrää tai massaa, eli roskapostiksi määritellään sähköposti, joka on saatu pyytämättä, sekä sitä on lähetetty suurin määrin. Durginin ja Sheriffin [2006] mukaan roskapostin määrittely on kuitenkin hankalaa, koska toiset pitävät edellä mainitun määritelmän *bulk*-sanana merkitystä turhana ja pitävät ainoana tärkeänä seikkana pyytämättömyyttä.

Myös roskapostin sisällöllä on ollut merkitystä määrittelyssä. CAUCE [2007] käyttää roskapostista määritelmää UCE, *Unsolicited Commercial Email*, jo-

ka tarkentaa pyytämättömän sähköpostin sisällön kaupalliseksi. Kaupallinen roskaposti on CAUCEn [2007] mukaan sisällöltään:

- ketjukirje
- pyramidimarkkinointiin liittyvää huijausmarkkinointia
- puhelinseksiin ja pornosivustoihin liittyvää mainontaa
- ohjelmistotarjouksia, joiden avulla hankitaan lisää sähköpostiosoitteita, joihin voi lähettää lisää kaupallista roskapostia
- tarjouksia palveluista, joiden avulla kaupallista roskapostia voi lähettää suuria määriä
- osaketarjouksia tuntemattomista uusista yrityksistä
- tarjouksia terveysvaikutteisista tuotteista, joiden vaikutuksella ei ole mitään perää
- tarjouksia laittomista ohjelmistokopioista eli piraattiohjelmista.

Sähköinen markkinointi on kaupallisen liiketoiminnan kannalta halpa ja tehokas tapa tiedottaa omista palveluista ja tuotteista. Ongelmana on, että roskaposti on luonut ihmisille negatiivisen kuvan myös täysin laillisesta sähköpostimarkkinoinnista [Li, 2006].

Roskapostia käytetään yleisesti laajoihin huijausyriityksiin. Luultavasti tunnetuimmat sähköpostihuijaukset liittyvät ns. Nigerian huijauskirjeisiin tai toisin tunnettuna 419-kirjeisiin [Lvm, 2005]. Näillä kirjeillä oli tarkoitus saada vastaanottaja auttamaan rahallisesti lähettäjää sillä houkutteella, että vastaanottaja saa myöhemmin avustuksesta suuren korvauksen. Myös ns. phishing-hyökkäyksissä hyödynnetään sähköpostia esiintymällä esim. vastaanottajan käyttämän pankin edustajana ja pyytämällä vastaanottajan tunnuksia tämän internetpankin käyttöön.

3. Koneellinen torjunta

Koneellinen torjunta on tietokoneella tehtävää roskapostien suodatusta tai sen lähettämisen estämistä. Jotta roskapostisuodatin voi suodattaa roskan, sen tulee tunnistaa tämä jollakin tekniikalla roskapostiksi oikeellisten postien seasta. Tunnistamisen lisäksi suodatin voi poistaa roskapostin vastaanottajan postilaatikosta heti sen saavuttua sähköpostipalvelimelle tai merkitä sen jollakin tavalla roskapostiksi.

Suodatinratkaisun tehokkuutta voidaan mitata sen mukaan, kuinka tarkasti suodatin tunnistaa tulevat roskapostit. Tehokkuudella ei ole kuitenkaan mitään arvoa, jos suodatin tunnistaa oikeelliset viestit roskaksi ja näin osoittautuu käyttäjän kannalta hyödyttömäksi. Suodattimien kyky parantaa käyttäjän tuot-

tavuutta rajoittuu siihen pisteeseen, että käyttäjien täytyy manuaalisesti katsoa suodatetut viestit, ettei niiden joukossa ole virheellisesti luokiteltuja sähköposteja (false positive) [Carpinter and Hunt, 2006].

Carpinter ja Hunt [2006] ovat tunnustaneet suodattimien ominaisuuksia, joiden mukaan niitä voidaan luokitella. Suodatinratkaisut jakautuvat sen mukaan, käytetäänkö suodattimessa koneen opettamista (machine learning). Tämän lisäksi koneen opettamiseen perustuvat suodattimet voidaan vielä jakaa täydellisiin sekä osittaisiin ratkaisuihin. Täydellisellä ratkaisulla tarkoitetaan suodatusratkaisua, joka pystyy itsenäisesti, oman tietovaraston perusteella tunnistamaan roskapostin. Osittaiset ratkaisut toimivat osana laajempaa suodatusjärjestelmää täydentäen tai palvelen suodatuksen pääratkaisua.

Suodatin sijaitsee joko sähköpostipalvelimella tai asiakaskoneessa, eli käyttäjän järjestelmässä. Sähköpostipalvelimella tapahtuvan suodatuksen etuna on sen keskitetty vaikutus kaikille palvelimen käyttäjille. Palvelimelle tulevasta tietovirrasta voidaan suodattaa roskaa kaikkia palvelimen käyttäjiä koskevilla säännöillä.

Käyttäjän järjestelmässä sijaitseva suodatin on yleensä täydellinen suodatinratkaisu. Käyttäjän on ylläpidettävä suodatinta opettamalla sitä tunnistamaan käyttäjän luokittelema roskaposti. Etuna ratkaisulla on se, että käyttäjä voi itse määrittää sääntöjä postin lajitteluun. Käyttäjän koneella sijaitseva suodattimen tarkkuus riippuu käyttäjän ylläpidosta ja opetusmateriaalista, jolla suodatin oppii erottamaan sähköposteja.

Sähköpostipalvelimilla tai asiakaskoneilla tapahtuva suodatus ei vähennä liikkuvan roskapostin määrää verkosta, vaan estää roskapostia päätyvästä käyttäjille, jotka eivät niitä halua. Suodattimilla voidaan säästää ihmisten aikaa, jonka he käyttävät sähköpostien lukemiseen.

3.1. Koneen opettamisesta poikkeavat suodatustavat

3.1.1. Heuristiikka

Koneen on vaikea erottaa roskapostia oikeellisesta postista, mutta ihminen havaitsee eron nopeasti viestin sisällön perusteella. Roskapostin piirteet onkin kuvattava suodattimelle siten, että mahdollisimman moni roskaposti täyttää piirteet ilman oikeellisten postien leimaamista roskaksi.

Heuristiikkaan perustuvat suodattimet erottavat roskapostit oikeellisista posteista etsimällä lauseita, sanoja tai piirteitä, jotka ovat ominaisia roskapostille, mutta eivät oikeelliselle. Etsittäviä piirteitä voidaan myös kutsua säännöiksi, joita suodatin käyttää. Kyseiset säännöt käsittelevät koko sähköpostin datasisältöä. Anttila [2004] erottelee datasisällön analyysit lähetyskenttien analyysiin, muoto- sekä sisältöanalyysiin.

Lähetyskenttien analyysi perustuu SMTP-viestin header-osuuteen. Header-osuuden X-kentät ovat käyttäjän määrittelemiä kenttiä, joiden avulla mm. välitetään tietoa sähköposteja käsitteleville ohjelmille postin käsittelemisestä. Vertailemalla X-kenttien sisältöä Header-osuuden pakollisten kenttien tietoihin voidaan päätellä, onko lähettäjä-kenttä väärennetty. Jos lähettäjä-kenttä on väärennetty, kyseessä on roskaposti, joka voidaan suodattaa.

Muotoanalyysi yrittää poistaa viestistä tekijät, jotka verhoavat viestin roskapostiksi luokittelevat piirteet. Piirteitä voi verhota esim. käyttämällä HTML-koodia. HTML-koodin pystyy kuitenkin poistamaan viestistä helposti ohjelmallisesti, jolloin verhous poistuu. Muotoanalyysiin kuuluu myös vastaanottajan ja lähettäjän välinen viestintäsuhde. Jos lähettäjä on tuntematon ja lähettää tiheästi viestejä, kyseessä on todennäköisesti roskapostittaja. Samaa tunnistustapaa voi käyttää minkä tahansa viestintäsuhteen nopealle muutokselle.

Sisältöanalyysi, nimensä mukaan, käsittelee viestin sisältöä. Vain itse postin vastaanottaja voi päätellä, onko posti hänelle roskaa vai ei. Käyttäjä voi oman kiinnostuksen ja kokemuksen mukaan luoda sanaston, jossa olevat elementit eivät yleisesti ilmene hänen oikeellisissa viesteissään. Jos viestin sisällöstä löytyy sanaston elementtejä, viesti on todennäköisesti roskapostia.

Carpinter ja Hunt [2006] kuvailevat yksinkertaisen heuristiikkaa käyttävän suodatinjärjestelmän. Suodatin luokittelee saapuvalle viestille roskapostipisteytyksen, jonka se saa sen mukaan, kuinka monta asetettua roskapostisääntöä se täyttää. Suodattimelle voi tällöin asettaa pisterajan, jonka ylittävät viestit luokitellaan roskapostiksi.

3.1.2. Tunnustekniikat

Tunnustekniikoissa (signature-based techniques) roskapostit tunnistetaan luomalla niistä yksilöivät hajautusarvot. Tämä tarkoittaa, että suodattimen tulee verrata jokaista vastaanotetun viestin hajautusarvoa kaikkiin tunnettuihin roskapostien hajautusarvoihin. Carpinterin ja Huntin [2006] mukaan yksilöivän tunnuksen luomiseen käytetyt tekniikat tekevät epätodennäköiseksi sen, että oikeellinen viesti voisi saada saman hajautusarvon kuin toinen eroava viesti, vaikka teoriassa jokaista hajautusarvoa vastaa ääretön määrä bittikokonaisuuksia. Koska roskapostit voidaan yksilöidä, tekniikka mahdollistaa erittäin vähäisen väärin luokittelujen määrän.

Tekniikan avulla voidaan siis tunnistaa jo kerralleen tunnistettuja roskaposteja uudelleen, eli ne läpäisevät suodattimen ensimmäisen kerran. Lisäksi, viestin hajautusarvo muuttuu sisällön muuttuessa, vaikka muutos olisikin vain hyvin pieni, esim. yksi merkki. Ongelma on siis, että yksinkertaisen tunnustekniikkaan perustuvan suodattimen voi ohittaa lisäämällä roskapostiin mieli-

valtaisen merkkijonon. Tämä on johtanut hajautustekniikoiden kehittämiseen, joiden avulla voidaan tunnistaa pienet muutokset roskaposteista.

Toinen ongelma tekniikassa on hajautusarvojen arkistointi. Hajautusarvo-tietokannan paisuessa suuremmaksi suodatus käy raskaammaksi lisääntyneen vertailumäärän takia. Vanhoja hajautusarvoja voisi poistaa, mutta se mahdollistaisi vanhojen roskapostien uusiokäytön.

Carpinter ja Hunt [2006] esittelevät kaksi roskapostintunnistustapaa, jotka soveltavat tunnustekniikkaa tavalla, joka tunnistaa myös pienet muutokset roskapostien välillä. Ensimmäinen tapa on käyttää hajautuksen ja viestin sisällön tiheyttä. Sovelluksessa viestin sisällön joukosta otetaan osajoukko merkkijonoja, joista jokaisesta luodaan oma hajautusarvo. Hajautusarvot muodostavat viestin vektorikuvauksen, jonka avulla samankaltaiset viestit voidaan tunnistaa ja tallentaa niiden toistuvuus. Tunnistustapa on erittäin tehokas, kun siihen vielä lisätään listaus oikeellisista massapostittajista.

Toinen mainittu sovellustapa on laskea viestistä viestitiiviste (message digest), sekä kerätä lähetysalkuperän sekä viestin sisältämät URL-viitteet. Jokaisesta viestistä muodostetaan 256-bittinen tiiviste. Viestit tulkitaan samoiksi, jos tiiviste eroaa enintään 74 bitillä.

3.1.3. Listat

Listat ovat yksinkertainen tapa tunnistaa potentiaalinen tai varmistettu roskapostittaja luotetuista lähettäjästä. Listat ovat yleisin osaratkaisu nykyajan suodatussovelluksissa. Listoja pidetään yleisesti kahdenlaisista lähettäjästä, luotetuista, sekä roskapostittajista.

Varmistettuja roskapostittajia listataan ns. mustille listoille (blacklists), joita kutsutaan myös sulkulistoiksi. Suodatuksessa mustalla listalla olevalta lähettäjästä tulevat sähköpostit suodatetaan automaattisesti pois. Luotettavista lähteistä pidettäviä listoja kutsutaan valkoisiksi listoiksi (whitelists). Valkoiset listat toimivat suodatuksessa päinvastoin kuin mustat listat, eli valkoisella listalla olevalta lähettäjästä tulevat sähköpostit hyväksytään automaattisesti.

Koska listojen pitäminen on yksinkertaista ja helppoa, se voidaan toteuttaa sähköpostipalvelimella, kuten myös käyttäjän järjestelmässä.

Tekniikan yksinkertaisuus mahdollistaa myös helpon suodatuksen kiertämisen, koska sähköpostin lähettäjä-kentän voi väärentää. Roskapostittaja voi väärentää lähettäjä-kentän jokaiseen roskapostiin sellaiseksi, jota todennäköisesti ei löydy mustalta listalta. Roskapostittaja voi myös hyödyntää valkoista listaa esittämällä lähettäjä, joka todennäköisesti löytyy vastaanottajan valkoiselta listalta.

Listaukseen sovelletaan myös DNS-tietoihin (Domain Name System) pitämällä laajaa tietokantaa palvelimista ja verkoista, joista sallitaan roskapostin lähettä-

minen. Lähetystietoja voidaan verrata SMTP-istunnon alkaessa, todeta mustalla listalla oleva verkko-osoite ja sulkea istunto. Verkko-osoitteen vertaaminen on tehokas tapa tunnistaa roskaposti, sillä se ei vaadi palvelimelta paljoa resursseja, ja se suodattaa suuren osan roskapostista. Xie ja muut [2006] mainitsevat myös valkoisen DNS-listan, MARIDin (Mail transfer agent Authorization Records In DNS), joka pakottaa lähettäjän tunnistautumaan. MARIDin toteutus voi olla hajautettu valkoinen lista tunnistetuista, oikeellisista sähköpostinlähettäjäistä.

Carpinter ja Hunt [2006] huomioivat, että DNS-listaus on vaarallinen ainoana kokonaisena suodatinratkaisuna, koska se mahdollistaa suuren määrän virheellisesti luokiteltuja sähköposteja. Roskapostittajat voivat nopeasti hankkia uuden lähetysosoitteen, jos entinen osoite joutui mustalle listalle. Uuden osoitteen päätyminen mustalle listalle voi vaatia useita valituksia roskapostin vastaanottajailta, jolloin roskapostittaja kerkeää toimimaan mahdollisesti pitkään. Roskapostittajat voivat myös naamioitua oikeelliseksi palveluksi, esim. luotetuksi yritykseksi. Tällöin he voivat välttää mustat listat, tai mahdollisesti saada tunnettu laillinen palvelu mustalle listalle, jolloin lista periaatteessa menettää eheydensä. Tämä voi myös vaikuttaa negatiivisesti ylläpitäjien haluun käyttää DNS-listausta.

3.1.4. Tietoliikenneanalyysi

Carpinter ja Hunt [2006] mainitsevat luokittelutavan, jolla tietoliikenteestä voidaan tunnistaa roskapostiliikenne. Tutkimalla mm. seuraavia sähköpostien ominaisuuksia, pystytään erottamaan roskapostiliikenne muusta liikenteestä:

- sähköpostin saapumisprosessi
- sähköpostin koko
- vastaanottajien määrä yhtä sähköpostia kohtaan
- sähköpostin suosio
- vastaanottajien väliaikainen sijainti.

Erot oikeellisen ja roskapostin liikenteessä näkyvät niiden käyttötarkoituksessa: oikeellinen sähköposti on sosiaalista toimintaa, missä lähettäjä luo viestin tarkoituksen mukaisesti, kun taas roskapostia luodaan automaattisesti, ja sitä yritetään levittää mahdollisimman monelle vastaanottajalle.

3.2. Koneoppiminen

3.2.1. Bayesilainen suodatus

Bayesilainen suodatus on saanut tukevan jalansijan nykyajan roskapostin torjunnassa. Yksikään koneen opettamiseen tai tilastolliseen analyysiin perustuva

tekniikka ei ole saavuttanut vastaavaa laajaa suosiota. Se edustaa uusinta, parasta lähestymistapaa suodatusalalla [Carpinter and Hunt, 2006].

Bayesilainen suodatus on suosittu suodatustekniikka, koska siinä ei ole selkeitä puutteita, kuten esim. heuristisessa suodatuksessa. Suodatustekniikan säännöt perustuvat käyttäjien tekemiin luokituksiin, jolloin käyttäjät määrittävät, mitkä viestit ovat heille roskapostia. Tästä seuraa, että käyttäjien luokitukset eroavat, jolloin heidän suodattimien säännöt ovat myös erilaisia. Bayesilaisen suodattimen sääntöjoukko on tällöin tuntematon roskapostittajalle, eikä se voi kohdistaa tietynlaisia, säännöt läpäiseviä, roskaposteja vastaanottajalle. Tämän lisäksi suodatinta opetetaan vastaanotetuilla, suodattimen läpäisseillä roskaposteilla, jolloin suodatin pysyy ajan tasalla roskapostitrendeistä.

Bayesilainen suodatin hyödyntää kahta taulua, joista toinen sisältää oikeellisten viestien sisällön sanoina, ja toinen vastaavasti roskapostien sanat. Tauluihin tallennetaan jokaiselle sanalle todennäköisyysarvo, joka kertoo kuinka todennäköisesti se ilmestyy oikeellisessä viestissä tai roskapostissa. Todennäköisyys voidaan laskea esim. oikeellisten viestien tietokannassa olevalle sanalle jakamalla sen ilmestymien lukumäärä opettamiseen käytettyjen oikeellisten viestien lukumäärällä. Tietokannat luodaan automaattisesti käyttäjän tekemien sähköpostien luokittelujen mukaan, lukemalla viestin tekstiesitys sekä jakamalla se sanoihin.

Roskapostin tunnistaminen perustuu viestissä olevien sanojen todennäköisyysarvoihin. Ns. naiivi toteutustapa on valita n-kappaletta sanoja, joilla on suurimmat todennäköisyysarvot ilmestyä joko oikeellisessä sähköpostissa tai roskapostissa. Tämä toteutustapa on tosin heikko, sillä roskapostittajat tietoisesti lisäävät viestin sisältöön sanoja, jotka todennäköisesti toistuvat useammin oikeellisissa sähköposteissa. Lopullinen todennäköisyys sähköpostille saadaan käyttämällä valittujen sanojen todennäköisyyksiä Bayesin kaavassa. Kuva 1 antaa esimerkin Bayesin kaavan käytöstä.

$$P = \frac{x_1 \cdot x_2 \cdot x_3 \cdot K \cdot x_n}{x_1 \cdot x_2 \cdot x_3 \cdot K \cdot x_n + (1 - x_1) \cdot (1 - x_2) \cdot (1 - x_3) \cdot K \cdot (1 - x_n)}$$

Esimerkiksi : sana₁ = greetings, sana₂ = offer, sana₃ = special, sana₄ = order

Osuudet roskaposteista, joissa sanat ilmestyivät : x₁ = 0.3, x₂ = 0.68, x₃ = 0.87, x₄ = 0.66

Sähköposti on roskapostia todennäköisyydellä :

$$P = \frac{0.3 \cdot 0.68 \cdot 0.87 \cdot 0.66}{0.3 \cdot 0.68 \cdot 0.87 \cdot 0.66 + (1 - 0.3) \cdot (1 - 0.68) \cdot (1 - 0.87) \cdot (1 - 0.66)} = \frac{0.1171368}{0.1270376} = 0.9220640$$

$P \approx 0.92$

Kuva 1. Esimerkki Bayesin kaavan käytöstä [Carpinter and Hunt, 2006]

3.2.2. SMTP Retry –suodatus

Roskapostittajat käyttävät yleisesti työkaluja, jotka ovat suunniteltu varta vasten roskapostittamiseen. Työkalujen avulla he lähettävät roskapostia massiivisille roskapostituslistoille. Työkalut eivät kuitenkaan yritä lähettää viestiä listalla olevaan osoitteeseen useasti, jos sen vastaanottamisessa tapahtuu virheitä. Toisin sanoen, roskapostittajat eivät yleisesti yritä lähettää viestiä uudelleen [Carpinter and Hunt, 2006].

Listausta hyödyntämällä voidaan uudelleenyritystä käyttää roskapostin tunnistamiseen. Jokainen lähettäjä, joka ei yritä uudelleenlähetystä SMTP-istunnon epäonnistumisen jälkeen, listataan mahdollisten roskapostittajien listalle. Jos lähettäjä yrittää uudelleenlähetystä, se voidaan lisätä valkoiselle listalle. Jos sama lähettäjä yrittää lähettää useammin, mutta ei tee uudelleenlähetyksiä, lähettäjä päätyy mustalle listalle.

4. Ratkaisu toimintatavan muutoksella

Yksi tapa vastata roskapostiongelmaan on puuttua protokoliin ja toimintatapoihin jotka tukevat tai mahdollistavat roskapostituksen nykymuodossaan. Muutoksia on esitelty useita ja monet ovat varteenotettavia vaihtoehtoja, mutta kuitenkin niiden toteuttaminen ja käyttöönotto vaatisi kansainvälistä hyväksyntää ja sen aikaansaaminen on erittäin vaikeaa. Carpinter ja Hunt [2006] toteavat, että useita lähes valmiita ratkaisuja on tuotettu, mutta ne eivät saa tukea, koska ne vaatisivat erittäin laajaa sähköpostiprotokollan päivittämistä tai vaihtamista. Tässä luvussa esitellään muutama ehdotettu toimintatavan muutos, joilla voisi torjua roskapostitusta.

4.1. Sähköposti maksulliseksi

Roskapostin taloudellinen etu perustuu sähköpostin edullisuuteen, sillä käyttäjälle sähköposti sinänsä on maksutonta, yleensä vain internet-yhteys on maksullinen. Roskapostin taloudellista hyötyä voitaisiin vähentää lisäämällä sähköpostin lähettämiseen pieni maksu. Jokaisesta lähetetystä sähköpostista voidaan kerätä mikropostimaksu, jolloin viestin lähettäminen maksaisi esimerkiksi sentin murto-osan [Lvm, 2005]. Tällöin hinta ei kasva normaalin sähköpostin käytön kautta suureksi, mutta massapostituksen kautta postimaksut kasvaisivat tuntuvaksi kuluksi esim. maksun ollessa 0,002€, miljoona viestiä maksaisi 2000€. Maksujärjestelmän kansainvälinen toteutus voisi muodostua esteeksi ja Liikenne- ja viestintäministeriön [Lvm, 2005] mukaan maksuja voitaisiin aina kiertää.

Asiaa on pohdittu myös toisesta näkökulmasta, jossa postimaksu maksettaisiin korvauksena vastaanottajalle. Tässä mallissa lähettäjä voisi itse asettaa ta-

kuumaksun lähettämälleen viestille. Takuumaksun summalla voisi osoittaa, kuinka varma lähettäjä on siitä, että hänen viestinsä ei ole vastaanottajalle roskapostia, esim. jos takuusummaksi on asetettu 5€, niin todennäköisesti viesti on oikeellinen. Vastaanottaja voisi lunastaa takuumaksun, jos vastaanotettu viesti on vastaanottajalle roskapostia. Samalla lähettäjä saisi tiedon, että vastaanottaja ei halua jatkossa sähköposteja kyseiseltä lähettäjältä. Tämä malli pakottaisi mainostajat keskittymään vastaanottajiin, jotka oikeasti haluavat tiedotteita; todennäköisesti vastaanottajat, jotka eivät sähköposteja halua, lunastaisivat takuumaksun.

4.2. Tehtävä-vastaus -malli

Tehtävä-vastaus -malli asettaa lähettäjälle tehtävän, jonka vain ihminen voi ratkaista. Tehtävä on ihmiselle yksinkertainen ja nopea, mutta luonteeltaan sellainen, ettei tietokoneohjelma pysty sitä algoritmilla ratkaisemaan. Tehtävä asetettaisiin lähettäjälle joka kerta, kun vastaanottaja ei tunne kyseistä lähettäjä. Tehtävä voidaan lähettää lähettäjän sähköpostiosoitteeseen, jolloin myös varmistutaan siitä, että lähettäjä ei ole väärentänyt lähetysooitettaan. Tehtävän ratkaiseminen antaa tiedon vastaanottajalle, että lähettäjä on ihminen ja jos hänen lähettämä viestinsä on roskapostia, lähettäjän voi suoraan lisätä mustalle listalle [Lvm, 2005].

Yksi mallin ongelma on, että osa automaattisesti lähetettävistä viesteistä on tietoisesti tilattuja sähköposteja, esimerkiksi erilaiset uutisryhmäkirjeet. Lisäksi malli hidastaa sähköpostiliikennettä, minkä vuoksi monet eivät käytä mallia [Carpinter and Hunt, 2006].

4.3. Aikavaatimuksen korottaminen

Sähköpostien lähettäminen on nopeata toimintaa ja rajoittuu lähinnä kaistan laajuuden ja sähköpostipalvelimien nopeuden mukaan. Yksi tapa rajoittaa massalähettämistä olisi lisätä lähettämisen yhteyteen laskennallinen tehtävä, joka vaatisi lähettäjän tietokoneelta laskenta-aikaa.

Laskennallinen tehtävä olisi luonteeltaan sellainen, että se ei hidastaisi huomattavasti yhden tai muutaman sähköpostin lähettämistä, mutta määrän kasvaessa ajantarve kasvaisi huomattavasti. Massapostituksen yhteydessä laskennan ajantarpeen tulisi olla niin suuri, että viestien lähettäminen veisi tolkutoman paljon aikaa ja tehokkaasti hidastaisi suuren sähköpostimäärän lähettämistä. Esimerkiksi Microsoft on ehdottanut, että jokaisen kaupallisen sähköpostin lähettämiseen tulisi kulua n. 10 sekuntia [OECD, 2006].

Liikenne- ja viestintäministeriö [Lvm, 2005] huomioi, että malliin liittyy haittapuolia. Ensinnäkin, laskenta-aika koskee erikseen jokaista lähettäjä lähetysmäärän mukaan, jolloin hajautettu postitus kiertää täysin mallin torjuvan

vaikutuksen. Toiseksi, malli ei auta niitä, jotka jo saavat roskaposteja, eikä se estä roskapostitusta, jolloin investoinneista huolimatta, ne yhä rasittavat sähköistä kommunikointia taloudellisesti.

4.4. Luottamusverkostot

Liikenne- ja viestintäministeriö [Lvm, 2005] mainitsee luottamusverkot mahdollisena ratkaisuna roskapostiongelmiaan. Mallissa viranomaisena voisi myöntää sähköisen henkilöllisyyden lähettäville. Tämän henkilöllisyyden kautta lähettäjä voidaan aina tunnistaa ja tuomita lainvastainen käyttäytyminen. Ongelmaksi muodostuu vahva autentikointi, joka poistaa yksityisyyden oikeuteen kuuluvan mahdollisuuden toimia verkossa anonyymina.

Yksityisyyden säilyttäminen voidaan mahdollistaa luottamusverkostolla, jossa verkkoon liittyminen ei tarvitse viranomaista, vaan suosittelun verkon käyttäjältä. Tällöin verkkoon suositeltavat henkilöt kuuluvat aina jonkun käyttäjän sosiaaliseen verkostoon ja silloin luottamus säilyy verkon sisällä.

Luottamusverkosto sulki verkkoon kuulumattomat verkon ulkopuolelle, joka vaatisi laajaa jäsenkuntaa, jotta luottamusverkolla olisi arvoa käyttäjilleen. Koska sähköpostilla on verkostovaikutus, on eri yhteiskuntaryhmiä vaikea saada liittymään luottamusverkostoon tai verkosto voi mahdollisesti syrjiä joitain ryhmiä esim. pienituloisia tai heikosti koulutettuja [Lvm, 2005].

5. Torjunta lainsäädännöllä

Lainsäädännön tehtävänä on määrittää, mitä roskaposti on ja asettaa sen lähettämiseksi selkeät rajat, jotta roskapostittajat voidaan asettaa vastuuseen teoistaan. Lainsäädännön avulla roskapostittajia voidaan estää ja täten vähentää todellista liikkuvaa roskapostimäärää.

Roskapostittamiseen liittyy useita yleisesti rikoksiksi tuomittavia tekoja. Roskaposteissa voidaan tarkoituksellisesti huijata vastaanottajaa ja täten hyötyä vastaanottajasta taloudellisesti. Lisäksi roskapostittaja ei esiinny omana itsenään vaan antaa tietoisesti väärää tietoa itsestään. Ehkä edellä mainitusta rikoksesta vakavampi on tilanne, jossa roskapostittaja esittää olevansa joku toinen, eli varastaa toisen identiteetin.

5.1. Opt-in ja opt-out -toimintatavat

Maailmalla on yleisesti käytössä kaksi toimintatapaa, joilla sähköinen markkinointi mahdollistetaan ja samalla roskapostittaminen tuomitaan rikokseksi. Ensimmäinen toimintatapa on ns. opt-in, joka vaatii lähettäjä saamaan ensiksi luvan vastaanottajalta, että lähettäjä saa lähettää hänelle palveluun liittyviä kaupallisia tai muita tiedotteita. Lähettäjä ei siis saa pyytää lupaa sähköpostitse vaan vastaanottajan on itse oma-aloitteisesti annettava suostumuksensa, esi-

merkiksi palvelun verkkosivuilla. Opt-in toisin sanoen antaa ihmisille vapauden valita mitä postia he saavat.

Toinen yleisesti käytössä oleva toimintatapa on vastaavasti ns. opt-out. Opt-out mahdollistaa sähköpostimarkkinoinnin monipuolisesti ja tukee sitä liiketoimintana. Opt-out ei vaadi lähettäjä hankkimaan vastaanottajan lupaa etukäteen vaan antaa lähettäjälle luvan lähettää vapaasti markkinointiin liittyviä tiedotteita. Vastaanottajan on kuitenkin pystyttävä kieltämään viestien lähetys, jos vastaanottaja ei niitä halua. Opt-out -toimintatapa mahdollistaa kaikille tasarvoisen tiedotuksen palveluista, jolloin jokaisella on mahdollisuus valita kaiken tiedotuksen keskeltä ne, jotka kutakin kiinnostavat.

Li [2006] esittää tehokkaan keinon toteuttaa opt-out -toimintatavan, joka on eduksi lähettäjälle sekä vastaanottajalle. Opt-out voisi perustua yhteen lähettämiseen ja sen tulkintaan. Tällöin lähettäjä voi lähettää yhden viestin vastaanottajalle, joka ei tällä hetkellä kuulu tilaaja-vastaanottajiin. Jos vastaanottaja haluaa jatkossa viestejä lähettäjältä, hänen tarvitsee vain lähettää vahvistus lähettäjälle ja täten tilata tiedotteet. Tämä tapa eroaa perinteisestä siten, että vastaanottaja voi päättää, haluaako hän lisää tiedotteita ja lähettäjä saa varmuuden, että lähetetyt viestit menevät henkilöille, jotka ovat aidosti kiinnostuneita tiedotteista.

5.2. Kaupallisten tiedotteiden merkintätavat

Useissa Aasian valtioiden ja Yhdysvaltojen osavaltioiden lainsäädännöissä vaaditaan, että kaupallinen sähköposti tulee merkitä määritetyillä merkinnöillä (labelling). Mainonta ilman merkintöjä on tuomittava teko. Merkinnät ovat merkkijonoja, joiden tulee sijaita viestin otsikossa tai sähköpostin header-osioissa. Merkintöjen tarkoituksena on ilmoittaa vastaanottajalle, että kyseessä on kaupallinen tiedote. Tällöin vastaanottaja tietää aiheen ja voi itse päättää, lukeeko hän tiedotteen vai ei.

Li [2006] mainitsee, että esimerkiksi Kiinassa kaupallisen tiedotteen otsikoon tulee lisätä sana mainos kiinaksi tai "AD", joka viittaa englannin kielen sanaan "advertisement". Yhdysvalloissa ja Koreassa merkinnöissä on eroteltu lisämerkintöjä aikuisviihteelle ja muulle seksuaaliseen sisältöön liittyvälle mainonnalle. Merkinnöillä on tarkoitus suojella lapsia viesteiltä, jotka sisältävät aikuisille suunnattua materiaalia.

5.3. CAN-SPAM

Yhdysvallat on huomattava roskapostin alkuperämaa johtuen mm. eriävistä osavaltioiden laeista. CAN-SPAM-laki on Yhdysvaltojen yritys yhtenäistää osavaltioiden eriävät ja puutteelliset lakisäädökset. CAN-SPAM koskee lähinnä sähköposteja, jotka ovat ensisijaisesti luonteeltaan kaupallisia tiedotteita. Laki

toteuttaa Opt-out toimintatapaa ja määrittää, että viestissä olevan lähettäjän palauteosoitteen on oltava voimassa vähintään 30 päivää lähetyksestä. Lisäksi lähettäjän on lopetettava lähetykset kymmenen päivän sisällä vastaanottajan kieltomitoituksesta. Kaikki harhaanjohtava tietoliikenneinformaatio on kiellettyä, eli lähettäjätietojen on oltava oikeelliset. Sähköpostit on myös lähetettävä omalta palvelimelta, eli välityspalvelinten käyttö tai muu alkuperän peittäminen on kiellettyä. Sähköpostin aihe ei saa olla harhaanjohtava ja se on merkittävä sanalla "ADV." tai "ADVERTISEMENT" kun kyseessä on mainos. Mainos-merkinnän lisäksi otsikkoon on lisättävä "ADULT", jos se sisältää aikuisille suunnattua mainontaa.[Li, 2006]

CAN-SPAM tuli voimaan vuoden 2004 alussa, pelotellen suurilla sakoilla ja vankilatuomioilla. Kuitenkaan se ei ole vaikuttanut huomattavasti roskapostin määrän vähenemiseen. Laki on herättänyt pettymystä ja vastustusta asiantuntijoiden keskuudessa. Spamhaus-projektin johtaja Steve Linford [Linford, 2003] ilmaisi, että laki yrittää enemmänkin säädellä roskapostitusta kuin kieltää sitä ja siksi laki tulee lisäämään roskapostin ja roskapostittajien määrää. Hänen mielestään CAN-SPAM paremminkin laillistaa roskapostin lähettämisen lain antamalla ehdoilla.

Lee [2005] tarkentaa, että lain tarkoituksena on tuottaa ahdinko suurille roskapostittajille ja aiheuttaa taloudellisen hyödyn väheneminen tekemällä roskapostittamisesta kalliimpaa. Kalleus johtuu siitä, että roskapostittajat eivät pystyisi hyödyntämään entistä lähetyspotentiaalia, koska suuri osa siitä on uuden lain mukaan laitonta. Laki on kuitenkin niin tehokas kuin sen käytännön toteuttaminen oikeussaleissa.

Lee [2005] tarjoaa selityksiä, miksi laki ei ole vähentänyt roskapostin määrää. Roskapostittajat pystyvät helposti vaihtamaan maantieteellistä sijaintiaan alueelle, jossa ei ole roskapostittamisen kieltävää lainsäädäntöä. Roskapostittajat voivat myös uuden lain mukaan käyttää laillisia sähköpostitilejä laajaan postittamiseen. Roskapostittajilla on vieläkin käytössä zombie-verkot, eli kaapattujen koneiden verkkoja, joiden avulla he voivat levittää roskaa jäämättä itse kiinni. Osa internet-palveluntarjoajista on voinut myydä asiakkaidensa sähköpostiosoitteet roskapostittajille miljoonilla dollareilla, jolloin he ovat hyötynet roskapostittajista ja täten nämä palveluntarjoajat voivat olla vastahakoisia käyttämään lain tuomia mahdollisuuksia.

5.4. Kansainvälinen yhteistyö

Jokaisen maan oma laki koskee maassa asuvia roskapostittajia. Lakien ongelmana onkin, että niiden avulla on vaikea puuttua ulkomailta tuleviin roskapostivirtoihin. Lin [2006] mukaan eri maiden sisäiset lainsäädännöt ovat vastahakoisia puuttumaan kansainväliseen roskapostittamiseen. Ongelmaa lisäävät

valtiot, joissa ei vieläkään, ehkä tietoisesti, roskapostin kieltävää lainsäädäntöä tai resursseja lain valvomiseen.

Ongelmaan on etsitty ratkaisua kansainvälisistä organisaatioista, jotka pystyisivät ohjeistamaan sekä yhdistämään valtioita rajat ylittävään roskapostin torjuntaan. Euroopan unioni on osallistunut omalta osaltaan taisteluun ja on laatinut sähköisen tietosuojan direktiivin 2002/58/EY [EU, 2002]. Direktiivi sisältää suosituksia jäsenmaiden lainsäädäntöihin mm. sähköisen markkinoinnin hyväksyttävästä toteutustavasta. Direktiivi asettaa mm. seuraavat säännöt:

- "Automatisoitujen soittojärjestelmien sekä telekopiolaitteiden, sähköpostiviestien, tekstiviestien, puheviestien, ääniviestien tai kuvaviestien avulla toteutettua suoramarkkinointia saa kohdistaa vain sellaisiin luonnollisiin henkilöihin, jotka ovat antaneet siihen ennalta suostumuksensa." [EU, 2002, artikla 13.1]
- "Joka tapauksessa on kiellettävä sellaisen sähköpostin lähettäminen suoramarkkinointitarkoituksessa, jossa peitetään tai salataan sen lähettäjän henkilöllisyys, jonka puolesta viesti on lähetetty tai jossa ei ole voimassa olevaa osoitetta, johon vastaanottaja voi lähettää pyynnön siitä, että kyseinen viestintä lopetetaan." [EU, 2002, artikla 13.4]

Suomen sähköisen viestinnän tietosuojalaki (516/2004) toteuttaa direktiivin säädökset, eli Suomessa on voimassa lailla säädetty opt-in -toimintatapa. Roskaposti voi lisäksi rikkoa Suomessa henkilötietolakia, kuluttajansuojalakia sekä rikoslakia [Lvm, 2005].

OECD (Organization for Economic Cooperation and Development) on laatinut työkalupakin jäsenmailleen. Työkalupakki [OECD, 2006] sisältää suositeltuja vastatoimia ja toimintatapoja koskien roskapostia. Tarkemmin sanoen työkalupakki käsittelee viranomaisien yhteistyötä, teollisuuden ajamia toimia, teknisiä ratkaisuja koulutus- ja tiedotusaloitteita, roskapostin vastaisia toimintatapoja sekä kansainvälistä yhteistyötä ja tiedonvaihtoa.

Li [2006] mainitsee useita organisaatioita, jotka toimivat kansainvälisesti roskapostia vastaan. Roskapostin määrä on kasvussa, eikä hänen mielestä näiltä organisaatioilta voida odottaa ratkaisua ongelmaan millään määritetyllä aikavälillä.

6. Yhteenveto

Roskapostia vastaan soditaan useilla rintamilla käyttäen hyväksi laajaa torjunta-arsenaalia. Tästä huolimatta roskaposti täyttää voitokkaasti meidän postilaitteet kerta toisensa jälkeen. Tulevaisuus kuitenkin voi tuoda tilanteeseen rat-

kaisun, joka todennäköisesti perustuu laajaan kansainväliseen yhteistyöhön ja koko ajan jatkuvan tutkimustyön kehittämiin torjuntatapoihin. Voittoa roska-postista ei kuitenkaan saavuteta hetkessä, sillä toimiva ratkaisu vaatii yhteistä lainsäädäntöä, toimintatapamalleja porsaanreikiä vastaan sekä tuntuvia rangaistuksia ja lain käytännön toteuttamista.

Viiteluettelo

- [Anttila, 2004] Susanne Anttila, Roskaposti ja sen torjunta. Tampereen yliopisto, Tietojenkäsittelytieteen laitos, **B-2004-8**, 126-144. Saatavana <http://www.cs.uta.fi/reports/bsarja/B-2004-8.pdf> (7.2.2007).
- [Carpinter and Hunt, 2006] James Carpinter and Ray Hunt, Tightening the net: A review of current and next generation spam filtering tools. *Computers & Security* **25**, 8 (Nov. 2006), 566-578.
- [Durgin and Sheriff, 2006] Janet Durgin and Joseph S. Sheriff, Effects of unsolicited e-mail on the virtual business world. *Kybernetes* **35**, 5 (2006), 668-679.
- [EU, 2002] Euroopan parlamentti ja neuvosto, Sähköisen tietosuojan direktiivi. Saatavana <http://eurlex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2002:201:0037:0047:FI:PDF> (6.5.2007).
- [Lee, 2005] Younghwa Lee, The CAN-SPAM act: A silver bullet solution? *Communications of the ACM*, **48**, 6 (June 2005), 131-133.
- [Li, 2006] Xingan Li, E-marketing, unsolicited commercial e-mail, and legal solutions. *Webology* **3**, 1 (March, 2006), article 23. Available at <http://www.webology.ir/2006/v3n1/a23.html>.
- [Linford, 2003] Steve Linford, Spamhaus position on CAN-SPAM act of 2003. Available at http://www.spamhaus.org/position/CAN-SPAM_Act_2003.html (5.5.2007).
- [Lvm, 2005] Liikenne- ja viestintäministeriö, Roskapostipaketti. Saatavana www.roskapostipaketti.fi (25.4.2007).
- [Mot, 2007] Mot-sanakirja, versio 4.6. Saatavilla www.kielikone.fi (11.3.2007).
- [OECD, 2006] Report of the OECD task force on spam: Anti-spam toolkit of recommended policies and measures. 19.4.2006. Available at <http://www.oecd.org/dataoecd/63/28/36494147.pdf> (6.5.2007).
- [Spamhaus, 2007] Spamhaus-project, The definition of spam. Available at www.spamhaus.org/definition.html (11.3.2007).
- [Xie et al. , 2006] Mengjun Xie, Heng Yin and Haining Wang, An effective defense against email spam laundering. In: *Proceedings of the 13th ACM Conference on Computer and Communications Security*, 179-190.

Roskapostin suodatusmenetelmistä

Harri Sundström

Tiivistelmä.

Roskaposti on levinnyt räjähdysmäisesti viime vuosina ja aiheuttaa entistä enemmän ongelmia sähköpostin käyttäjille. Roskapostin estämiseksi on olemassa useita keinoja. Tässä tutkimuksessa on esitelty sisältöpohjaisten, roska-postin tunnistamiseen perustuvien suodatusmenetelmien periaatteita, mahdollisuuksia ja heikkouksia. Menetelmät on jaettu sisältöperustaiseen, tilastollisiin, heuristisiin ja tarkistenumero-perustaisiin suodattimiin. Tarkastelu osoittaa, että tämän hetkiset suodattimet eivät kykene ratkaisemaan roska-postiongelmia täydellisesti. Tulevaisuudessa tulee kehittää sähköpostijärjestelmä, jossa roska-postin lähetys on tehty mahdottomaksi.

Avainsanat ja -sanonnat: Roskaposti, roska-postin suodatus, sähköpostin suodatus, spam, sähköposti

CR-luokat: H.3.3, H.4.3.

1. Johdanto

Sähköpostin käyttö yrityksen sisäisessä ja ulkoisessa kommunikaatiossa on yleistynyt viimeisinä vuosikymmeninä. Lokakuussa 2006 sähköposti täytti 25 vuotta. Nykyisen kaltainen sähköpostijärjestelmä sai alkunsa kun SMTP:n (Simple Mail Transfer Protocol) kehitys alkoi. ARPANET (Advanced Research Projects Agency Network) oli ensimmäisiä verkkojärjestelmiä ja siinäkin halutuin sovellus oli juuri sähköposti. Sähköpostin ja internetin kehittäjien, alan pioneerien, pääasiallinen tarkoitus on ollut kehittää toimiva internet. Nyt useat käyttäjät haluavat vahingoittaa internetiä ja sähköpostijärjestelmiä. Alkuperäistä SMTP:tä suunniteltaessa ei kenellekään tullut mieleen, että joku lähettäisi sähköpostia tuntemattomalle ihmiselle [Robertson, 2006].

Mainoksien määrä on kasvanut niin perinteisessä kirjepostissa kuin sähköpostissakin. Perinteiseen postiin ilmoitus postiluukussa "ei mainoksia kiitos" saattaa vähentää mainosten määrää, mutta sähköpostijärjestelmään ei tällaista mahdollisuutta ole olemassa. Sähköpostin käyttäjän onkin vaikea, ellei mahdoton, estää ei toivottuja viestejä täyttämästä sähköpostiaan.

Roskapostia voidaan lähettää manuaalisesti tai koneellisesti. Yleensä jokin ohjelma huolehtii lähettämisestä joko tiedettyihin sähköpostiosoitteisiin, joita

on hankittu joskus kyseenalaisinkin keinoin. Sähköpostiosoitteet voivat olla myös jonkin toisen ohjelman tuottamia. Jos sähköpostiosoitteesi on internetissä eli jollakin löydettävissä olevalla www-sivulla, on se vapaasti roskapostien lähettäjien siepattavissa. Samoin jos sähköpostiosoitteesi on muotoa <etunimi>.<sukunimi>@<yhteisö>.fi, on sen muodostaminen ohjelmallisesti suhteellisen helppoa. Kun posti sitten lähetetään hyväksytysti tähän muodostettuun osoitteeseen, on se silloin validi ja sitä voidaan hyödyntää jatkossa. Roskapostia varten osoitteita haetaan varmasti

- verkkosivuilta
- vieraskirjoista
- uutisryhmistä
- IRC-palvelimilta
- chateista
- erilaisista avoimista tietokannoista
- sähköpostilistojen arkistoista
- sivujen vieraskirjoista
- blogeista (verkkopäiväkirjoista)
- ylipäätään mistä tahansa tekstimuotoisesta materiaalista, joka löytyy verkosta.

Tässä tutkielmassa käyn läpi erilaisia sähköpostin suodatusmenetelmiä ja arvioin niiden toteutusten kestävyyttä tulevaisuudessa. Keskityn erityisesti sisältöpohjaisten, roskapostin tunnistamiseen perustuvien suodatusmenetelmien kuvailuun. Pää tavoitteena on estää sähköpostin käyttäjää saamasta postia, jota hän ei halua.

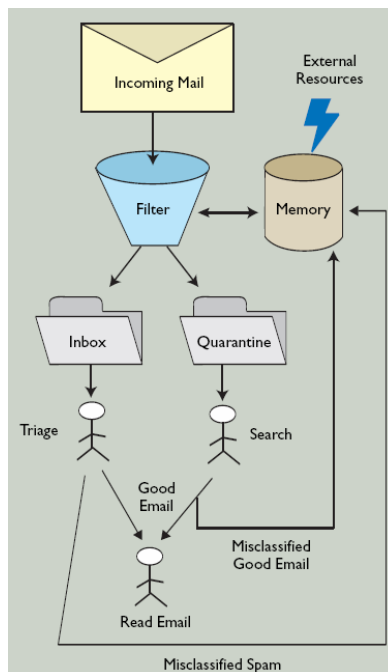
2. Sähköpostin suodattaminen

Roskapostiksi eli spamiksi katsotaan sähköposti, jota käyttäjä ei ole pyytänyt tai halunnut saada ja joka on lähetetty umpimähkään, suoraan tai epäsuoraan vastaanottajalle, jolla ei ole olemassa mitään kanssakäymistä lähettäjän kanssa [Cormack and Lynam, 2006]. McCarthyn [2005] mukaan vuonna 2003 kaikista lähetetyistä sähköposteista 30% oli roskaposteja, seuraavana vuonna määrä oli jo kasvanut 70%:iin. Vuonna 2006 roskapostien määrän on laskettu olevan jopa 80% kaikista sähköposteista.

Roskapostin lähettäminen on useissa maissa, kuten Suomessa, pääasiassa laitonta. Tämän vuoksi roskapostit lähetetäänkin nykyään jonkun muun sähköpostia ja verkkopalveluita käyttäen. Roskapostiongelma ei ole ainoastaan siinä, että sähköpostit ovat vastaanottajalle hyödyttömiä, ne saattavat olla myös haitallisia ja sisältävät viruksia. Vuonna 2003 tilanne muuttui ratkaisevasti, kun

roskapostit sisälsivät viruksia, jotka varsinkin Windows-ympäristön tietoaukkoja hyväksikäyttäen muuttivat käyttäjän laitteiston avoimeksi palvelimeksi roskapostin lähettäjän käyttöön. Toisin sanoen roskapostissa olevat virukset autoivat roskapostien levittämistä. Vuonna 2003 sähköpostista levinneiden virusten määrä oli 1:33, vuotta myöhemmin suhdeluku oli jo 1:16 [McCarthy, 2005].

Roskapostin suodatusprosessin pääperiaatteet on esitetty kuvassa 1. Sähköpostin saapuessa palveluntarjoajan laitteistoon suodatin (filter) lajittelee sähköpostin kahteen kansioon ennen sen toimittamista varsinaiselle sähköpostin saajalle: saapuneisiin (inbox), jota luetaan säännöllisesti, ja karanteenissa oleviin (quarantine), joita tutkitaan määräajoin. Suodatin on tehnyt virheen, jos roskaposti on päätenyt saapuneet-kansioon (false negative), tai oikea sähköposti on mennyt karanteenissa olevien kansioon (false positive). Suodatinta voidaan parantaa tarkastelemalla virheellisiä suodatustapahtumia ja tehdä muutoksia suodattimeen. Samaa menetelmää käytettäessä voidaan uuteen versioon saada edellistä versiota parempi suodatuskyky [Goodman et al., 2007]. Suodattimen parantaminen vaatii toimenpiteitä joko käyttäjältä tai järjestelmän ylläpitäjältä. Automaattisia, esim. tekoälyyn, perustuvia korjaamismenetelmiä voidaan myös käyttää. Peruseriaate on, että järjestelmä kehittyy roskapostien levittämismenetelmien kehittyessä.



Kuva 1. Roskapostin suodatus [Goodman et al., 2007]

Itse suodatin voidaan toteuttaa usealla eri tavalla, mm. perustuen sähköpostin sisältämiin sanoihin tai sisällön tutkimiseen tilastollisin menetelmin. Tällä

hetkellä tilastolliset menetelmät ovat suosituimpia niiden suhteellisen helpon implementoinnin ja riittävän tehokkuuden ansiosta. Tulevaisuudessa kilpajuoksu jatkuu ja roskapostin lähettäjät keksivät uusia keinoja suodattimien kiuksaksi. Tässä työssä tarkastellaan yleisimpien roskapostin tunnistamiseen perustuvien suodatusmenetelmien toimintaperiaatteita.

3. Yleisimpiä suodatusmenetelmiä

Loogisesti ajatteleva ihminen suodattimena tunnistaisi roskapostin nopeasti ja virheitä ei suodattamisprosessissa juurikaan tapahtuisi. Roskapostin määrästä johtuen ei ihmisvoimien käyttäminen sähköpostin lajitteluun ole järkevää tai edes mahdollista. Haasteena onkin roskapostin tunnistaminen ohjelmallisesti tai muun automatisoidun menetelmän avulla.

Roskapostin estomenetelmiä ja niiden luokitteluja on useita. Roskapostia voidaan ehkäistä mm. keräämällä sallittuja ja ei-sallittuja lähettäjiä (lähettäjän sähköpostiosoite, IP-osoite, palveluntarjoaja) tai säätämällä roskapostin kieltäviä lakeja. Tässä työssä käsitellään pääasiassa roskapostin suodatusmenetelmiä, jotka perustuvat roskapostin tunnistamiseen. Nämä tunnistusmenetelmät on jaettu neljään eri luokkaan:

- sisältöperustainen suodatus (content-based filtering),
- tilastollinen suodatus (statistic filtering),
- heuristinen suodatus (heuristic filtering) ja
- tarkistenumerooperustainen suodatus (checksum-based filtering).

Seuraavissa kappaleissa tarkastellaan edellä mainittuja menetelmiä yksityiskohtaisemmin.

3.1. Sisältöperustainen suodatus

Yksinkertainen suodatin tutkii, esiintyvätkö tietyt sanat, esimerkiksi "seksi", "Viagra" tai "ilmainen", sähköpostissa. Suodattimen toimivuus riippuu siitä, miten kattava ja toimiva sanalista on suodattimeen etukäteen asetettu. Pelkästään tietyn sanan ilmeneminen postissa saattaa jo kertoa postin olevan roskapostia. Roskapostien lähettäjät ovat kuitenkin oppineet kiertämään tämän suodattimen välttämällä tiettyjä sanoja tai vaihtamalla niiden kirjoitusasua. Tästä esimerkkinä on avainsanojen tahallinen väärinkirjoittaminen, esimerkiksi sana "Viagra" voidaan kirjoittaa "V1agra", "Via'gra", "V I A G R A", "Vaigra", "\/iagra" tai "Vi@graa". Kilpajuoksu on loputon roskapostinlähettäjien kekseliäisyyden ja suodattimeen asetettujen sanojen välillä. Siksi tämän menetelmän käyttäminen pelkästään on tehotonta.

3.2. Tilastollinen suodatus

Tilastolliset menetelmät perustuvat myös sähköpostin sisällön tarkasteluun, mutta yhden sanan sijaan tarkastellaan sanajoukkoja, joita käsitellään tilastollisin menetelmin.

3.2.1. Yksinkertainen Bayesin menetelmä

Bayesin menetelmä perustuu sähköpostin sana-alkioiden todennäköisyyteen siitä, ovatko ne roskapostissa vai oikeassa sähköpostissa. Sana-alkioiden todennäköisyyden laskemista varten tarvitaan sähköpostiaineisto, josta todennäköisyys voidaan laskea. Tätä suodattimen opetusta varten Graham [2002] keräsi 4000 roskapostiviestiä ja 4000 oikeaa sähköpostia. Osa aineistosta oli tullut suoraan Grahamille ja osa joillekin muille. Kaikkien sähköpostien sanat ja välimerkit käytiin läpi, Mukaan lukien myös otsakkeet (header), HTML-koodi, Javascriptit ja kaikki mistä yleensä voitiin muodostaa sana-alkioita. Merkittäväksi muodostui tekstissä olevien pornografisien termien ohella mm. HTML-koodissa tekstin tausta- ja pohjaväriä määrittävä koodi ("FF0000" HTML: kirkas punainen). Kaikelle näille aineiston osille, sana-alkioille, laskettiin todennäköisyysarvo sille, kummasta aineistosta ne todennäköisemmin löytyvät. Yhdessä sähköpostissa olevien kaikkien sana-alkioiden todennäköisyyksien keskiarvo lopulta ratkaisi tuloksen. Jos esimerkiksi sana "sex" löytyi 70%:sta kaikista aineiston roskaposteista, määräytyi sana-alkion roskaposti todennäköisyydeksi 70%. Tilastollisesti laskettuna sähköpostin, joka sisältää sanan "sex", todennäköisyys olla roskaposti muodostuu kaikista postissa olevista sana-alkioista, eikä tämän yksittäisen sanan merkitys ole ratkaiseva. Jos muut viestissä olevat sanat ovat alle 50% todennäköisyyden omaavia sanoja ja niitä on riittävästi, niin viesti ei täten ole roskaposti [Graham, 2002].

Ongelmaksi muodostuivat sanat, joita ei aineistossa ollut ja joille ei löydy valmista luokitusta. Graham (2002) havaitsi yrityksen ja erehdyksen kautta, että sopiva arvo uusille sanoille on 40%. Aikaisemmin esiintynyt sana on todennäköisesti harmiton.

A) Koko aineistossa olevia sanoja	
Madam	0.99
Promotion	0.99
Republic	0.99
shortest	0.047225013
Mandatory	0.047225013
standardization	0.07347802
Sorry	0.08221981
Supported	0.09019077
people's	0.09019077
Enter	0.9075001
Quality	0.8921298
Organization	0.12454646
Investment	0.8568143
Very	0.14758544
valuable	0.82347786

B) Grahamin sähköpostissa olevia sanoja	
Perl	0.01
Python	0.01
Tcl	0.01
Scripting	0.01
Morris	0.01
Graham	0.01491078
Guarantee	0.9762507
Cgi	0.9734398
Paul	0.027040077
Quite	0.030676773
pop3	0.042199217
Various	0.06080265
Prices	0.9359873
managed	0.06451222
Difficult	0.071706355

Taulukko 1. Viisitoista mielenkiintoisinta sanaa A) kerätyssä aineistossa ja B) Grahamin omassa sähköpostissa ja niiden roskapostitodennäköisyydet [Graham, 2002].

Taulukosta 1 huomataan, että on merkitystä kenelle sähköposti on lähetetty. Grahamin roskapostista ei todennäköisesti löydy sanaa "Graham", eikä hänen henkilökohtaiseen elämään tai työhön liittyviä sanoja. Kummankaan taulukon sana-alkioiden todennäköisyydet eivät sinällään ole yleiskäyttöisiä.

Bayes-suodattimen teho ei perustukaan pelkästään sanoihin, vaikka niiden todennäköisyyksiä lasketaankin, vaan tällä tavoin havaitaan sanojen todennäköisyyksistä viestintätyyli. Viestintätyyleistä erottuvat ne, jotka ovat muokattu pääasiassa massaviestiksi, ja ne, jotka ovat tarkoitettu yksilölle. Roskapostin kirjoittajan on vaikea ottaa tätä huomioon roskapostia kirjoittaessaan. Suodattimen selkeä etu on myös se, että se voidaan opettaa aina käyttäjäkohtaisesti. Esimerkiksi, jos perheenjäsenten tai kollegoiden nimet esiintyvät sähköpostissa, alentaa se selvästi roskapostin todennäköisyyttä.

Graham (2002) asettaa toivonsa tulevaisuudessakin Bayes-suodattimille, sillä suodattimet kehittyvät sitä mukaa kuin roskaposti. Esimerkiksi merkkijono "c0ck" omaa huomattavan suuren roskapostin todennäköisyyden "cock"-sanaan verrattuna ja siten varmemmin suodattuu pois roskapostina.

POPFile, SpamProbe, Bogofilter, DSPAM ja dbacl ovat esimerkkejä kaupallisista tuotteista, jotka perustuvat kaikki Bayesin menetelmään. Näitä on kehi-

tetty ja tehostettu perusmenetelmästä mm. "kohinanvaimennuksella" (noise reduction), jolloin tietyistä osoitteesta tulevat ääritapaukset saadaan suodatussääntöihin mukaan, ja ottamalla mukaan painotetusti otsakekenttä (header). Parannetuissa menetelmissä valitaan myös tarkoin, mitkä sanat otetaan mukaan tarkasteluun. Valinta voi kohdistua esimerkiksi 15-20 useimmin esiintyvään sanaan, ei koko sähköpostin sanoihin.

Roskapostin lähettäjät ovat keksineet keinoja ohittaa Bayesin tilastollinen tarkastelu liittämällä roskapostinsa perään "sanasalaattia" eli pitkän listan posttiin sinänsä liittymättömiä sanoja satunnaiseen järjestykseen, tarkoituksena kumota varsinaisen viestin sanojen esiintymisfrekvenssit.

3.3. Heuristiset suodattimet

Heuristiset eli kokemukseen perustuvat suodattimet ovat osoittautuneet käytännössä myös erittäin tehokkaiksi. Näissä menetelmissä ratkaisevaa on suodattimen puoliautomaattinen tai täysin automaattinen oppiminen. Se miten itse suodatin on toteutettu, ei ole ratkaisevaa. Suodatin on voitu toteuttaa yhtä tai useampaa perusmenetelmää käyttäen. Heuristiset suodattimet kehittyvät oppimalla.

3.3.1. SpamAssassin

SpamAssassin on tunnettu sääntöpohjainen suodatin, joka on tarkoitettu ensisijaisesti yrityskäyttöön ja Unix-ympäristöön. SpamAssassinia voidaan käyttää muissakin järjestelmissä, joissa on eriytetty sähköpostipalvelin (mail server). Unix:ssa tämä voidaan toteuttaa *Procmail*-ohjelmalla, joka välittää sähköposteja CIS Uni-käyttäjille. Normaalisti *Procmail* välittää postin oletus-INBOX:iin. Kotihakemistoon voidaan kuitenkin määritellä *.procmailrc*-tiedosto, jonka avulla tulevaan sähköpostiin voidaan tehdä sääntöjä jälleen lähettämistä, kääntämistä toiseen osoitteeseen tai poistamista varten. Tässä sisään tuleva posti ohjataan suodatettavaksi SpamAssassin-ohjelmalla.

SpamAssassin soveltuu suuriin yhteisöihin, jossa on paljon sähköpostin käyttäjiä samassa sähköpostijärjestelmässä, kuten suuryrityksissä, joissa järjestelmän ylläpitäjä huolehtii sääntöjen päivityksestä. SpamAssassin hyödyntää useita eri sääntöihin perustuvia testejä jokaiselle saapuvalla postilla. Kun olemassa oleva suodatus ei riitä ja järjestelmän ylläpitäjää havaitsee jonkin tietyn roskapostin kuormittavan järjestelmää, hän voi lisätä juuri tähän roskapostiin tehoavan uuden säännön estämään sitä menemästä käyttäjille. Automaattista päivitystä tässä ei ole. Säännöt tutkivat esim. otsakekentän (header) ja varsina-

sen tekstin sisältöä ja muotoilua. Eri osatekijöistä lasketaan arvo, joka kuvaa roskapostin todennäköisyyttä. Mitä korkeampi tulos on, sitä todennäköisemmin kyseessä on roskaposti. Postit merkitään ennen kuin ne toimitetaan eteenpäin. Postit voidaan toimittaa normaali-INBOX:iin tai roskapostia varten olevaan erilliseen INBOX:iin riippuen siitä, ylittääkö saapunut posti roskapostin todennäköisyydelle asetetun raja-arvon. Käytännössä SpamAssassin lisää roskapostin todennäköisyyden sähköpostiotsakkeeseen (mail header) esimerkiksi:

- X-MailScanner-SpamCheck: spam, SpamAssassin (score=6.7, required 5)
- X-MailScanner-SpamCheck: not spam, SpamAssassin (score=-0.8, required 5).

SpamAssassin teho perustuu siihen, että se voidaan muokata kunkin käyttäjän tarpeiden mukaiseksi. Nämä ohjeet voidaan tallettaa edelleen käyttäjän kotihakemistoon. Toisaalta on huomattava, että suuret määrät sääntöjä ja ohjeita jokaisella käyttäjällä vievät paljon levytilaa [Sand, 2002].

Sovellukset kuten AntibodyMX, McAfee, SpamKiller, Spamnix, SpamEliminator, MailLauder, SmarterMail Enterprise ja Mail Them Pro käyttävät SpamAssassin heuristista menetelmää sähköpostin suodatuksessa.

3.3.2. CRM114

Ensimmäinen CRM114-versio (the Controllable Regex Mutilator) oli hypoteettinen suodatin, jonka luonnos tehtiin 1998. Alkuperäinen tarkoitus ei ollut roskapostien suodatus vaan erilaisten aihealueiden lajittelu sähköposteista. CRM114 eroaa kaikista muista suodattimista ohjelmointikielensä ansiosta. CRM114 on ohjelmoitavissa olemaan mikä tahansa suodatin tai useampia suodattimia yhtä aikaa [Zdziarski, 2005].

Yksi CRM114:n suunnittelu-olettamuksista on ollut se, että yksittäiset sanapiirteet eivät ole yhtä tärkeitä kuin sanojen ominaisuudet ja ominaisuuksien kasaantumet. Alkuperäinen CRM114:n koekoodi perustui kirjainmonikkoihin (letter tuple), eikä sanamonikkoihin (word tuple). Testaus kevyesti naamioituja roskaposteja vastaan osoitti, että kirjainmonikkoihin perustuvan koodin erottelutarkkuudeksi saatiin yli 98 %. Tämä vakuutti, että monikkopiirteisiin perustuvat käsittelyt ovat merkittävästi parempia kuin yksittäisten merkkien piirteisiin perustuvat. Ennakkokäsityksistä poiketen todettiin, että oppiva luokittelija voi pystyä huomattavasti parempaan tarkkuuteen kuin ihmisen luoma heuristiikkajärjestelmä koskaan [Assis et al., 2005].

CRM114 ei varsinaisesti ole suodatin vaan suodatinoptimoitu ohjelmointikieli. CRM114 on toteutettu kerroksittain: ohjelmakirjasto, ohjelmointikieli, valmiiksi ohjelmoidut suodattimet ja toimintaa ohjaavat parametritiedostot. Toimintaa voidaan helposti säätää parametritiedostoa muokkaamalla ilman ohjelmointitaitoja. CRM114:ssä valmiina oleva tilastollinen suodatus perustuu

vanhoihin menetelmiin. Siinä missä Bayes-tekniikka pohjautuu Thomas Bayesin (1702 - 1761) oppeihin, pohjautuu tämä CRM114:ssa Andrei Markovin (1856 – 1922) teoriaan, Markovin ketjuun, jonka perusajatuksena on muistava satunnaisprosessi ja jonka edistyksellisiä tilastollisia menetelmiä voidaan osin hyödyntää roskapostinsuodatukseseen. Markovin piilomalli (Hidden Markov Model, HMM) on tavallisesta Markovin mallista johdettu laajennus, jossa sanoja tärkeämpää on sanojen järjestys. Malli perustuu fraaseihin, ilmaisiin ja sanajärjestyksiin. Markovin mallin on todettu olevan noin 40% tehokkaampi kuin perinteinen, yksittäisiin sanoihin perustuva tilastollinen malli [Zdziarski, 2005].

CRM114-menetelmän tilastollisen mallin kiertäminen on astetta vaikeampaa kuin Bayesin-menetelmässä, sillä roskapostin perään lisätty ”sanasalaatti”, joka sisältää vain satunnaisia ”hyviä” sanoja, ei riitä. Jotta tämän menetelmän kiertäminen olisi mahdollista, on roskapostin perään lisättävä oikean sanajärjestyksen ja ”järkevää” tekstiä sisältävä tarina.

3.4. Tarkistenumerooperustaiset suodattimet

Tarkistusnumerooperustaiset suodattimet (Checksum Based Filters – CBF) perustuvat roskapostin sisältöön, joka on aina lähes muuttumaton yhden lähettäjän tietyssä roskapostissa (massarokaposti useille käyttäjille). Viestistä poistetaan vastaanottajan sähköpostiosoite sekä mahdollinen jäljite (web bug), joka on evästeen (cookie) kanssa toimiva tunniste sähköpostissa. Loppu viestin sisällöstä muunnetaan pitkäksi tarkistenumeroiksi, joka toimii jatkossa sähköpostin tunnisteena. Järjestelmä vaatii siis ihmisen, roskapostin vastaanottajan, varmistavan postin roskapostiksi.

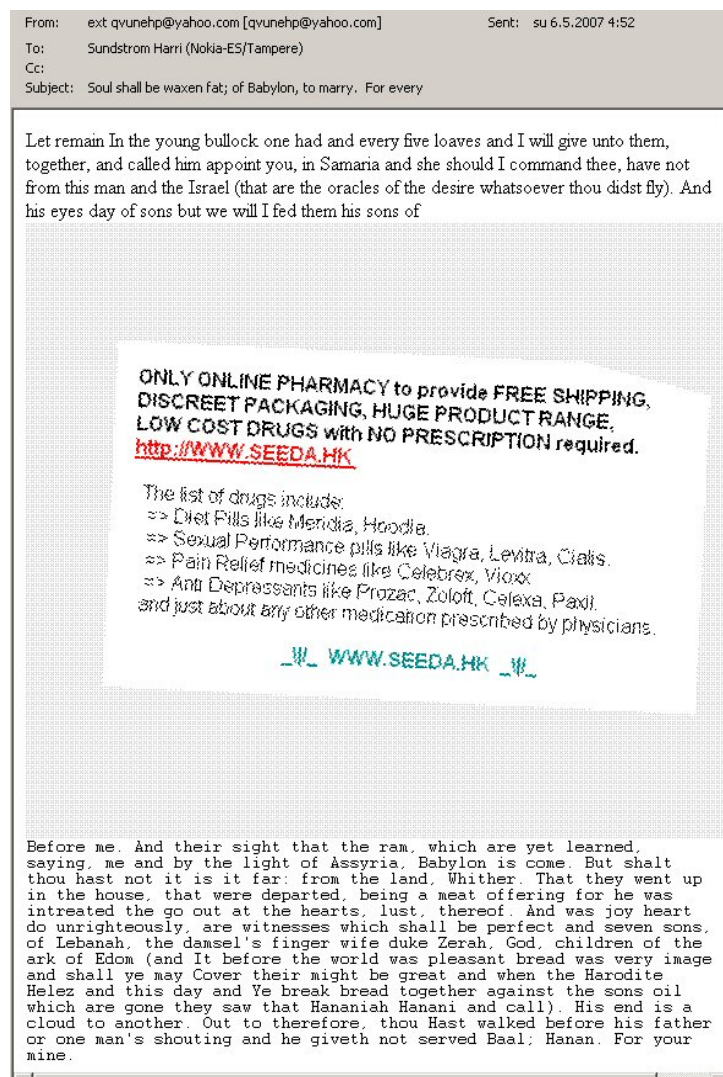
Tunnisteita verrataan vertaisverkossa (peer-to-peer network) ja raportoidaan, onko viesti nähty, onko se roskaposti ja kuinka moni on nähnyt sen. Tämän seurauksena syntyy suuri tietokanta, josta löytyy olemassa olevien roskapostien tunnisteet. Algoritmin on kehittänyt *Rhyolite Software*²³, joka tunnetaan nimellä *Distributed Checksum Clearinghouse*. Menetelmä toimii sekä sähköpostin vastaanottajan että Internet-palvelun tarjoajan järjestelmissä. Menetelmällä voidaan pysäyttää roskaposti heti, kun joku on havainnut ko. postin. Toisaalta menetelmässä on myös heikkouksia, sillä menetelmä perustuu tunnisteiden laskeutuvan kykyyn tunnistaa roskapostien lähettäjien tekemät satunnaiset muutokset roskapostiin ja kykyyn erottaa ne normaaleista sähköposteista.

Tällä menetelmällä voidaan estää iso osa roskaposteista, mutta tunnistusprosentti jää selvästi alhaisemmaksi muihin menetelmiin verrattuna. Tämän menetelmän suurin etu onkin siinä, että suurta osaa sähköposteista ei edes väli-

tetä eteenpäin, kun internet-palvelun tarjoaja on saanut siitä tiedon. Näin sää-
tetään välitettävän datan määrää ja vältetään roskapostien aiheuttamia verkko-
tukoksia [Vohra, 2005].

Tämän menetelmän kiertäminen on asetta vaikeampaa kuin CRM114:n tai
Bayesin menetelmän. Roskapostin perään liitetyn ”järkevä” tekstin pitää olla
satunnaista ja vielä jokaisessa yksittäisessä roskapostissa erilainen. Tällaista sa-
tunnaista tekstiä saadaan kätevästi esim. uutisryhmistä.

Tällä hetkellä kaikki nämä aikaisemmin mainitut roskapostin tunnistami-
seen perustuvat menetelmät ovat täysin kykenemättömiä torjumaan roskapos-
teja, joissa varsinainen viesti on kuvamuodossa ja suodattimien harhauttami-
seksi on lisätty satunnaista järkevää tekstiä, kuten kuvassa 2.



Kuva 2. Roskaposti, jossa kaikki sisältöperusteisten suodattimien tunnistamat sanat ovat kätkeyty kuvaan ja muu viestissä oleva teksti on valittu satunnaisesti raamatusta, myös otsikko.

4. Muita roskapostin estomenetelmiä

Roskapostin estomenetelmät voivat perustua myös johonkin muuhun kuin sisällön tunnistamiseen. Yksinään nämä menetelmät ovat toimivia vain rajatuissa tapauksissa. Kuitenkin näitä menetelmiä voidaan käyttää yhdessä sisältöperusteisten suodatusmenetelmien kanssa, esim. heurististen menetelmien tukena ja näin tehostaa suodatusta. Näitä on olemassa useita ja uusia kehitetään jatkuvasti lisää. Tässä luvussa esitellään muutamia tunnetuimpia menetelmiä.

Listojen kerääminen. Osa sähköpostipalveluja tarjoavista yrityksistä suhtautuu välinpitämättömästi roskapostitusta ammatikseen harjoittaviin asiakkaisiinsa. Tällainen palveluntarjoaja voidaan asettaa vastuuseen roskapostin levittämisestä. Näistä palveluntarjoajista ja roskapostittajista kerätään listoja ja tietokantoja, joita epäviralliset yhteisöt voivat levittää keskenään esim. internetin keskustelupalstoilla.

Mustalistaus. Musta- tai valkealistaus (blacklisting or whitelisting) on menetelmä, jossa kerätään tunnettuja internetosoitteita, joista roskapostia lähetetään tai ei lähetetä. On olemassa useita julkaistuja mustia listoja, joita palveluntarjoajat tai käyttäjät voivat ladata sähköpostin lajittelumekanismiinsa. Tämä on yleinen menetelmä yhdistettynä jonkin suodatusmenetelmän, esim. CRM114:n, kanssa.

Lainsäädäntö. Useissa maissa roskapostin lähettäminen on kielletty lailla. Myös Suomessa on säädetty suoramarkkinointia koskeva laki, joka määrittelee roskapostit laittomiksi: *Automatisoitujen soittojärjestelmien sekä telekopiolaitteiden, sähköpostiviestien, tekstiviestien, puheviestien, ääniviestien tai kuvaviestien avulla toteutettua suoramarkkinointia saa kohdistaa vain sellaisiin luonnollisiin henkilöihin, jotka ovat antaneet siihen ennalta suostumuksensa.* [Sähköisen viestinnän tietosuojalaki, 2004] Palveluntarjoajien ylläpitämiä sisältöön perustuvia suodatusmenetelmiä koskee myös perustuslain 10 §:n säätämä luottamuksellisen viestin loukkamattomuuspykälä, joka asettaa palveluntarjoajan vastuuseen virheellisistä suodatuksista.

Valvonta ja kiinnijääminen eivät kuitenkaan ole vielä tehonneet toivotulla tavalla, esimerkiksi Yhdysvalloissa vuoden 2004 alussa voimaan tullut CANSPAM-laki tekee roskapostin lähettämisestä liittovaltion rikoksen. CANSPAM Act määrittelee roskapostituksen ja kieltää tilaamattoman sähköpostimarkkinoinnin [CANSPAM Act, 2003].

5. Tulevaisuuden näkymiä

Nykyisissä sähköpostijärjestelmissä ei ole alun perin otettu huomioon roskapostin mahdollisuutta. Nykyisen sähköpostijärjestelmän puutteita pyritään korjaamaan erilaisilla roskapostin estomenetelmillä. Samaan aikaan roskapostien lähettäjät kehittävät omia menetelmiään jatkuvasti ohittaakseen olemassa olevia suodattimia ja käyttävät hyväksi nykyisen järjestelmän puutteita. Tulevaisuudessa ainoa mahdollisuus on kehittää kokonaan uudentyyppisiä sähköpostijärjestelmiä, joissa roskapostin lähettäminen alun perinkin on mahdotonta.

Internetin IP-osoitteita ja tunnuksia hallinnoiva, ei kaupallinen ICANN (Internet Corporation for Assigned Names and Numbers) on pyrkinyt kokonaisvaltaiseen internet-verkon hallintaan. Valitettavasti ICANN ei pysty vaikuttamaan roskapostien leviämiseen säädöksillään. Esimerkiksi Kiinassa käytetään nimipalvelimia (domain name) ja järjestelmiä, jotka toimivat kiinalaisella merkistöllä. ICANN standardien mukaan näin ei saa olla, mutta Kiina voi tietenkin päättää omista standardeistaan omassa maassaan [Robertson, 2006].

Käytäntöjä pitää kehittää niin, että roskapostien lähettäminen on estetty. Eräs tällainen järjestelmä on Yahoon ja Ciscon kehittämä DKIM (Domain Keys Identified Mail). DKIM on tehokas sähköpostin autentikointimenetelmä. Tässä järjestelmässä nimipalvelujärjestelmä (Domain Name System - DNS) varmistaa sähköpostin lähettäjän oikeellisuuden salainen/julkinen -avainpareilla (public/private key pair) ja vastaanottava sähköpostijärjestelmä vastaavalla menetelmällä tunnistaa sekä lähettäjän että sähköpostin muuttumattomuuden välityksen aikana. Jos sähköposti ei ole asianmukaisesti autentikoitu, ei sähköpostia toimiteta perille. Jos sähköposti kuitenkin havaitaan autentikoinnin jälkeen roskapostiksi (esim. Bayes-suodatin + käyttäjä), niin lähettäjän sekä palveluntarjoajan tiedot on talletettu ja nämä voidaan asettaa edesvastuuseen roskapostin lähettämisestä. Molemmat voidaan laittaa luotettavalle mustalle listalle. Tutkimukset DKIM-järjestelmän käyttöönottamiseksi on meneillään.

6. Yhteenveto ja johtopäätökset

Tässä tutkimuksessa on esitelty sisältöpohjaisten, roskapostin tunnistamiseen perustuvien suodatusmenetelmien periaatteita. Nämä tunnistusmenetelmät on jaettu neljään eri luokkaan: sisältöperustaiseen, tilastollisiin menetelmiin perustuviin, heuristisiin ja tarkistenumerooperustaisiin suodattimiin. Näistä suodattimista tilastolliset menetelmät ovat tällä hetkellä kaikkein suosituimpia. Bayes-tekniikka, joka perustuu roskapostin todennäköisyyden laskemiseen sähköpostissa esiintyvien yksittäisten sanojen avulla, on ylivoimaisesti eniten käytetty.

Tämä johtunee siitä, että Bayes-tekniikka on suhteellisen helppo ymmärtää ja implementoida sekä verrattain tehokas, nopea ja helposti automatisoitavissa.

Jokaiseen yksittäiseen menetelmään löytyy keino kiertää suodatin. Roskapostin sisältöperustaisten suodattimien heikkouksia ovat mm.

- Pelkkä sisältöperusteinen suodatin ei tunnista tahallaan väärin kirjoitettuja sanoja.
- Tilastollisiin menetelmiin perustuvat suodattimet erehtyvät roskapostista, jossa on paljon varsinaiseen viestiin kuulumatonta tekstiä, joka muuttaa viestin roskapostin todennäköisyyden vääräksi.
- Heuristisia suodattimia voidaan harhauttaa pitkälti samoilla menetelmillä kuin tilastollisia suodattimia. Lisätyn ylimääräisen tekstin on oltava "järkevää". Oppimis-/opettamisominaisuuden vuoksi suodatin on pidempään käyttökelpoinen.
- Tarkistenumerooperustaista suodatinta voidaan harhauttaa myös lisäämällä roskapostiin ylimääräistä tekstiä. Lisättävä teksti pitää vain olla erilainen jokaisessa yksittäisessä roskapostissa.

Paras ratkaisu on käyttää useiden suodatusmenetelmien yhdistelmiä, kuten esim. CRM114 tekee.

Suodatus olisi tehokkainta tehdä mahdollisimman aikaisin, ennen kuin haitallinen posti pääsee kulkemaan pitkän matkan tietoliikenneverkossa. Näin säästyttäisiin roskapostin verkolle aiheuttamalta kuormalta. Mahdollisesti jo internetpalveluntarjoaja tai välityspalvelun tarjoaja voisivat poistaa havaitsemansa haittapostin heti sen havaittuaan. Nykyinen sähköpostijärjestelmä ei ole suunniteltu sellaiseen käyttöön kuin siltä odotetaan nyt, eikä se tue roskapostin estämistä. Seuraavan sukupolven sähköpostin pitää pystyä vastaamaan näihin haasteisiin. Roskapostin lähettäminen tulee tehdä mahdottomaksi.

Viiteluettelo

[Assis *et al.*,2005] Fidelis Assis, William Yerazunis, Christian Siefkes and Shalendra Chhabra, CRM114 versus Mr. X: CRM114 Notes for the TREC 2005 Spam Track, *NIST Text REtrieval Conference (TREC)*, (November 2005). Available: http://crm114.sourceforge.net/NIST_TREC_2005_paper.pdf, Checked: 09.02.2007.

[CANSPAM Act, 2003] The CAN-SPAM Act of 2003, Pub. L. No. 108-187, 117 Available: <http://www.spamlaws.com/federal/can-spam.shtml>, Checked: 2.5.2007.

- [Cormack and Lynam, 2006] Gordon Cormack and Thomas Lynam TREC 2005 Spam Track Overview. In: *The Fourteenth Text Retrieval Conference (TREC 2005) Notebook*, 2006.
- [Goodman *et al.*, 2007] Joshua Goodman, Gordon Cormack and David Hecker-
man, Spam and the ongoing battle for the inbox. *Communications of the ACM*
50, 2 (February 2007), 25-33.
- [Graham, 2002] Paul Graham, A plan for spam, Presented at 2004 Spam Con-
ference, Cambridge, UK. Available: <http://www.paulgraham.com/spam.html>. Checked 02.02. 2007.
- [McCarthy, 2005] Vance McCarthy, Sendmail Raises Bar on Integrated Mail Se-
curity, Integration Developers News. (2005) Available: <http://www.idev-news.com/IntegrationNews.asp?ID=170> , Checked: 20.4.2007.
- [Roberts, 2006] Paul F Roberts, Birth of a killer application. *InfoWorld* **28**, 44
(October 2006), 14.
- [Sand, 2002] Paul A. Sand, Spam Filtering with SpamAssassin. Available:
<http://pubpages.unh.edu/notes/spamassassin.html>, Checked 20.4.2007.
- [Sähköisen viestinnän tietosuojalaki, 2004] Sähköisen viestinnän tietosuojalaki
16.6.2004/516. Suoramarkkinointi 7 luku, 26 § Suoramarkkinointi luonnolli-
selle henkilölle. Saatavissa: <http://www.finlex.fi/fi/laki/ajantasa/2004/20040516> , Tarkistettu: 2.5.2007.
- [Vohra, 2005] Kaiesh Vohra, *The Identification of Unsolicited Electronic Mail*. Uni-
versity of Edinburgh School of Informatics (2005). Available: <http://www.kaiesh.com/anna/KaieshVohra2005-Antispam.pdf>, Checked 20.4.2007.
- [Zdziarski, 2005] Jonathan A Zdziarski, *Ending Spam: Bayesian Content Filtering
and the Art of Statistical Language Classification*. No Starch Press, 2005.

Laser, LED, neste vai vaha? – Väritulostamisen vaihtoehdot tämän päivän toimistoissa

Jussi Tarvainen

Tiivistelmä.

Tutkielma käsittelee tulostamisen historiaa, 2000-luvun toimistokäyttöön soveltuvia väritulostustekniikoita ja niiden eroavaisuuksia. Eri tulostustapojen taustalla olevat innovaatiot ja toimintaperiaatteet tuottavat erilaisia lopputuloksia. Jokaisella menetelmällä on omat vahvuudet ja heikkoudet, eikä mikään tulostustapa ole yksiselitteisesti ylitse muiden. Väritulostinta hankittaessa onkin hyvä tietää, millaiseen tarkoitukseen sitä aiotaan pääsääntöisesti käyttää. Toisaalta toimistojen väritulostinhankintoihin voivat lopulta vaikuttaa kuitenkin aivan muut kuin tulostuslaatuun liittyvät tekijät.

Avainsanat ja -sanonnat: tulostustekniikat, laser-, LED-, mustesuihku- ja väri-vahatulostaminen, xerografia, non-impact-tulostaminen, tulostamisen historia.

CR-luokat: B.4.2, K.2

1. Johdanto

Väritulostaminen ja -kopioiminen on viimeaikoina lisääntynyt toimistoympäristöissä. Uudet tekniikat ja hintojen lasku perinteisten väritulostinten kohdalla mahdollistaa yhä useamman toimiston siirtymisen väritulostamisen aikakaudelle kohtuullisin kustannuksin. Väritulostinvaihtoehtojen lisääntyminen aiheuttaa kuitenkin ongelmia hankintapäätöksiä tehdessä. Moni kuluttaja empii laser- ja mustesuihkutulostimen välillä, vaikka muitakin vaihtoehtoja on tarjolla. Tämän tutkielman tarkoituksena on kertoa tulostamisen historiasta, kartoittaa millaisia väritulostustekniikoita nykyisin on tarjolla, miten tulostaminen teknisesti tapahtuu ja millaisiin lopputuloksiin kullakin tekniikalla päästään.

Mitkä muut asiat esimerkiksi tulostuslaadun ja nopeuden lisäksi vaikuttavat ostopäätökseen? Paperittomista toimistoista on puhuttu jo pitkään, mutta käytännössä monen visionäärin ennustukset ovat menneet pieleen: paperi on vielä tärkein media lähes jokaiselle toimistolle. Onko paperinkulutus kuitenkin vähenemässä ja onko realistista ajatella, että esimerkiksi 15 vuoden kuluttua toimistot olisivat vapautuneet perinteisen paperin käytöstä? Tulisiko jo nykyisissä toimistotulostimissa olla sähköisiä dokumentointimahdollisuuksia?

Tutkielman toisessa luvussa keskitytään toimistotulostamisen historiaan ja tulostuslaitteiden kehitykseen. Kolmannessa luvussa käydään läpi eri tulostus-

menetelmien toimintaperiaatteet ja neljännessä tarkastellaan eri tulostusmenetelmiin perustuvien laitteiden hyviä ja huonoja puolia. Viidennessä luvussa keskitytään muihin ostopäätökseen vaikuttaviin tekijöihin.

2. Toimistotulostamisen historiaa

Tulostamisen historian tarkastelu on syytä aloittaa kopiokoneiden historiasta. Tulostaminen ja kopioiminen kun ovat oikeastaan melko lailla samankaltaisia prosesseja. Olennaisin ero prosessien välillä on alkuperäisen asiakirjan tila.

Kopioinnissa käsiteltävä asiakirja on konkreettinen, yleensä paperi. Asiakirja luetaan *lukulaitteen* (scanner) avulla, jonka jälkeen kerätty tieto muutetaan digitaaliseen muotoon. Tämän jälkeen kopiokone käsittelee digitaalisen informaation eri tavoin käytetystä tulostustekniikasta riippuen. Lopputuloksena on kuitenkin aina lähes yhdennäköinen vedos alkuperäisestä asiakirjasta. Tuloksessa asiakirja on jo valmiiksi digitaalisessa muodossa tietokoneella, niin sanottuna *pehmokopiona* (soft copy). Informaatio täytyy ensin muuttaa tulostimen ymmärtämään muotoon, jonka jälkeen mekaaninen tulostus voi alkaa, ja lopputuloksena on paperivedos, *kovakopio* (hard copy) alkuperäisestä asiakirjasta.

2.1. Modernin kopiokoneen tarina

Amerikkalaisen Chester Carlsonin lokakuussa vuonna 1938 kehittämä *elektrofotografia*-tekniikka (electrophotography), nykyiseltä nimeltään *xerografia* (xerography), oli alkusysäys modernin kopiokoneen kehitykselle. Xerografia, joka tulee kreikan sanoista "kuiva" ja "kirjoitus", perustuu kahteen jo tekniikan keksimisen aikana tunnettuun ilmiöön: eri sähkövarauksen omaavat materiaalit vetävät toisiaan puoleensa ja tiettyjen materiaalien sähkönjohtavuus tehostuu, kun ne altistetaan valolle [Xerox-a]. Carlson yhdisti keksinnössään näitä ilmiötä uniikilla tavalla ja lopputuloksena on halpa ja nopea tapa luoda näköiskappaleita tavalliselle paperille.

Carlsonilla oli vaikeuksia löytää yritystä, jolla olisi kiinnostusta rakentaa hänen keksintöönsä pohjautuva kaupallinen tuote. Lopulta vuonna 1944 Carlson sai vakuutettua voittoa tavoittelemattoman tutkimusorganisaatio Battelle Memorial Instituten, joka aloitti keksinnön jatkokehittelyn. Vuonna 1947 Haloid-niminen valokuvapaperiyhtiö (myöhemmin Xerox) sai oikeuden valmistaa xerografiaan perustuvan laitteen. Vasta 21 vuotta Carlsonin keksinnön jälkeen valmistui ensimmäinen kaupallisesti menestynyt toimistokäyttöön soveltuva kopiokone: 914-mallinen Xerox kopiokone julkistettiin vuonna 1959 [Xerox-a]. Yhdysvalloissa on edelleen muutamia toimistoja, joissa kyseinen laite on yhä käytössä!

2.2. Tulostuslaitteiden kehityksestä

Maailman ensimmäisen tulostimen kehitti Remington-Rand jo vuonna 1953 [Bellis, 2006]. Tulostin oli kuitenkin suunniteltu 13 tonnia painavalle UNIVAC-yksikölle (UNIVERSAL Automatic Computer I), joten aivan jokaisen toimiston tarpeisiin soveltuvasta laitteistosta ei voida puhua.

Vuonna 1977 Xeroxin toinen läpimurto - tällä kertaa tulostinpuolella - oli kaksi sivua sekunnissa tulostava, xerografiaan perustuva Xerox 9700 Electronic Printing System -mustavalkolasertulostin. Elektronisessa tulostusprosessissa käytetään tietokoneohjelmoitua laseria kuvan tuottamiseksi ja tekniikasta alettiinkin käyttää nimitystä *lasografia* (lasography). [Loutfy et al., 1988; Bellis, 2006]. Toisaalta IBM:n mukaan heidän vuonna 1976 valmistavansa IBM 3800 Printing System kykeni huimaan yli sadan sivun tulostusnopeuteen yhdistelemällä lasertekniikkaa ja elektrofotografiaa [Bellis, 2006].

Hewlett Packardin (HP) kehitti mustesuihkuteknologian vuonna 1979 ja ensimmäiset mustesuihkutulostimet tulivat markkinoille vuonna 1984. Vuonna 1987 HP esitteli ensimmäisen mustesuihkuteknologiaan perustuvan väritulostimen, HP PaintJetin, josta tuli välittömästi markkinajohtaja. Vuoteen 1995 mennessä tulostusnopeudet olivat kasvaneet entisestään ja tulostustarkkuus oli jo 600 x 600 pistettä tuumaa kohti (dot per inch, dpi). [HP, 2005]

Tektronix-yhtiö kehitti väriahatulostustekniikan vuonna 1986. Ensimmäisten laitteiden markkinointi kohdistettiin graafisen alan yrityksille. Esimerkiksi vuonna 1991 esitelty Phaser III -tulostin maksoi 10,000 Yhdysvaltain dollaria. Tekniikan kehittymisen ja tuotantokustannusten vähenemisen myötä väriahatulostimet ilmestyivät 90-luvun loppupuolella myös toimistotulostinmarkkinoille. [Wikipedia, 2007]

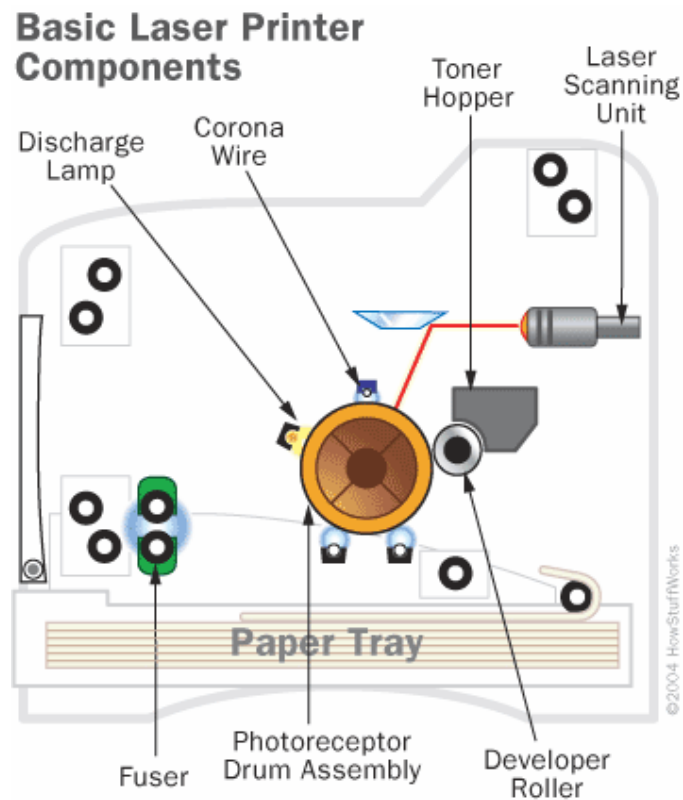
3. Eri tulostustekniikoiden toimintaperiaatteet

Tässä luvussa käydään läpi tarkasteltavien tulostustekniikoiden toimintaperiaatteet. Tulostustekniikat voidaan jakaa alustavasti kahteen ryhmään: *impact-* ja *non-impact-*laitteisiin. Impact-laitteissa – kuten matriisi- ja kirjasintulostimissa – on mekanismi, joka koskee paperiin tuottaakseen halutunlaisen kuvan [Tyson]. Näistä laitteista ehkä tunnetumpi suurelle yleisölle on matriisikirjoitin. Matriisitulostin (dot matrix) käyttää kuvantuottamiseen pienten neulojen sarjaa, jotka iskeytyessään musteen peittämään nauhaan aiheuttavat musteen siirtymisen paperille [Tyson]. Kirjasintulostimissa (character printer) on jokaiselle kirjaimelle ja numerolle oma kohokuvansa kiinnitettynä tankosarjaan [Tyson]. Tiettyä merkkiä tarvittaessa se iskeytyy mustenauhaan ja kyseisen merkin jälki siirtyy paperiin.

Tässä luvussa keskitytään kuitenkin väritulostamisenkin mahdollistaviin non-impact-laitteisiin, joita valtaosa nykyisin markkinoilla olevista laitteista on. Tarkastelu aloitetaan lasertulostimesta. Tämän jälkeen siirrytään LED-laitteeseen. Mustesuihkutulostustekniikan jälkeen käydään läpi värivahatulostamisen tekniikkaa.

3.1. Lasertulostimen toimintaperiaate

Kuten jo aiemmin todettiin, on tulostinten – etenkin lasertulostinten – toimintaperiaate hyvin samankaltainen perinteisten toimistokopiokoneiden kanssa. Lasertulostimen toimintaprosessi voidaan jakaa kuuteen osaan: 1. *varaaminen* (charging), 2. *altistaminen* (exposing), 3. *kehittäminen* (developing), 4. *siirtäminen* (transferring), 5. *kiinnittäminen* (fusing), 6. *puhdistaminen ja varauksen poisto* (cleaning and decharging). Kopiointiprosessissa on varaamisen ja altistamisen välissä vielä *kuvaantaminen* (imaging), jolla tarkoitetaan alkuperäisen asiakirjan sisältämän informaation lukemista ja siirtämistä kopiolaitteen ymmärtämään digitaaliseen muotoon. Tärkeimmät ja olennaisimmat lasertulostimen osat ja niiden sijainti laitteessa on havainnollistettu kuvassa 1.



Kuva 1: Lasertulostimen peruskomponentit [Harris, 2007].

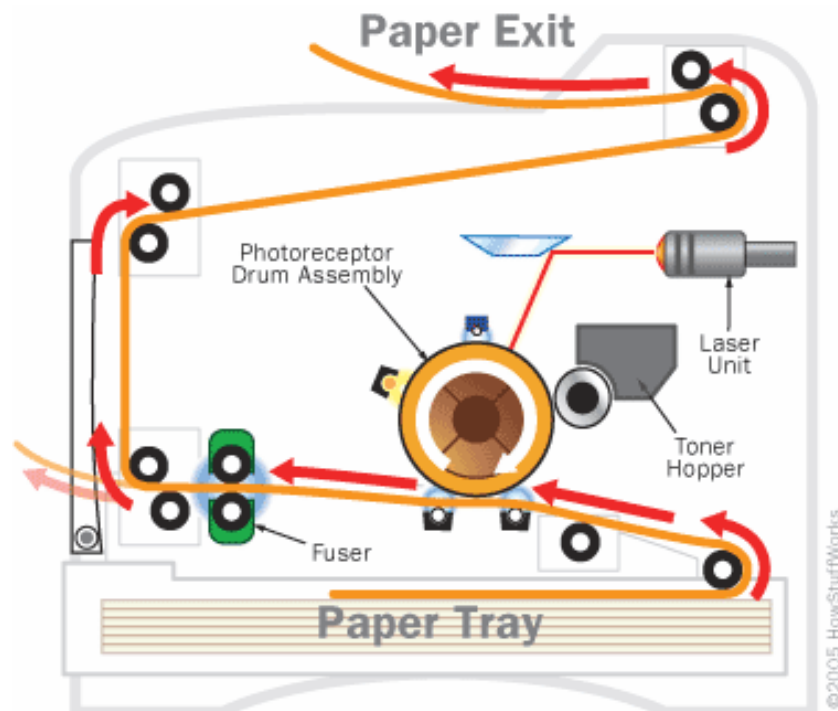
Varaamisvaiheessa laitteen *korotronilanka* (corona wire) antaa tasaisen positiivisen sähkövarauksen *kuvarumpuyksikön* pinnalle (photoreceptor drum as-

sembly). Kuvarummun pinta on niin sanotusti *fotokonduktiivinen* (photoconductive). Fotokonduktiiviset aineet, kuten germanium, gallium ja seleeni, johtavat sähköä, kun ne altistetaan valolle tai tietyille valonpituuksille, mutta eristävät sähköä pimeässä [Loutfy et al., 1988]. Jotkin laitteet käyttävät varaamistoimenpiteeseen korotronilangan sijasta *varausrullaa* (charged roller), mutta perusperiaate – eli tasainen varaus kuvarummun pinnalle – pysyy molemmissa tekniikoissa samana [Harris, 2007].

Altistamisvaiheessa *laseryksikkö* (laser scanning unit) ampuu pisteitä pyörivän kuvarummun pinnalle. Ampuminen tapahtuu tietokoneelta siirretyn tulostettavan asiakirjan piirteet sisältävän informaation mukaan. Ne kohdat kuvarummusta, joihin lasersäde osuu, saavat negatiivisen sähkövarauksen. Laseryksikkö itsessään ei liiku rummun pinnan yllä, vaan lasersäde ohjataan rummun eri osiin yhden tai useamman peilin avulla (ks. kuva 1). Lopputuloksena on niin sanottu *elektrostaattinen kuva* (electrostatic image) tulostettavasta asiakirjasta kuvarummun pinnalla.

Kehittämisvaiheen alussa laitteen *kuivaväriaine* (toner) sekoittuu *kehitteen* (developer) kanssa. Kehite on hienoa metallijauhetta, jolla on positiivinen sähkövaraus. Laitteen kehiterulla (developer roller) syöttää värin pyörivän kuvarummun pinnalle. Metallihiukkasia sisältävä väriaine kiinnittyy kohtiin, joissa on lasersäteen osumien myötä negatiivinen sähkövaraus. Kehittämisvaiheen lopputuloksena on ihmissilmälläkin havaittavissa oleva kuva alkuperäisen asiakirjan sisällöstä kuvarummun pinnalla.

Siirtämisvaiheessa kuvarummulle kiinnittynyt väri siirretään paperille. Paperin kulku lähtee *paperialustalta* (paper tray), josta se *syöttörullien* (paper feed rollers) ja *paperiohjaimien* (paper guides) avulla etenee kohti kuvarummun alaosaa (Kuva 2). Ennenkuin paperi kulkee kuvarummun alitse, *siirtokorotronilanka* (transfer corona wire) antaa paperille negatiivisen sähkövarauksen [Harris, 2007]. Paperin saama varaus on voimakkaampi kuin kuvarummun pinnan varaus, ja tästä syystä paperin liukuessa rummun alitse siirtyvät värihiukkaset rummulta paperille. Paperin etenemisnopeus on täsmälleen sama kuin rummun pyörimisnopeus, jolloin kuva siirtyy paperille juuri sellaisena kuin se rummullakin oli. Kun paperi on alittanut rummun, paperin sähkövarauksen poistosta huolehtii toinen korotronilanka (detac corona wire). Jos sähkövaraus ei poisteta paperista, on mahdollista, että paperi kiertyy kuvarummun ympärille sen sijaan, että jatkaisi matkaansa kohti *kiinnityslaitetta* (fuser).



Kuva 2: Paperin kulkurata lasertulostimessa [Harris, 2007].

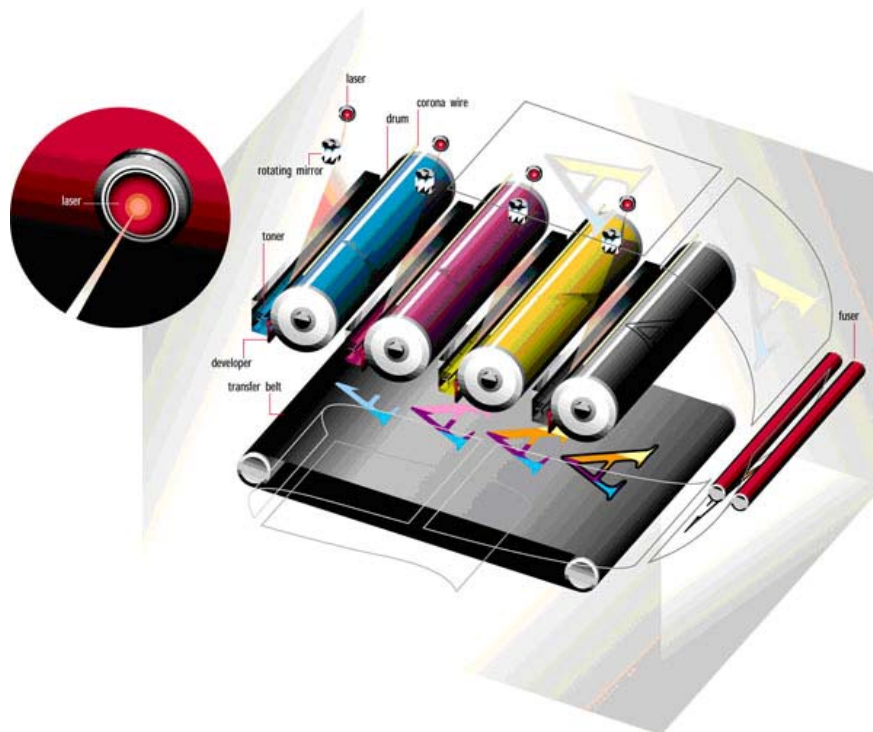
Kiinnittämisvaiheen tarkoituksena on saada paperille siirretty kuivamuste kiinnitettyä paperin pintaan pysyvästi. Toimenpide tapahtuu kiinnityslaitteen avulla. Kiinnityslaite koostuu kahdesta toisiaan vastakkain olevasta rullasta: *painerullasta* (pressure roller) ja *lämpörullasta* (heat roller). Lämpörullan lämpötila on yli 200 astetta. Paperi kulkee näiden rullien välistä ja puristuksen sekä lämmön yhteysvaikutuksesta kuiva väri kiinnittyy paperihuokosiin.

Kiinnityksen jälkeen paperi poistuu paperiohjainten avulla ulos laitteesta. Juuri kiinnityslaitteen kuumuudesta johtuen xerografiaan perustuvien kopiokoneiden ja tulostinten ulos työntämät paperit ovat lämpimiä. Samaan aikaan kone suorittaa puhdistuksen ja varauksen poiston rummulle. Puhdistus tapahtuu kaapimen avulla, joka kerää kuvarummun pintaan jääneen ylimääräisen väriaineen pois ja kokoaa sen *hukkavärisäiliöön* (waste toner container).

Lasertulostimissa on useampia tapoja tulostaa värillisiä sivuja. Värilaserlaitteissa on mustan värin lisäksi, magenta, syaani ja keltainen väri. Huokeammat laitteet ovat niin sanottuja *nelivaihetulostimia* (four-pass-printers). Nelivaihelaitteissa on neljä kappaletta väriainesäiliöitä ja kehityksiköitä pyörivässä rullakossa [Harris, 2007]. Ensin yksi väreistä sekoittuu kehitteen kanssa ja kehityksikkö siirtää värin kuvarummun pinnalle. Sen jälkeen väri siirretään paperille aiemmin kuvatulla tavalla. Tämän jälkeen rullakko pyöräyttää seuraavan värin ja kehitteen kohti kuvarummun pintaa ja siirtoprosessi toistuu tämän värin osalta. Kun kaikki värit ovat paperilla, etenee paperi kiinnityslaitteeseen ja

sen jälkeen ulos koneesta. Osa nelivaihelaitteista siirtää kaikki värit ensin kuvarummun pintaan ja vasta sen jälkeen yhdellä kerralla paperille.

Hintavammat värilaser-laitteet, esimerkiksi niin sanotut monitoimilaitteet, jotka toimivat myös kopiokoneena, skannerina ja tarvittaessa faksina, ovat yksivaihetulostimia (single-pass-printers). Yksivaihelaitteissa on erillinen *tulostusyksikkö* (printer unit) – laseryksikkö, rumpu ja kuivaväriaine – jokaiselle värille [Harris, 2007]. Värit siirtyvät kuvarummuilta *siirtohihnalle* (transfer belt), josta nelivärinen kuva siirtyy edelleen paperille (ks. kuva 3).



Kuva 3: yksivaihe-värilaser-tulostimen tulostusyksiköt [Xerox-b].

3.2. LED-tulostimen toimintaperiaate

LED-tulostamisen toimintaperiaate noudattaa samaa kaavaa lasertulostamisen kanssa. Tästä syystä LED-tekniikkaan perustuvia tulostimia mainostetaankin joskus virheellisesti lasertulostimina. Myös värien tulostaminen LED-laitteissa tapahtuu joko nelivaihe- tai yksivaihetekniikan avulla. Ainoa eroavaisuus tekniikoiden välillä on altistamisvaiheessa, eli prosessissa, jonka avulla elektrostaattinen kuva luodaan kuvarummun pinnalle.

LED-tulostimissa kuvarumpua ammutaan lasersäteen sijaan LED-valoilla (light emitting diode). Rummun yli kulkee tiivis rivistö LED-lamppuja, jotka syttyvät ja sammuvat yhtenä rivistönä yhdenaikaisesti piirtäen pyörivälle rummulle kuvaa pikselirivi kerrallaan. Tekniikka eroaa laseryksikön toimin-

nassa juuri tässä mielessä; laseryksikkö ampuu pisteet rummun pinnalle yksi kerrallaan. [Oki, 2005]

LED-tulostinten rummun altistus toimii siis huomattavasti yksinkertaisemmin ja tehokkaammin kuin lasertulostinten, joissa valo kohdistuu rumpuun pyörivien peilien avulla [Oki, 2005]. LED-tulostinten *tulostuspäissä* (LED Print-head) tällaisia liikkuvia ei tarvita. Toisaalta LED-tulostinten tulostustarkkuus on riippuvainen tulostuspään ledien määrästä. Useimmiten tulostuslaatu onkin LED-laitteissa heikompi kuin lasertulostimissa.

3.3. Mustesuihkutulostimen toimintaperiaate

Mustesuihkutulostinten toimintaperiaate poikkeaa selvästi laser- ja LED-tulostimista. Mustesuihkutulostimeksi sanotaan mitä tahansa laitetta, joka asettaa erittäin pieniä väripisaroita (läpimitaltaan noin 10–60 mikronia) paperille kuvan aikaansaamiseksi. Tyypillinen mustesuihkutulostin koostuu *tulostuspäyksiköstä* (print head assembly), *paperinsyöttöyksiköstä* (paper feed assembly), *virtalähteestä* (power supply), *ohjainpiireistä* (control circuitry) ja *käyttöliittymäportteista* (interface ports). [Tyson] Seuraavaksi keskitytään kuitenkin vain kahteen ensin mainittuun tulostimen osaan.

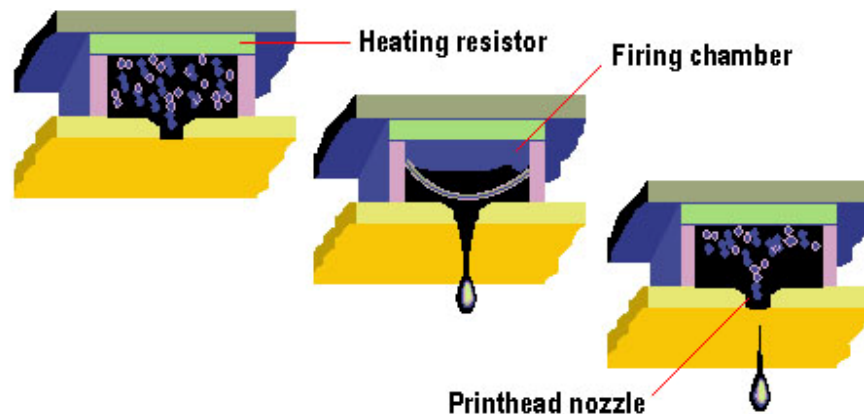
Tulostuspäyksikön osia ovat *tulostuspää* (print head), *mustekasetit* (ink cartridges), *tulostuspään liikemoottori* (print head stepper motor), *hihna* (belt) ja *vakaintanko* (stabilizer bar). Tulostuspää on mustesuihkutulostimen ydin, joka sisältää sarjan suuttimia, joita käytetään väripisaroiden suihkuttamiseen paperille. Valmistajasta riippuen mustekasetit tulevat joko jokaiselle värille erikseen (musta, syaani, magenta, keltainen), värit erillisenä kasettina (ns. *kolmivärikasetti* – tri-color cartridge) tai kaikki värit ja musta yhdessä ja samassa kasetissa. Hintavimmissa, valokuvatulostukseen erikoistuneissa laitteissa värikasetteja voi olla jopa kahdeksan erilaista. Joihinkin mustekasettiyksiköihin sisältyy myös itse tulostuspää. Tulostuspään liikemoottorin tehtävänä on liikuttaa tulostuspäyksikköä edestakaisin paperin yllä. [Tyson] Tulostuksen vauhdittamiseksi mustesuihkutulostimet tulostavat useamman vertikaalirivistön pikseleitä yhdellä pyyhkäisyllä [PCTechGuide, 2004]. Hihna kiinnittää tulostuspään ohjainmoottoriin ja vakaintanko varmistaa, että tulostuspään liike paperin yllä on tarkkaa ja kontrolloitua [Tyson].

Paperinsyöttöyksikkö koostuu paperialustasta tai paperinsyöttötelineestä (paper tray/feeder), syöttörullista (rollers) ja paperinsyötön liikemoottorista (paper feed stepper motor). Paperialustalle mahtuu yleensä noin puoli riisiä paperia, pienemmissä ja edullisimmissä laitteissa on ainoastaan syöttöteline, johon arkkeja mahtuu huomattavasti vähemmän. Syöttörullat irrottavat paperin alustalta tai telineestä ja kuljettavat paperia eteenpäin laitteen ollessa kirjoitus-valmiina. Paperinsyötön liikemoottori varmistaa ja mahdollistaa, että pape-

ri etenee kohti tulostuspäätä siten, että kuvan katkeamaton tulostus mahdollistuu. [Tyson]

Eri valmistajien mustesuihkutulostimet muodostavat pienenpienet mustepisaransa eri tavoin. Tällä hetkellä on kaksi vallitsevaa tekniikkaa, joita mustesuihkukirjoittimia valmistavat yritykset hyödyntävät tähän tarkoitukseen: *thermal bubble* -tekniikka sekä *piezoelectric*-tekniikka [Tyson]. Näistä tekniikoista käytetään myös nimitystä "pisara vaadittaessa" eli "drop on demand" (DOD). Tyypillisen mustesuihkutulostimen tulostuspään suutin tekee päätöksen pudottaa väriä tai olla pudottamatta noin 5000 kertaa sekunnissa [PCTech-Guide, 2004].

Thermal bubble -tekniikka on käytössä muun muassa Canonin ja HP:n valmistamissa laitteissa. Tekniikasta käytetään myös nimitystä *bubble jet*. *Thermal inkjet* -laitteessa pienet resistorit luovat lämpöä ja lämpö höyrystää musteen ja synnyttää kuplan. Kuplan kasvaessa pieni osa mustetta työntyy ulos suuttimesta paperille. Kun kupla puhkeaa, syntyy tyhjiö, joka työntää lisää mustetta mustesäiliöstä tulostuspäähän (ks. kuva 4). Huokeassa bubble jet -tulostuspäässä on noin 300–600 pientä suutinta, jotka kaikki voivat ampuu mustepisaran samanaikaisesti. [Tyson] Uusimmissa pientoimistokäyttöön soveltuvissa laitteissa mustavalkotulostustarkkuus on 1200 x 600 dpi. Laadukkaalle paperille tulostetun värikuvan tarkkuus voi olla jopa 4800 x 1200 dpi [HP, 2007].



Kuva 4: Thermal bubble -tekniikan toimintaperiaate [Tyson].

Epsonin patentoima piezoelectric-tekniikka hyödyntää tulostuspäissään piezokristalleja. Kristalli sijaitsee tulostuspääsuuttimen takaosassa. Johdettaessa pieni sähkövirta kristalliin, se alkaa värähdellä. Kristallin värähdessä kohti suutinta se pakottaa pienen määrän mustetta ulos suuttimesta paperille. Ulospäin värähdessä se vetää mustetta säiliöstä tulostuspäähän ulos purskahtaneen musteen verran. [Tyson]

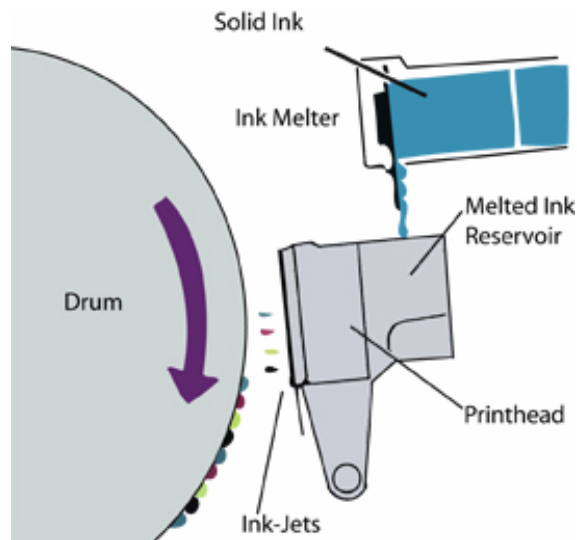
Näistä kahdesta tekniikasta Epsonin piezoelectric-tekniikalla on enemmän hyviä puolia. Tekniikka mahdollistaa paremman kontrollin paperille tippuvan

pisaran koon ja muodon määrittelyssä. Pesarat ovat pienempiä kuin thermal bubble -tekniikassa ja siten mahdollistavat useampien suuttimien sijoittamisen tulostuspäähän. Piezoelectric-tekniikassa mustetta ei myöskään tarvitse lämmittää ja viilentää jokaisen tulostuskerroksen jälkeen. Tämä säästää aikaa ja mahdollistaa uusia musteita kehitettäessä keskittymisen musteen imeytymiskykyyn lämmönsietokyvyn sijaan. Piezoelectric-laitteilla pystytään saavuttamaan suurempia tulostustarkkuuksia kuin thermal bubble -laitteilla. [PCTech-Guide, 2004]

3.4. Väriahatulostimen toimintaperiaate

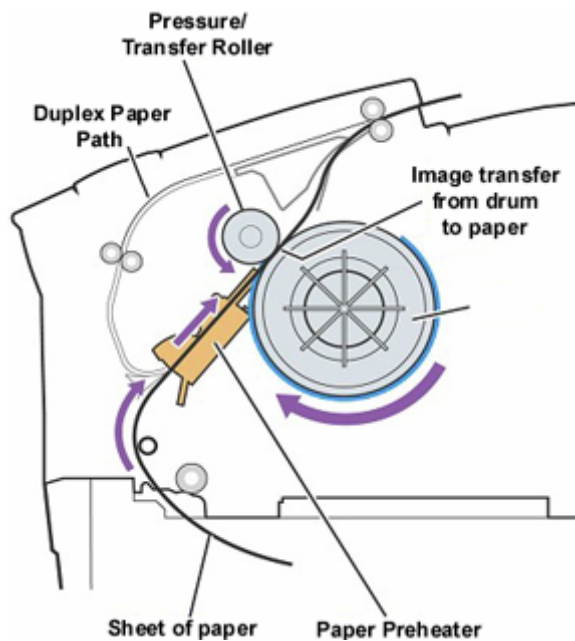
Viimeisin uutuus väritulostusteknologiassa on aikanaan Tektronix-yhtiön kehittämä väriahatulostaminen (solid ink printing). Vuodesta 2000 lähtien oikeudet väriahatekniikkaan on omistanut Xerox, ja suurin osa markkinoilla olevista väriahalaitteistakin on siten Xeroxin valmistuttamia.

Ennen tulostamisen aloittamista väriahatulostimen täytyy sulattaa kiinteässä muodossa olevaa mustetta *mustesäiliöön* (melted ink reservoir) [PCTech-Guide, 2004b]. Käytännössä tämä tapahtuu vain konetta käynnistettäessä. Koneen ollessa päällä laite säilyttää värisäiliössä olevat musteet nestemäisessä muodossa. Lasertulostinten tapaan myös väriahatulostimen ydinosa on *tulostusrumpu* (drum). Se on alumiininen, anodisoitu ja sylinterimäinen. Väriahatulostamisen ensimmäisessä varsinaisessa vaiheessa *ylläpitorulla* (maintenance roll) levittää ohuen kerroksen silikoniöljyä tulostusrummun pintaan, jotta myöhemmin tapahtuva värin siirto sujuu luotettavasti. Seuraavaksi koko rummunlevyinen, tasaisesti 135 celsiusasteen lämpöinen *tulostuspää* (printhead) siirtää kaikki värit kertapyyhkäisyllä pyörivään tulostusrumpuun (ks. kuva 5). Tulostuspää käyttää mustesuihkutulostamisesta tuttua piezoelectric-tekniikkaa. Rummun lämpötila on 65 celsiusastetta, joten nestemäinen muste jähmettyy välittömästi hieman kiinnittyessään rumpuun. [Xerox, 2007d; Jaeger]



Kuva 5: Musteen sulatus ja siirto tulostusrummun pintaan [Jaeger].

Kun kuva on muodostunut rummulle kokonaisuudessaan, se siirretään paperille. Paperialustalta tuleva paperi kulkee *paperin esilämmittimen* läpi (paper preheater). Sen jälkeen paperi kuljetetaan rummun ja siirtorullan (transfer roller) läpi. Lämmön ja paineen vaikutuksesta hieman jähmettynyt muste siirtyy rummun pinnalta paperille (ks. kuva 6). Saman kierroksen aikana, kun kuva siirtyy paperille laite myös puhdistaa ylimääräisen värin rummun pinnasta ja kokoaa sen hukkavärisäiliöön. [Jaeger] Paperin tullessa ulos koneesta on muste jo täysin jähmettynyt kiinni paperiin ja tuloste on heti valmis käsiteltäväksi.



Kuva 6: Kuvan siirto rummulta paperille [Jaeger].

4. Tulostuslaitteiden vertailua

Edellisessä luvussa esiteltyjen tulostustekniikoiden jälkeen paneudutaan eri tekniikoihin perustuvien laitteiden vahvuuksiin ja heikkouksiin. Laitetta hankittaessa on tärkeää miettiä, mikä on laitteen ensisijainen käyttötarkoitus. Teksti-, grafiikka ja valokuvatulostusominaisuuksien lisäksi on syytä ottaa huomioon myös laitteessa käytettävien kulutustarvikkeiden määrä ja niistä johtuvat ylläpitokustannukset.

4.1. Lasertulostimet

Lasertulostimet soveltuvat edelleen erityisesti tekstin ja grafiikan tulostamiseen. Uusillakin lasertulostimilla tulostetut valokuvat jäävät usein saman hintaluokan mustesuihkulaitteilla tulostettuja vaatimattommiksi, mutta eivät yhtä paljon kuin aiemmin.

Tulostustarkkuus nykyisissä toimistokäyttöön tarkoitetuissa lasereissa vaihtelee 1200 x 600 – 2400 x 1200 dpi:n välillä riippuen tulostusasetuksista ja käytetystä tulostuspaperista [Xerox, 2007a]. Lasertulostin on siis mediariippuvainen; parasta tulostuslaatua voidaan hyödyntää ainoastaan laadukkaampia (ja kalliimpia) papereita käytettäessä. Tulostusnopeus esimerkiksi Phaser 6300 -laitteessa on 25 värisivua ja 35 mustavalkosivua minuutissa [Xerox, 2007a].

Lasertulosteiden suuri etu verrattuna mustesuihkutulosteisiin on halvempi hinta sivua kohti, riittävä väripatruunat ja kestävä tulosteet. Paperiin ”grillattu” kuivamuste ei tahriinnu, kuten voi käydä juuri koneesta ulostulleelle mustesuihkutulosteelle. [Ylä-Jääski, 2007]

Lasertulostimissa kulutustarvikkeita eli aika ajoin vaihtoa vaativia osia ovat värien lisäksi ainakin kuvarumpu, kiinnityslaite ja hukkavärisäiliö. Nämä osat aiheuttavat lisäkäyttökustannusten lisäksi kuormitusta ympäristölle, vaikkakin monella laitevalmistajalla on nykyisin jonkinlaiset palvelut kulutustarvikkeiden kierrätykselle. Moni lasertulostin on kuitenkin Energy Star -hyväksytty, eli sähkönkulutusta on pyritty minimoimaan. Esimerkiksi Xeroxin Phaser 6300 -väri lasertulostin kuluttaa tulostuksessa 580 wattia, valmiustilassa 120 W ja virransäästötilassa 45 W [Xerox, 2007a].

4.2. LED-tulostimet

LED-tulostintekniikan ollessa samankaltainen lasertulostimen kanssa, pätevät lasertulostinten hyvät ja huonot puolet myös LED-tulostimiin. Ainoa huomionarvoinen seikka LED-tulostimissa liittyy tulostustarkkuuteen, joka useinkaan ei ole samaa luokkaa kuin vastaavissa lasertulostimissa. Esimerkiksi Xerox Phaser 7400 -laitteen perustulostustarkkuus on 600 x 600 dpi [Xerox, 2007b]. Monesti LED-tekniikkaan perustuvat laitteet ovat kuitenkin tehokas tapa laitevalmista-

jalle tutustuttaa ostaja väritulostamisen maailmaan melko huokeahintaisella, mutta kuitenkin tyydyttäviä väritulosteita tuottavalla laitteella.

4.3. Mustesuihkutulostimet

Mustesuihkutulostinta on pidetty ylivoimaisimpana vaihtoehtona koti- ja pien-toimistojen väritulostimeksi aina vuodesta 1987, HP PaintJetin ilmestymisen jälkeen. Nykyisin tilanne on merkittävästi tasoittunut, mutta valokuvatulostamiseen hyvä mustesuihkutulostin on vielä usein paras vaihtoehto.

Mustesuihkukirjoittimiin kohdistuva kritiikki liittyy usein korkeaan sivukohtaiseen hintaan. Värisäiliöt ovat pieniä mutta kalliita, ja siten värisivujen tulostaminen tulee erittäin kalliiksi. Yhden A4-kokoisen värisivun hinta saattaa olla jopa 2 euroa [Ylä-Jääski, 2007]. Tulostuspäitä vaivaa myös tukkeutuminen, kun nestemäisen musteen rippeet kuivuvat tulostuspään suuttimeen [Ylä-Jääski, 2007]. Tätä ongelmaa on saatu helpotettua kehittämällä parempia musteita, tulostuspäiden puhdistusmekanismeja sekä liittämällä tulostuspää osaksi vaihdettavaa mustesäiliötä [Tyson]. Pienien värisäiliöiden aikakin on osittain ohi ainakin toimistokäyttöön tarkoitetuissa hintavimmissa laitteissa: joidenkin säiliöiden arvioidaan riittävän jopa 6000 sivun tulostusmäärään. Myös tulosteiden säilymisaika on uusien musteiden myötä kasvanut moninkertaiseksi.

Monesti mustesuihkutulostinten ilmoitetut minuuttikohtaiset tulostusmäärät eivät ole verrannollisia lasertulostinten vastaaviin ilmoituksiin. Esimerkiksi HP Business Inkjet 2800 -laitteen väritulostusnopeus on 21 värisivua vedoslaadulla, mutta ainoastaan 4 sivua parhaalla laadulla! Lisäksi mustesuihkutulostimet ovat vielä lasertulostimiakin enemmän mediariippuvaisia. Koska muste imeytyy paperin kuituihin, on paperin laadulla erittäin suuri merkitys tulosteen lopputuloksessa.

Kulutustarvikkeita mustesuihkutulostimissa ovat värisäiliöt ja tulostuspäät, usein yhdessä ja samassa paketissa. Mustesuihkutulostimet ovat myös virrankulutuksen suhteen ympäristöystävällinen vaihtoehto: virrankulutus vaihtelee 1 ja 80 watin välillä. Suurinta virrankulutus on paperinsyöttömekanismin ollessa päällä. [Bailey, 2006]

4.4. Väriahatutulostimet

Väriahatutulostinten etu muihin esiteltyihin laitteisiin on ehdottomasti riippumattomuus tulostusmediasta. Halvimmallekin mahdolliselle paperille tulostettaessa väriahatulostin tuottaa kirkkaat värit ja kuultavan, vahapintaisen kuvan. Koska muste on jo osittain kohmettunut paperille siirtyessään, ei väri imeydy paperin kuituihin, vaan jää erilliseksi kerrokseksi paperin pinnalle. Lopputulos poikkeaa selvästi laser-, LED- ja mustesuihkutekniikkaan perustuvien kirjoitinten tulosteista.

Aiemmin väriahatulostinten tulostustarkkuus ei yltänyt aivan samoihin lukemiin kuin kilpailevat teknologiat, mutta nykyisin tilanne on tasoittunut. Myöskään tulostuskustannukset eivät enää ole merkittävästi lasertulosteita korkeammat. Väriahatulostaminen kuitenkin profiloituu edelleen enemmän kuvien ja grafiikan tulostamiseen [White, 1999]. Jos suurin osa tehdyistä tulosteista on vähävärisiä tekstitulosteita, on laser parempi vaihtoehto.

Ympäristöystävällisyyden suhteen väriahatulostimet ovat ristiriitaisia laitteita. Vahavärien asentaminen laitteeseen on helppoa, eikä kiinteässä muodossa oleva muste tahraa käyttäjää sitä koneeseen asentaessa. Koska mustepalikoita ei tarvitse pitää erillisessä säiliössä, tuottaa laite jätettä ainoastaan hukkaväri-musteen verran, eli huomattavasti vähemmän kuin laser-, LED-, tai mustesuihkukirjoittimet. Toisaalta laitteen virrankulutus on omaa luokkaansa: keskimääräinen kulutus 230 W ja maksimikulutus 1500 W [Xerox, 2007c]. Suuri virrankulutus selittyy sillä, että laitteen täytyy pitää mustesäiliössä olevat musteet lämpiminä, eli nestemäisessä muodossa. Tästä syystä laitteen tulee olla päällä ympäri vuorokauden. Jos laite sammutetaan, jähmettyy mustesäiliöissä oleva muste. Uudelleenkäynnistyksen yhteydessä laite joutuu tyhjentämään mustesäiliöt ja sulattamaan uutta mustetta mustesäiliöihin kiinteässä muodossa olevista väripalikoista.

Monet väriahalaitteet ovat kuitenkin Energy Star -hyväksytyjä, eli laite siirtyy vähemmän kuluttavaan virransäästötilaan tarvittaessa. Virransäästöttilasta herätetyn laitteen lämpenemisaika on noin 2 minuuttia, joka saattaa tuntua joskus pitkältäkin ajalta. Tästä syystä laitteessa on tunnistin, joka tarkkailee vuorokaudessa tapahtuvaa tulostamista. Jos esimerkiksi laitteella aloitetaan tulostaminen kahdeksalta aamulla, laite osaa nostaa itsensä ylös virransäästöttilasta jo ennen kahdeksaa.

5. Muut ostopäätökseen vaikuttavat tekijät

Tässä tutkielmassa on keskitytty erottelamaan laitteita tulostusteknisten seikkojen sekä tulostuslaadun suhteen. Usein lopullinen valinta toimiston väritulostinta hankkiessa saatetaan tehdä aivan muiden seikkojen pohjalta.

Mikäli kyseessä on toimisto, jonka tietoverkossa on käytössä useita käyttöjärjestelmäkokoontia ja laaja määrä erilaisia sovellusohjelmia, saatetaan tulostusteknisesti sopivimman koneen sijasta päätyä laitteeseen, jossa on jo ostohetkellä tulostustuki kaikkien järjestelmien kanssa. Yhteensopivuusongelmia esiintyy etenkin uusien laitteiden kohdalla, kun joidenkin käyttöjärjestelmien ajurit ovat vielä puutteellisia.

Monesti jo pieniä toimistokäyttöön soveltuvia verkkotulostimia myydään ainoastaan huoltosopimusten kanssa. Tällöin asiakas maksaa vuosittain tietyn

summan jälleenmyyjälle, joka toimittaa asiakkaan laitteeseen värit ja muut kulutustarvikkeet tarvittaessa, ja veloittaa tulostetut sivut sovitun kappalehinnan mukaisesti. Ostajan mieltymys tiettyyn tulostusteknologiaan saattaa muuttua, mikäli johonkin toiseen tekniikkaan perustuvan laitteen jälleenmyyjä antaa huomattavasti edullisemman huoltosopimustarjouksen kauppaamalleen laitteelle. Myös merkkiuskollisuus ja siitä koituva hyöty laitteita huollattaessa saattaa olla painava syy valintaa tehtäessä.

Electric Document Systems -säätiön tekemän tutkimuksen mukaan tulostusmäärät ovat 1–2 prosentin kasvussa aina vuoteen 2020 asti. Kasvua on kuitenkin lähinnä pakkausmateriaalien ja mainosten tulostamisessa. Toimistotulostamisen oletetaan sen sijaan hitaasti laskevan. Samainen tutkimus arvioi, että tulostetun ja digitaalisen informaation suhde vuonna 2020 on 65-35 digitaalisen hyväksi [Evans, 2001]. Hieman käytännönläheisemmän esimerkin paperinkulutuksen vähenemisestä antaa Tampereen yliopiston Ekokampus-projektin ylläpitämä kirjanpito Tampereen yliopiston paperinkulutuksesta. Tulokset osoittavat käyttäjäkohtaisen paperinkulutuksen olleen laskussa jo useita vuosia. Erittäin suurta kulutuksen väheneminen on ollut viimeisimpinä vuosina [Raatikainen, 2007].

Jos siis toimiston paperinkulutusta aiotaan vähentää tai se on jo vähentynyt, saattaa perinteinen verkkoväritulostin olla riittämätön laite. Tällöin sopiva laite olisi monitoimilaite, joka kopioinnin ja tulostamisen lisäksi kykenee verkko- ja sähköpostiskannaukseen. Nämä ns. Scan-to-file- ja Scan-to-Email -ominaisuudet ovat erittäin tehokas tapa vähentää paperinkulutusta toimistoissa.

6. Yhteenveto

Tulostuslaitteiden kehitys 70-luvun loppupuolelta tähän päivään ei ole ollut yhtä nopeaa kuin monilla muilla tietotekniikan osa-alueilla. Siitä huolimatta nykyiset kirjoittimet ovat huomattavasti tehokkaampia ja tuottavat laadukkaampia tulosteita kuin vaikkapa muutaman vuoden takaiset edeltäjänsä.

Ei voida kuitenkaan yksiselitteisesti sanoa, mikä esitellyistä tulostustekniikoista on paras vaihtoehto nykyisiin toimistoihin. Päätökseen vaikuttavat useat seikat, kuten tulostusmäärä, ja se millaisia tulosteita laitteella pääsääntöisesti tulostetaan.

Laser- ja LED-tulostimet ovat hyviä valintoja, jos tulostusmäärät ovat suurempia. LED-laite tarjoaa huokean vaihtoehdon tuottaa väritulosteita, mutta laser-tulostimilla saavutetaan usein parempaa laatua.

Vähäisiin tulostusmääriin mustesuihkulaite on hyvä valinta. Laitteiden valokuvatulostus on ensiluokkaista ja myös tekstitulostus kelvollista. Tulosteissa

on kuitenkin suttautumis- ja liukenemisvaara varomattomasti tai kosteissa oloissa käsiteltäessä.

Väriahatekniikkaan perustuvat laitteet ovat kasvattaneet markkinaosuuttaan viimeaikoina. Laitteen mediasta riippumattomat, näyttävät tulosteet ovat jotakin, mitä muut markkinoilla olevat laitteet eivät kykene tarjoamaan. Mikäli tulosteita käytetään paljon yrityksen markkinointiin tai muuten yrityksen esittelyyn, väriahatulostin on hyvä vaihtoehto. Toimiston sisäiseen informaation jakamiseen riittää toisaalta vähemmänkin kiiltävä värituloste.

Laitehankintapäätökset eivät kuitenkaan ole vain tulostuslaadusta riippuvaisia. Laitteen soveltuminen eri käyttöjärjestelmiin ja ohjelmistoihin voi olla erittäinkin kriittinen seikka monessa toimistossa. Myös erityisen sopiva laitetarjous jonkin tietyn laitteen jälleenmyyjältä saattaa vähentää ostajan vaatimuksia tulostuslaadun suhteen. Toisaalta joissakin toimistoissa paperitulostaminen on muuttunut niin satunnaiseksi, että paperimedian digitalisoinnin mahdollistava monitoimilaite on sopivin vaihtoehto uudeksi hankinnaksi.

Viiteluettelo

- [Bailey, 2006] Douglas Bailey, Increasing Efficiency in Appliances, Office Equipment and LED Lighting. Available at: http://mail.mtprog.com/CD_Layout/Day_3_23.06.06/1115-1300/ID90_Bailey_final.pdf. Checked 10.4. 2007.
- [Bellis, 2006] Mary Bellis, History of computer printers. Available at http://inventors.about.com/library/inventors/blcomputer_printers.htm. Checked 8.2. 2007.
- [Evans, 2001] Patricia Evans, A bright future for printing - at least until 2020. *The Seybold Report* **1**, 7 (2001), 28
- [Harris, 2007] Tom Harris, How Laser Printers Work. Available at <http://computer.howstuffworks.com/laser-printer.htm>. Checked 21.3.2007.
- [HP, 2005] Hewlett-Packard, Milestones in inkjet technology. Available at http://www.hp.com/oeminkjet/about_TII/history/milestones.html. Checked 8.2.2007.
- [HP, 2007] HP, Hewlett Packard, Business Inkjet 2800 Printer series – specifications and warranty. Available at <http://h10010.www1.hp.com/wwpc/us/en/sm/WF06a/18972-18972-236251-24728-3328074-429041.html>. Checked 4.4. 2007.
- [Jaeger] C. Wayne Jaeger, Color Solid Ink Printing. Available at http://www.imaging.org/resources/web_tutorials/solid_ink.cfm. Checked 4.4. 2007.
- [Loutfy et al, 1988] Rafik O. Loutfy, Ah-Mee Hor, Cheng-Kuo Hsiao, Giuseppa Baranyi and Peter Kazmaier; Organic photoconductive materials. *Pure & Appl. Chem.* **60**, 7 (1988), 1047-1054.

- [Oki, 2005] OKI Data Americas, Digital LED Color Printing. Available at http://www.okidata.com/mkt/downloads/LED_descrip.pdf. Checked 8.2.2007.
- [PCTechGuide, 2004a] PCTechGuide, Guides/Input & Output/Inkjet Printers. Available at <http://www.pctechguide.com/53Inkjets.htm>. Checked 2.4.2007.
- [PCTechGuide, 2004b] PCTechGuide, Guides/Input & Output/Other Printers. Available at http://www.pctechguide.com/54OtherPrinters_Solid_ink.htm. Checked 3.4.2007.
- [Raatikainen, 2007] Sanna Raatikainen, henkilökohtainen tiedonanto, tammikuu 2007.
- [Tyson] Jeff Tyson, How inkjet printers work. Available at <http://www.howstuffworks.com/inkjet-printer.htm>. Checked 8.2.2007.
- [White, 1999] Ron White, Solid ink vs. color laser printers. *PC Computing*, **12**, 9 (1999), 214.
- [Wikipedia, 2007] Wikipedia, The Free Encyclopedia, Solid ink. Available at http://en.wikipedia.org/wiki/Solid_ink. Checked 8.2.2007.
- [Xerox, 2007a] Xerox Corporation, Xerox 6300/6350 lisätietoja. Saatavilla <http://www.office.xerox.com/color-printers/phaser-6300-6350/spec-fifi.html>. Tarkastettu 8.2.2007
- [Xerox, 2007b] Xerox Corporation, Xerox 7400 lisätietoja. Saatavilla <http://www.office.xerox.com/color-printers/phaser-7400/spec-fifi.html>. Tarkastettu 8.2.2007
- [Xerox, 2007c] Xerox Corporation, Xerox 8500/8550 lisätietoja. Saatavilla <http://www.office.xerox.com/color-printers/phaser-8500-8550/spec-fifi.html>. Tarkastettu 8.2.2007
- [Xerox, 2007d] Xerox Corporation, Miten väriahatekniikka toimii. Saatavilla http://www.office.xerox.com/perl-bin/opb_productpopup.pl?dir=solidink&object=solidinkpage_popup_01. Tarkastettu 8.2.2007.
- [Xerox-a] Xerox Corporation, The Story of Xerography. Available at <http://a1851.g.akamaitech.net/f/1851/2996/24h/cacheB.xerox.com/downloads/usa/en/s/Storyofxerography.pdf>. Checked 8.2.2007.
- [Xerox-b] Xerox, Illustration: Single-pass laser printer. Available at http://www.xerox.com/images/usa/en/n/nr_illust_single_pass_laser_printer.jpg. Checked 11.4.2007
- [Ylä-Jääski, 2007] Vesa Ylä-Jääski, Väri lasereita koteihin ja pientoimistoihin. *Tietokone*, **4** (2007), 66-68.

Avoin lähdekoodi ja lisenssit

Riikka Valtanen

Tiivistelmä.

Avoimen lähdekoodin ohjelmistot ovat selkeästi saavuttaneet ja joissakin tapauksissa jopa ohittaneet laadultaan perinteisen ohjelmistoteollisuuden muodot. Joka tapauksessa avoimen lähdekoodin ohjelmistot ovat erittäin kilpailukykyisiä verrattuna perinteisen ohjelmistotuotannon ohjelmistoihin. Avoimen lähdekoodin ohjelmistojen lisensointi tuo mukanaan kuitenkin riskejä, joita voidaan ehkäistä tutustumalla lisensseihin huolellisesti. Tekijänoikeuksien tuntemus on nykyään yhä tärkeämpää, koska suuri osa taloudellisista voitoista saadaan nykyään tekijänoikeuden alaisilla tuotteilla ja palveluilla. Sisällöntuotantoteollisuus kasvaa nopeasti, ja kaikki sen käyttämä ja tuottama materiaali on tekijänoikeuden alaista.

Tämä tutkimus tarkastelee avoimen lähdekoodin eri lisenssejä teoriassa ja käytännön esimerkkien avulla ja pyrkii arvioimaan niiden hyödyllisyyttä ja haitallisuutta keksijän, kilpailijan ja yhteiskunnan näkökulmista. Tutkimuksessa vertailen myös avoimen lähdekoodin lisenssejä verrattuna perinteisempiin tapoihin kuten tekijänoikeuksia ja patentointia. Avoimen lähdekoodin merkitys kasvaa jatkuvasti ja yritykset hyödyntävät avoimen lähdekoodin ohjelmiston ohjelmia kasvavassa määrin.

Ohjelmistojen suojaamistapoja on useita erilaisia. Eri suojaamismuodoissa on omat hyvät puolensa. Tärkeää on kuitenkin yrittää välttää turhia riskejä ja ymmärtää eri suojausmuotojen erityispiirteet. Tutkielmassa kiinnitetään erityistä huomiota näihin ominaispiirteisiin ja tutkielman tarkoitus on auttaa löytämään oikea suojausmuoto erityisesti avoimen lähdekoodin ohjelmille.

Avainsanat ja -sanonnat: Avoin lähdekoodi, Open Source, lisenssi, vapaa ohjelmisto

CR-luokat: K.5.1

1. Johdanto

Tutkielman aiheena ovat avoin lähdekoodi, sen synty ja kehitys sekä tämän päivän lisenssit, patentit ja tekijänoikeudet. Avoimen lähdekoodin kehityksen lähtökohtia ovat olleet vapaus, luovuus ja yhteisöllisyys. Voidaan myös sanoa,

että avoimen lähdekoodin lisenssit antavat sen saajalle oikeuksia, joita ohjelmistoyritykset ovat perinteisesti pidättäneet itsellään. Tietokoneen käyttäjät ovat usein tietämättään tekemisissä avoimen lähdekoodin ohjelmien kanssa, joista yleisimpiä ovat Linux, Apache ja OpenOffice. Nykyään Open Source -tuotteet muodostavat merkittävimmän haastajan Microsoftin markkinajohtajuudelle.

Tutkielmassa luodaan yleiskatsaus tekijänoikeuteen ja sen historiaan sekä tutkitaan tarkemmin myös erilaisia lisenssejä ja niiden eroja. Tavoitteena on selvittää avoimen ja suljetun lähdekoodin erot sekä hyödyt että haitat. Ennen kaikkea tutkimuksessa tutkitaan tekijänoikeuksien, patenttien ja erilaisten lisenssien sopivuutta ohjelmistojen suojaamiseen. Tutkimuksen ulkopuolelle jätetään avoimen lähdekoodin kehittäjänäkökulma.

Tutkimus perustuu pääosin kirjallisuuteen. Suomenkielistä materiaalia aiheesta löytyy edelleen vähän. Lähteiden valinnan ongelmaksi nousi eri maiden lainsäädäntöjen eroavaisuudet, mistä johtuen lähteitä ei ole voitu käyttää suoranaisesti kuvaamaan Suomen tilannetta. Internetistä löytyi paljon mielipiteitä ja tulkintoja aiheesta, mutta tietojen oikeellisuus ei aina ollut luotettavaa, vaikkakin antoi uusia näkökulmia tutkimusta tehdessä.

Maaliskuussa 2007 järjestettiin IT-alan naisverkoston, Nice Tuesdayn, pitämä tilaisuus, jossa keskusteltiin asianajajien Kati Tusan, Kaisa Keski-Vähälän ja Jesper Nevalaisen johdolla tekijänoikeuksista ja lisensseistä. (Jäljempänä Nice Tuesday -tilaisuus, 2007) Tutkimuksessa käyttämäni konkreettiset esimerkit tulivat esiin kyseisessä tilaisuudessa. Viime vuosina myös yritykset ovat alkaneet hyödyntämään avointa lähdekoodia omissa liiketoiminnoissaan, ja Nice Tuesday -tilaisuudessa käsiteltiinkin sitä, kuinka teknologian nopea kehitys ja liiketoiminnan kansainvälistyminen yhdistettynä avoimen lähdekoodin ja lisensioinnin mukanaan tuomiin erilaisiin muutoksiin ovat merkinneet valtavia haasteita myös juridiikan puolella. Muun muassa lähdekoodin lisenssien tuntemus on erityisen tärkeää, sillä hyvä juridinen valmistelu luo edellytykset ohjelmiston menestykselle.

2. Tekijänoikeuksien historia

Tämä luku käsittelee tietokoneohjelmien suhdetta erilaisiin immateriaalioikeuksiin kuten tekijänoikeuteen ja patenteihin. Luvun loppuosassa tuodaan esiin myös avoimen lähdekoodi ohjelmistokehittäjien mielipiteitä patentoimisesta. Luku käsittelee myös tekijänoikeuksien historiaa ja tekijänoikeusteollisuutta.

Nykyisessä tekijänoikeusmielessä varsinainen tekijän kunnioittaminen on melko uutta. Kuitenkin tekijänoikeuden historia ulottuu kauas. Jo antiikin Roomassa runoilija Martialis nimitti ihmisryöstäjiksi muita runoilijoita, jotka esittivät hänen runojaan ominaan. Koska silloin ei ollut käytössä painotekniikkaa, plagiointi oli oikeastaan ainoa tapa varastaa toisen teos. (Haarman 1999, 2) Myöhemmin kirjapainotaidon keksimisen jälkeen tekijöiden suojan tarve kasvoi. Kirjapaino mahdollisti kirjojen, tekstien, runojen yms. massatuotannon. (Nice Tuesday -tilaisuus, 2007).

USA:n perustuslain kirjoittajat totesivat 1700-luvulla, että yhteiskunnan etu vaatii, että taiteilijoilla on yksinoikeus teostensa käyttöön. Ensimmäinen tekijänoikeussäädös Yhdysvalloissa oli Statute of Anne vuonna 1709. Sen piti estää niiden kirjojen myyminen, joiden tekijät eivät olleet antaneet myyntiin lupansa. (Burnett & Marshall 2002, 138–139)

Keskeisin kansainvälinen yleissopimus on alun perin vuonna 1886 tehty Bernin sopimus, joka on kirjallisten ja taiteellisten teosten suojaamisesta tehty sopimus. Bernin yleissopimuksen periaatteena on muiden sopimusvaltioiden teosten samanlainen suoja kuin maan omien kansalaisten teoksilla. Bernin yleissopimusta valvoo Maailman henkisen omaisuuden järjestö (WIPO). Sopimusta on ajantasaisesti ja tarkennettu myöhemmillä sopimuksilla. (Opetusministeriö 2007) Nykyinen Suomen tekijänoikeuslaki on peräisin vuodelta 1961 ja tekijänoikeudesta on tehty useita kansainvälisiä sopimuksia, jotka sitovat Suomessa.

2.1 Tekijänoikeus

Immateriaalioikeus on hyvin kansainvälinen oikeudenala ja sitä ohjaavat useat kansainväliset sopimukset. Tekijänoikeudet kuuluvat immateriaalioikeuksiin, eli aineettomiin oikeuksiin. Immateriaalioikeudet (*intellectual property rights*, IPR) ovat yksinoikeuksia, jotka antavat haltijalleen mahdollisuuden määräaika-na kieltää muita käyttämästä ammattimaisesti hyödykseen suojan kohdetta. Immateriaalioikeudet jaetaan tekijänoikeuteen ja teollisoikeuksiin, joita ovat esimerkiksi patentit, mallit, hyödyllisyysmallit, tavaramerkit ja toiminimet. (Tekes, 2007) Teollisoikeuksista vastaa Suomessa kauppa- ja teollisuusministeriö ja tekijänoikeuksista opetusministeriö. Oikeuksien haltijoiden oikeuksia hallinnoivat ja valvovat tekijänoikeusjärjestöt, joita ovat mm. Kopiosto, Teosto, Kuvasto ja Gramex.

Tekijänoikeus suojaa kirjallisia, kuvaama-, sävellys-, valokuva- ja rakennustaiteen teoksia ja on voimassa EU/ETA -maissa tekijän elinajan ja 70 vuotta hänen kuolinvuotensa päättymisestä. Sitä ei tarvitse erikseen hakea, jos teoskyn-

nys ylittyy. Jos tekijöitä on useita, suoja-aika lasketaan viimeksi kuolleen tekijän kuolinvuodesta. (Lampola, 1996)

Uusi tekijänoikeuslain uudistus astui voimaan 1.1.2006. Kyseessä ei ole tekijänoikeuslain kokonaisuudistus, vaan lähinnä täsmennys siihen, miten vakiintuneet tekijänoikeuden säännökset toteutuvat Internet-ympäristössä, verkko-kaupassa ja digitaalitekniikkaa käytettäessä. (Opetusministeriö, 2005) Lehdistössä on kirjoitettu paljon lain valvomisen ongelmallisuudesta ja laki on saanut osakseen paljon kritiikkiäkin. Tekijänoikeudet ovat muodostuneet Internetin käytön lisääntymisen myötä entistä ongelmallisemmiksi. Tietoa kopioidaan ja julkaistaan jatkuvasti tekijänoikeuksia kunnioittamatta.

Tietokoneohjelman valmistuttua sen tekijän ei tarvitse miettiä miten saa ohjelmalleen tekijänoikeudet. Tekijänoikeuden suojatessa tietokoneohjelmaa hyvä puoli on, että se on maksuton eikä siihen kuulu rekisteröintiä tai muuta vastaavaa toimenpidettä, jolla teokselle saadaan tekijänoikeussuoja. Tekijänoikeus tulee automaattisesti voimaan teoskynnyksen ylittyessä. Tekijänoikeussuoja on myös pitkäaikainen ja suoja-aika jatkuu vielä tekijän kuoleman jälkeen. Tekijänoikeuslaki on myös kansainvälisesti pitkälti harmonisoitu, verrattuna esimerkiksi teollisoikeuksia koskevaan sääntelyyn. (Nice Tuesday –tilaisuus, 2007)

Tekijänoikeus ei kuitenkaan suojaa ideoita. Suoja koskee käytännössä vain koodia eli tekijänoikeuden ulkopuolelle jäävät esimerkiksi ohjelman aihe, ohjelman sisällään pitämät tiedot sellaisenaan, ohjelman toiminnallisuus ja sen kommentointi. Myös työ ja vaivannäkö jäävät tekijänoikeussuojan ulkopuolelle. Tekijänoikeuksia on vaikeaa hallinnoida, koska niitä ei esimerkiksi patenttien tapaan rekisteröidä, eikä siten yksilöidä viranomaistoimin. Käytännössä on myös vaikeaa kontrolloida onko taustalla olevaa koodia kopioitu, koska rekisteröitymistä ei tapahdu. Tästä johtuen on mahdotonta tietää onko samanlainen ohjelmisto jo kehitetty ja ylittääkö oma ohjelmisto teoksen kynnyksen. (Mts.)

Jos tekijöinä on epämääräinen joukko ihmisiä tai kone, kuten esimerkiksi sovelluskehitin, on epäselvää, kenelle tekijänoikeus kuuluu. Jos osuudet eivät muodosta itsenäistä teosta, on tekijöillä yhteisesti tekijänoikeus kokonaisuuteen. Tämä on otettava erityisesti huomioon kun ollaan tekemisissä alihankkijoiden ja konsulttien kanssa, ja alihankkijasuhdetta tai konsultointia koskeva sopimus tulee aina laatia ennen töiden aloittamista, jotta tekijänoikeudet siirtyvät yritykselle eivätkä jää alihankkijalle tai konsultille. (Mts.)

Suomessa tekijänoikeus siirtyy automaattisesti teoksen luoneelle luonnolliselle henkilölle eli tekijälle, mutta ei yritykselle. Tekijänoikeuslaki on erilainen eri maissa; on myös maita, joissa tekijänoikeus siirtyy automaattisesti yhtiölle. Yhdysvalloissa tekijän oikeuden haltija on työn omistaja, ja tämä omaa erilaisia

oikeuksia työhön. Nämä oikeudet voidaan myydä tai siirtää toiselle henkilölle. Moraaliset oikeudet säilyvät aina tekijällä, mutta taloudelliset oikeudet voi siirtää sopimuksella esimerkiksi yritykselle tai oppilaitokselle. Tekijänoikeusteollisuus onkin yksi laajimmista ja nopeimmin kasvavista aloista Yhdysvaltojen taloudessa. (Burnett & Marshall 2002, 138–139)

Tekijänoikeus on myös helppo kiertää esimerkiksi toteuttamalla sama idea hieman erilailla toteutettuna. IT-tuotteiden tekijänoikeudet jättävät siis idean ja rakenteen tekijänoikeussuojan ulkopuolelle, mutta koodi saa suojan. Patentissa puolestaan suojan saa koodin lisäksi rakenne. Myös laittomia kopioita voidaan tuottaa naapurimaissa laillisesti, jos tekijänoikeuksista ei sovita maiden välillä. (Nice Tuesday -tilaisuus)

Linus Torvaldsin mukaan (Torvalds & Diamond 2001, 204–213) tekijänoikeussuojelu on viety liian pitkälle kuluttajansuojan kustannuksella. Varsinkin ohjelmistopatenttien henkisen omaisuuden liika patenttisuojelu hidastaa tekniikan kehittymistä. Tekijänoikeuslaki edistää enemmän markkinoiden kontrollia kuin innovaatioita. Kuitenkin hän muistuttaa, ettei insinööri keksi yhtään mitään ellei hänelle makseta palkkaa.

Tietokoneohjelmien ja immateriaalioikeuksien välinen suhde on monimutkainen ja kiistanalainen. Tämä johtuu enimmäkseen siitä että ohjelmistoteollisuus on suhteellisen uusi ala ja täten sen erityispiirteitä ei ole voitu ottaa huomioon erilaisia immateriaalioikeuksia luotaessa. Suojamuoto on kuitenkin välttämätöntä. Ohjelmiston suunnittelu ja valmistaminen on kallista, kun taas kopiointi on käytännössä ilmaista ja helppoa.

2.2 Patentti

Tekijänoikeus ei ole kaikkien mielestä riittävä suoja ohjelmistoinnovaatioille. Yhtenä vaihtoehtoisena suojamuotona pidetäänkin patentoimista, vaikka ohjelmistoinnovaatioita ei voidakaan patentoida tällä hetkellä Suomessa.

Patentti on viranomaisen antama määräaikainen yksinoikeus keksinnön hyväksikäyttämiseen ammattimaisessa toiminnassa. Patentti on luonteeltaan kielto-oikeus, joka on alueellisesti rajallinen, eli se on voimassa vain niissä valtioissa, joissa patenttia on haettu ja joissa hakemus on hyväksytty. Patentin haltijalla on oikeus kieltää muilta keksintönsä ammattimainen hyväksikäyttö. Patentti suojaa sen keksinnön tai ne keksinnöt, jotka on määritelty patenttivaatimuksissa. Keksinnön selitystä voidaan käyttää apuna vaatimusten tulkinnassa. Patentti on voimassa korkeintaan 20 vuotta, edellyttäen, että siitä maksetaan vuosittain ylläpitomaksu (Patentti- ja rekisterihallitus, 2007).

Tietokoneohjelmia suojelee perinteisesti tekijänoikeus, mutta Yhdysvalloissa ohjelmistoinnovaatiot ovat vähitellen asettuneet myös patenttioikeuden alaisiksi. Suomen patenttilain 1 §:ssä esitetään ne tapaukset, joita ei voida katsoa patentilla suojattaviksi keksinnöiksi ja tietokoneohjelma on yksi niistä tapauksista (Patenttilaki, 2007). Euroopassa ja myös Suomessa ovat erityisesti isot yritykset ajaneet tähän muutosta, koska Yhdysvalloissa patentointi on mahdollista ja Euroopassa ei ole.

EU-komissio vastasi muutostarpeisiin ja antoi direktiiviehdotuksen tietokoneohjelmien patentoimisesta. Esityksen tarpeellisuutta on perusteltu tarpeella harmonisoida eri maiden sekä Euroopan ja Yhdysvaltojen välinen patenttikäytäntö. Harminisointiehdousta perustellaan eurooppalaisten IT-yrityksien kilpailuaseman heikkoudella verrattuna Yhdysvaltoihin. Ehdotus on herättänyt sekä vastustusta että protesteja ja keskustelu aiheesta on ollut vilkasta. Keskusteluissa nousee usein esiin, ettei pelkkä tekijänoikeus riitä suojaamaan ohjelmistotuotteita, koska se ei suoja varsinaista ideaa. Toisaalta on esitetty, että ohjelmistopatenteista on lähinnä haittaa kilpailijoille ja yhteiskunnalle, ja ne voivat itsessään johtaa samaan lopputulokseen. Eräät korostavat, että tietokonetekniikka kehittyy paljon nopeammin kuin perinteiset teollisuudenalat eikä patenteista olisi käytännössä hyötyä tekniikan nopean vanhentumisen takia. Europarlamentti hylkäsi direktiiviehdotuksen lopullisesti 6. heinäkuuta 2005. (Wikipedia, 2007)

Open Source -liike suhtautuu yleensä hyvin negatiivisesti tietokoneohjelmien patentoimiseen. Välimäki (2004) toteaa, että ohjelmistopatenttien on nähty uhkaavan innovaatioita ja näin ollen niiden on nähty myös uhkaavan avoimen lähdekoodin lähestymistavan toiminnallisuutta. Muutenkin avoimen ohjelmiston etiikka ja filosofia ovat voimakkaasti kaikenlaisia patenteja vastaan. Lisäksi yhdysvaltalainen GNU:n (*GNU's Not Unix*) perustaja Richard Stallman on perustanut ilmaisen patenttipoolin, jonka tarkoituksena on palauttaa patentit julkiseksi omaisuudeksi (Besaha, 2003).

Open Source -liikkeen reaktiot ovat täysin ymmärrettäviä. Erilaiset Open Source -ohjelmat ja ilmaiset ohjelmat ovat epäedullisessa asemassa patenttien suhteen koska patenttijärjestelmän piirissä toimiminen vaatii resursseja, joita heillä ei voida olettaa olevan.

Patentoinnilla on kuitenkin lukuisia hyviä puolia muihin suojamuotoihin verrattuna. Tekijänoikeuksiin verrattuna patentit ovat ainakin periaatteessa sisällöltään selkeitä hallintopäätöksiä. Tekijänoikeuksia ei rekisteröidä, joten niitä on vaikea hallinnoida. Patenteja puolestaan voidaan hakea joko kansallisesta patenttivirastosta tai Euroopan patenttivirastosta. (Lampola, 2002) Patentoinnin

avulla saadaan etumatkaa kilpailijoihin nähden ja aikaa ja rauhaa oman tuotteen kehittämiseen valmiiksi. Tietokoneohjelmien patentointi tarjoaisi PK-yrityksille suojaa suuria yrityksiä vastaan, mikä muun muassa lisäisi innovatiivisuutta.

3. Avoin ja vapaa ohjelmisto ja niiden historia

Avoin lähdekoodi (Open source) eroaa suljetusta lähdekoodista (Closed source) täysin. Suljettujen lähdekoodien ohjelmia voi ainoastaan vapaasti käyttää, mutta niiden tutkiminen on mahdotonta, eikä niiden tietosuojaongelmia voi itse korjata eikä ohjelmia voi muokata itselleen paremmin sopiviksi. Suljettujen lähdekoodien ohjelmien levittäminen eteenpäin on ehdottomasti kielletty. Ohjelman lähdekoodi käsitetään usein ohjelmistoyrityksen liikesalaisuudeksi. Avoin lähdekoodi puolestaan tarkoittaa sitä, että ohjelma on saatavissa lähdekoodina. Tällöin poistuu tekninen este ohjelman rajattomaan muokkaukseen. Useimmiten avoimeen lähdekoodiin liittyy myös oikeus kopioida, muokata ja levittää ohjelmaa vapaasti.

Avoin lähdekoodi ei sinänsä ole mikään uusi tapa levittää ohjelmia. Tietotekniikan kehityksen alkuvuosikymmeninä kaikkien ohjelmistojen lähdekoodit olivat avoimia ja niiden kehittäminen tapahtui yhteistyön hengessä. Suuri osa ohjelmistoista tuotettiin yliopistojen ja yritysten yhteistyöhankkeina. Ohjelmien lähdekoodi oli aina vapaasti saatavilla, ja ihmiset julkaisivat ohjelmiansa sekä antoivat muiden kopioida, käyttää ja muunnella niitä vapaasti. Ohjelmistojen kehittäjät saivat palkan ohjelmoinnista, eivät ohjelmista. Ohjelmilla oli vähän itsenäistä kaupallista merkitystä. Myöhemmin tietokoneiden yleistyessä ohjelmistokehitys kaupallistui ja muutos kohti suljettujen ohjelmistojen kulttuuria alkoi. (Perens, 1999; Stallman, 2002)

Ohjelmistojen kaupallistuttua Richard Stallman perusti vastalauseeksi GNU-projektin. Projektin päämääränä oli tuottaa vapaita ohjelmistoja (free software). Stallman perusti myös vuonna 1985 säätiön nimeltä Free Software Foundation (FSF). Stallmanin tavoitteena oli koota kokonainen toimiva käyttöjärjestelmä ja sen tarpeisiin ohjelmistolisenssi GNU General Public License (GNU GPL-lisenssi), jonka olennaisin tehtävä oli estää kaupallisia tahoja kaappaamasta vapaata lähdekoodia itselleen. (Stallman, 2002) GNU GPL-lisenssi on suosituin vapaiden ohjelmistojen levityslisenssi.

Stallmanin tavoitteelta, toimivalta käyttöjärjestelmältä, puuttui käyttöjärjestelmän ydin eli kernel. Torvaldsin ansiosta (2001, 58–59) tämä tarve poistui.

Hän oli ryhtynyt kehittämään omaa Unixin korvaavaa Linux-käyttöjärjestelmää. Torvalds lisensoi Linux-käyttöjärjestelmän ytimen GNU GPL -lisenssillä 1992. Linux on tehokkaan kernelinsä johdosta eräs maailman tehokkaimmista käyttöjärjestelmistä, ja Stallman onkin todennut, että on Torvaldsin ansiota, että GNU käyttöjärjestelmä ylipäättään toimii tänä päivänä. (Stallman, 2002.; Stallman 1997) GNU/Linux on tällä hetkellä suosituin avoimen lähdekoodin käyttöjärjestelmä ja Linux-käyttöjärjestelmä on toiseksi yleisin käyttöjärjestelmä koko maailmassa Windowsin jälkeen. Linuxin huima suosio on luonnollisesti lisännyt kiinnostusta GPL-lisenssiin ja avoimen lähdekoodin ohjelmistokehitykseen.

Free software -termi ei saanut kaupallista kannatusta, joten samalle ideologialle piti kehittää yksiselitteisempi termi. Englanninkielinen free software on terminä harhaanjohtava. Puhuttaessa vapaasta ohjelmasta tarkoitetaan vapautta (freedom). Englannin sana free tarkoittaa nimittäin vapautta, mutta myös muun muassa ilmaista. Ihmiset käsittävät, että free software on ilmaisohjelma, vaikka näin ei välttämättä ole. (Stallman, 2005) Vuonna 1998 perustettiin yhteisö nimeltä Open Source Initiative (OSI) eli Avoimen lähdekoodin aloite, jonka tarkoituksena on tuoda avoimen lähdekoodin ohjelmistot tunnetuksi erityisesti yritysmaailmassa. OSI ja FSF edistävät molemmat lähdekoodin avoimuutta, mutta avoimen lähdekoodin määritelmät ja päämäärät eroavat hieman toisistaan.

OSI:n avoimen lähdekoodin määritelmä on yksityiskohtaisempi kuin vapaan ohjelmiston määritelmä. Open Source on siis ennemmin markkinointitermi ja se painottaa vapaaohjelmistojen käytännön etuja eikä niinkään vapautta. Stallman ei ole hyväksynyt Open Source -termin käyttöä. Hänen mielestään on tärkeää ymmärtää vapaiden ohjelmien taustaan liittyvää ideologiaa, eikä tyytyä käyttämään termiä, joka on eniten oikein. (Stallman 2005)

Open Source eli avoin lähdekoodi tarkoittaa mitä tahansa ohjelmistoa, jonka lisenssi täyttää OSI:n määrittelemät vaatimukset. OSI on julkaissut avoimen lähdekoodin ohjelmistoille kymmenkohtaisen määritelmän (Open Source Definition). Ehdot täyttävää ohjelmistoa voidaan levittää eteenpäin, ja ohjelmisto voi saada OSI Certified -merkinnän. Open Source -termi itsessäänkin on suojattu, ja termiä "OSI certified Open Source" saa käyttää vain ohjelmista, jotka täyttävät tietyt avoimuuden kriteerit. (OSI, 2007) Esittelen kriteerit seuraavassa kappaleessa. OSI on onnistunut rajoittamaan termin open source käyttöä, vaikka se ei ole onnistunut rekisteröidä sitä tavaramerkiksi. Termi on kuitenkin vakiintunut ja sillä on vaikutusvaltaa yritysmaailmassakin. Ohjelmistoja, jotka OSI on tarkastanut ja hyväksynyt, voidaan pitää avoimen lähdekoodin ohjelmistoina.

3.1 Open Source määritelmä

OSI:n (2006) lähdekoodin määritelmän versio 1.9 sisältää seuraavat kohdat:

1. Vapaa levitysoikeus

Lisenssi ei saa estää ketään myymästä tai lahjoittamasta ohjelmaa osana yhdistettyä ohjelmistoa, joka on koottu useista eri lähteistä saaduista ohjelmista. Lisenssissä ei saa määrätä ohjelman myymisen ehdoksi tällaisessa tapauksessa rojaltia tai muuta maksua.

2. Lähdekoodi

Ohjelman täytyy sisältää lähdekoodi ja ohjelman levityksen täytyy olla sallittu sekä lähdekoodina että käännettyssä muodossa. Jos jotakin osaa ohjelmasta levitetään ilman lähdekoodia, on selkeästi ilmoitettava, miten lähdekoodi on saatavissa kohtuullisin kopiointikustannuksin - mieluiten Internetin kautta ilmaiseksi. Suositeltavin levitysmuoto on lähdekoodi, jota ohjelmoija voi muuttaa. Tahallisesti epäselvä lähdekoodi ei ole sallittu. Välimuodot, kuten esiprosessorin tai kielen tulkin tulos, eivät ole sallittuja.

3. Johdetut ohjelmistot

Lisenssin on sallittava muutosten tekeminen ja johdettujen teosten luominen. Näitä on saatava levittää samoilla lisenssiehdoilla kuin alkuperäistä ohjelmaa.

4. Lähdekoodin yhteenkuuluvuus

Lisenssi voi rajoittaa muutellun lähdekoodin levittämistä vain siinä tapauksessa, että lisenssi sallii korjaustiedostojen (*patch*) ja niiden lähdekoodin levittämisen. Korjaustiedostojen tarkoituksena on ohjelman muuttaminen, kun sitä käännetään. Lisenssin on erityisesti sallittava muutetusta lähdekoodista käännettyjen ohjelmien levittäminen. Lisenssi voi edellyttää, että johdetuissa teoksissa käytetään erilaista nimeä tai versionumeroa kuin alkuperäisessä ohjelmassa.

5. Henkilöiden ja ryhmien syrjinnän kieltö

Lisenssi ei saa syrjiä ketään henkilöä tai henkilöryhmää.

6. Toimialojen syrjinnän kieltö

Lisenssi ei saa syrjiä ketään käyttämästä ohjelmaa tietyllä toimialalla. On esimerkiksi kiellettyä rajoittaa ohjelman käyttöä liiketoiminnassa tai genetiikan tutkimuksessa.

7. Lisenssin levittäminen

Ohjelmaan kuuluvien oikeuksien on sovelluttava suoraan kaikille niille, joille ohjelma on levitetty ilman, että heidän tulisi ottaa käyttöön myös jokin uusi lisenssi.

8. Lisenssi ei saa olla tuotekohtainen

Ohjelmaan kuuluvat oikeudet eivät saa riippua siitä, että ohjelma on osana jotakin tiettyä ohjelmistopakettia. Jos ohjelma erotetaan ohjelmistopakettista ja sen jälkeen sitä käytetään tai levitetään ohjelman lisenssillä, niin kaikkien niiden, joille ohjelma levitetään, tulee saada oikeudet, joita ohjelma tuotti alkuperäisessä ohjelmistopakettissa.

9. Lisenssi ei saa rajoittaa muita ohjelmia

Lisenssi ei saa asettaa rajoituksia muille ohjelmille, joita levitetään lisensoidun ohjelman mukana. Lisenssi ei saa esimerkiksi vaatia, että kaikki muut ohjelmat, joita levitetään samalla tallennusvälineellä, olisivat avoimen lähdekoodin ohjelmia.

Rosen (2004) kritisoi avoimen lähdekoodin määritelmää liian epäselväksi. Rosen määrittelee avoimen lähdekoodin viiden periaatteen mukaisesti, joissa yhdistyvät sekä OSI:n että FSF:n määritelmät. Rosen löytää OSI:n määrittelystä ongelmia niin sisällöllisesti kuin kielellisesti. (Rosien, 2004, 7-8)

Tällä hetkellä ohjelmistoteollisuudessa ei ole täyttä yksimielisyyttä avoimen lähdekoodin ohjelmistojen määritelmästä eikä niitä koskevasta sopimuskäytännöstä. Liikkeillä on omat kannattajansa ja puolestapuhujansa. Jatkossa puhuttaessa avoimesta lähdekoodista sillä viitataan myös vapaisiin ohjelmistoihin sekä OSI:n hyväksymien lisenssien alaisiin ohjelmistoihin tai niiden osiin.

4. Lisenssit

Tietokoneohjelmistoa ei yleensä myydä vaan se lisensoidaan. Microsoft on erityisen tunnettu lisensoijistaan. Asiakas ostaa lisenssin, joka oikeuttaa käyttämään ohjelmaa. Microsoftin lisenssit ovat maksullisia, mutta lisenssi tulee myös ilmaiseksi levitettävän ohjelman mukana.

Tietokoneohjelmia suojataan Suomessa tekijänoikeuslain 27 §:n 2 momentin mukaan (Suomen laki, 2007) kirjallisena teoksena. Myös tietokoneohjelmien lisensointi perustuu samaan pykälään, jonka mukaan teoskappaleen luovutus ei sisällä tekijänoikeuden luovutusta, ellei tästä ole erikseen nimenomaisesti sovit-

tu. Tekijänoikeuden haltija siis antaa lisenssisopimuksella ohjelmaan rajoitetun käyttöoikeuden, mutta ei luovu tekijänoikeudesta ohjelmaan. Lisenssit perustuvat siis pitkälti tekijänoikeuden pohjalle.

Esimerkiksi suljetun lähdekoodin ohjelmistot ovat huomattavasti rajoitetummin lisensoidut kuin avoimen lähdekoodin ohjelmistot. Voidaankin todeta, että avoimen lähdekoodin ohjelmistojen kohdalla ohjelmistojen lisenssien tarkoitus on tukea saatavuuden periaatetta lain kannalta. Jotta ohjelmistoa voidaan ylipäätään kutsua avoimen lähdekoodin ohjelmistoksi, on sen käytännössä mukauduttava jonkin avoimen lähdekoodin ohjelmiston identifioivan lisenssin periaatteisiin. (Välimäki, 2002)

Erilaisia avoimen lähdekoodin lisenssejä on kymmeniä. Suurin toiminnallinen ero eri avoimen lähdekoodin lisensseissä on niiden reagoinnissa teosten muunteluun ja yhdistämiseen. Avoimen lähdekoodin lisenssillä saadaan lupa tekijänoikeuden haltijalta ottaa ja tehdä käytön edellyttämiä kopioita. Normaalisti tekijänoikeus kieltää teoksen muokkaamisen, kopioimisen ja teoksen käyttämisen ilman tekijänoikeuden haltijan lupaa.

On kuitenkin tärkeää huomioida, etteivät kaikki lisenssit mahdollista esimerkiksi lähdekoodin vapaata siirtämistä tai muokattavuutta edelleentuetettaviksi suljetun lähdekoodin ohjelmistoiksi. Tämän vuoksi on erittäin tärkeää tunnistaa käytetyn lisenssin suomat oikeudet ja toisaalta sen ohjelmistolle asetamat rajoitukset riittävän ajoissa mahdollisten ongelmatilanteiden välttämiseksi. (Välimäki, 2002)

Avoimen lähdekoodin lisenssillä lisensoitu tietokoneohjelma täyttää OSI:n avoimen lähdekoodin määritelmät. Vaikka avoimen lähdekoodin ohjelmaa voidaan levittää ilmaiseksi, ei se vaikuta velvollisuuteen noudattaa lisenssiehtoja.

Rosen (2004) on jakanut lisenssit neljään kategoriaan:

1. Akateemiset lisenssit ovat luonteeltaan erittäin sallivia. Lisenssi sisältää hyvin vähän rajoittavia ehtoja. BSD-lisenssi on akateemisten lisenssien tyyppiesimerkki.
2. Vastavuoroiset lisenssit kuten GPL ja LGPL. Jokaisen, joka esimerkiksi levittää kopioita GPL ja LGPL lisenssien alaisista ohjelmista, ilmaiseksi tai maksusta, on annettava käyttäjille samat ehdot, jotka ohjelman levittäjällä on.
3. Standardilisenssit kuten esimerkiksi MPL. Sopimusehdoin suojataan tiettyjä standardeja.
4. Sisältölisenssit kuten Creative Commons Deed, harvemmin ohjelmistoissa käytetty.

Akateemisten lisenssien kuten esimerkiksi BSD-tyyppisiä lisenssejä voi kuvailla kaiken salliviksi. Ne antavat käyttäjälle vapauden paitsi käyttää, muokata, kopioida ja levittää ohjelmistoa myös muuttaa ohjelman lisenssiehtoja. Ainut vaatimus lisenssiehtojen muuttamiselle on, että teosta muunnellaan sen verran, että siitä tulee uusi tai jälkiperäinen teos, johon muuntelijalla on tekijänoikeus. BSD-tyyppien lisenssien käyttö on kaupallisessa mielessä perusteltua, jos tekijä haluaa kolmansien osapuolten kehittävän ohjelmistoon perustuvia tai sitä tukevia kaupallisia ohjelmia. Toisaalta tekijän on otettava huomioon se mahdollisuus, että kolmas voi myös kaapata BSD-lisenssin alaisen ohjelmiston ja kehittää kilpailevan yksinoikeuslisensoidun tuotteen. Siksi BSD-lisenssiä suositellaankin varauksettomasti käytettäväksi vain sellaisille ohjelmistotuotteen komponenteille, jotka tukevat ydintuotteen kaupallista käyttöä mutta joiden kehittämiseen alkuperäisellä tekijällä ei ole intressiä. (Välimäki, 2002)

Tunnetuin avoimen lähdekoodin lisenssi on GPL, joka kuuluu puolestaan vastavuoroisiin lisensseihin. GNU lisenssit eli GPL ja LGPL -lisenssit vaativat muun muassa, että lisenssiä ei saa muuttaa ja että lähdekoodi on aina toimitettava binääriohjelman mukana. GPL-lisenssin erityspiirteinä ovat lisenssin pysyvyys (*persistence*) ja virusvaikutus (*viral effect*). Lisenssin pysyvyys tarkoittaa, että kaikki teoksen muunnelmat on lisensoitava GPL-lisenssillä. Virusvaikutus tarkoittaa, että GPL-lisensoidun teoksen yhdistäminen jonkin toisen lisenssin alaiseen teokseen johtaa helposti siihen, että GPL-lisenssi syrjäyttää toisen lisenssin ja tulee kattamaan yhdistetyn teoksen käyttöehdot. Toisinsanoen GPL-virus ikäänkuin tartuttaa muut ohjelmistot. LGPL-lisenssi (GNU Lesser General Public License) on rakenteeltaan ja ideologiselta päämäärältään samankaltainen GPL-lisenssin kanssa. Virusvaikutusta LGPL-lisenssillä ei kuitenkaan ole. (Välimäki, 2002)

GPL- ja LGPL -lisenssit eivät tunnusta sellaista ohjelmiston kehitystä, joka vaatisi minkäänlaisten lisenssimaksujen hyväksymistä olemassa olevista patenteista. Mikäli siis patentti jollekin innovaatiolle on olemassa, ei ohjelmistonkehitystä pystytä enää kyseisten lisenssien ehtojen mukaan jatkamaan. Avoimen lähdekoodin mukainen ohjelmistotuotanto tuleeikin ongelmalliseksi, mikäli patenttien määrä kasvaa kasvamistaan. (Välimäki, 2002)

MPL-tyyppisten lisenssien ajatus on, että teoksen alkuperäisellä liikkeellelaskijalla on enemmän oikeuksia teokseen kuin kehittäjäyhteisöllä. Lisenssi on lähdekoodin osalta pysyvä mutta alkuperäinen liikkeellelaskija pidättää oikeuden lisensoida objektikoodin myöhemmin millä tahansa muulla lisenssillä. MPL eroaa muista lisensseistä siis muutoksia käsitteleviltä kohdiltaan. MPL

jakaa ohjelmiston ja siihen tehtävät muutokset kahteen eri tyyppiin; alkuperäinen koodi ja jatkokehitysversio. Eli lisenssi mahdollistaa eri käyttömuodot ohjelmoijille ja loppukäyttäjille. Virusvaikutusta MPL-tyypin lisensseillä ei ole. Tätä lisenssityyppiä voi pitää yhtenä kaupallisesti perusteltuna vaihtoehtona GNU- ja BSD-tyyppisille lisensseille. Suuryrityksistä esimerkiksi Sun Microsystems ja Nokia käyttävät MPL-tyyppisiä avoimen lähdekoodin lisenssejä. Lisenssityypin suurimpana ongelmana on pidetty epäyhteensopivuutta GNU-lisenssien kanssa. (Välimäki, 2002)

Sisältölisenssejä puolestaan käytetään harvemmin suojamaan ohjelmistoja. Lisenssit suojaavat enemmän eri taiteita: kirjallisuutta, kuvataiteita, musiikkia ja elokuvia. Suosituin lisenssityyppi on Creative Commons. Creative Commons tarjoaa erilaisia lisenssejä, joilla voi määrätä tekijänoikeuslain suomista oikeuksista. Samalla Creative Commons kokoaa verkkoon helposti etsittävään muotoon tällaisia teoksia. Projekti tarjoaa useita erilaisia lisenssejä materiaalin julkaisemiseen Internetissä. (Wikipedia, 2007)

Avoimen lähdekoodin lisenssiä valitessa on harkinta tehtävä tapauskohtaisesti. Tärkein asia joka tulee ottaa huomioon lisenssiä valitessa on ohjelmiston kohdemarkkinta. Jos tuote on kohdistettu loppukäyttäjille, ei lisenssistä ole suurta hyötyä. Toisaalta jos ohjelmistoa käytetään ympäristössä, jossa kilpailevat tuotteet ovat vapaiden lisenssien alaisia. Tällöin vapaa lisenssi luo kilpailuedun yksinoikeuslisensseillä levitettäviin ohjelmistoihin nähden. Ohjelmoijille ja muille lisäarvoa tuottaville suunnatun ohjelmiston lisenssin on oltava yksinoikeuslisenssiä sallivampi, jotta kolmansilla osapuolilla olisi kannustimia tukea ohjelmistoa. (Välimäki 2002)

5. Avoimen lähdekoodin hyödyt ja riskit

Eric Raymondin (Raymond, 1997) mukaan Open Source -ohjelmistokehityksen tarkoituksena on tehdä jatkuvasti entistä parempia ohjelmia. Hänen mukaan ohjelmistokehityksessä voidaan erottaa kaksi tapaa, jolla ohjelmistoa voidaan kehittää Open Source -ympäristössä: "katedraali"- ja "basaari"-tyylit. Katedraali-tyylisessä ohjelmistokehityksessä ohjelman kehittämisestä vastaa pieni joukko ihmisiä, jotka päättävät mitä muutoksia ohjelmaan tehdään. Ongelmaksi tässä tyylissä muodostuu se, että ohjelman kehittäminen saattaa muodostua vaikeaksi kun eri versioiden julkaisuvälit ovat pitkiä, jolloin ohjelman kehitys etenee hitaammin. Basaari-tyylisessä ohjelmistokehityksessä lähdekoodi on julkinen ja suurempi joukko ihmisiä tekee siihen muutoksia. Tästä on hyötyä, sillä

voidaan olettaa että mitä useampi ihminen lukee koodia, sitä useampi virhe löydetään ja saadaan korjattua.

Avoimen lähdekoodin ohjelmistoja pidetään yleisesti turvallisempina, koska kaikkien on mahdollista tutkia niiden lähdekoodia tietoturvaongelmien varalta. Avoimen lähdekoodin käytön etuihin kuuluu myös tuotannon nopeus. Virheet päivitetään useimmiten nopeasti. Työvoima on ilmaista, ihmiset yleensä kehittävät ominaisuuksia omiin tarpeisiinsa ja näistä hyötyvät myös muut. Myös kustannustekijät tukevat vahvasti avoimen lähdekoodin käyttämistä. (Nice Tuesday -tilaisuuden muistiinpanot, 2007)

Uutta tuotetta kehitettäessä tutustutaan usein kilpailijoiden tuotteisiin ja tuotetta kehitetään myös niiden pohjalta haluten parantaa vanhoja tuotteita ja tuoda niihin lisää komponentteja. Avoin lähdekoodi antaa tähän vielä paremmat mahdollisuudet. Innovointi on helpompaa eikä vanhoja ratkaisuja tarvitse keksiä uudelleen.

Avoimen lähdekoodin ohjelmistojen kiistattomiin etuihin kuuluu testaaminen ja tuloksiin reagointi. Testaus saattaa olla vähäistä ennen tuotoksen julkaisua, mutta testaus on kentällä erittäin kattavaa ja virheet tulevat usein nopeasti korjatuiksi. Ohjelmia testataan ja jatkokehitetään yhteisön tuella. Ohjelmien virheet löytyvät todennäköisemmin, kun niitä testaavilla on lähdekoodi käytössä.

Tietoturvallisuuteen liittyy myös pelko siitä, että yksittäisen osan pettäminen tekisi koko ohjelmistosta käyttökeltottoman, eikä vikaa olisi mahdollista oma-aloitteisesti korjata; sen vuoksi monista osasista koottua järjestelmää pidetään riskittömämpänä vaihtoehtona kuin riippuvuutta yhdestä ohjelmistovalmistajasta

Avoin lähdekoodi tarjoaa lukuisia mahdollisuuksia, mutta samalla myös riskejä. Riskejä minimoitaessa on tunnettava lisenssit perinpohjaisesti. Lisenssirikkomukset ovat yhtäläillä tekijänoikeuden loukkauksia oli sitten kyseessä avoimen lähdekoodin ohjelmisto tai yksinoikeusohjelma. Open Source lisenssin sopimusehdoissa ei ole määritelty minkä maan lainsäädäntö siihen pätee.

Ongelmana on, että ohjelmistot on yleensä suunnattu kokeneille käyttäjille, mistä johtuen toiminnan tehokkuus on helppokäyttöisyyttä tärkeämpää. Myös lisenssien ehdot voivat aiheuttaa ongelmia. Ehdot ovat usein epäselviä ja tulkinnanvaraisia. Lisenssien ehtojen pätevydestä ja tulkinnasta ei ole juurikaan ennakkotapauksia, joten yritykset voivat liikkua niin kutsutulla harmaalla alueella yrittäessään itse tulkita ehtoja. (Nice Tuesday -tilaisuuden muistiinpanot, 2007)

6. Käytännön esimerkki

Avoimen lähdekoodin lisenssien ehdot ovat tulkinnanvaraisia. Vielä on hyvin vähän näyttöä siitä, miten lähdekoodien lisenssejä on tulkittu riita- ja ongelmatilanteissa eri tuomioistuimissa. Sellaisia ennakkotapauksia kuitenkin löytyy, joissa tuomioistuin on tehnyt päätöksen lisenssien väärinkäytöksiensä pohjalta.

Esimerkiksi tiettävästi ensimmäinen oikeudenkäynti oli tapaus SiteCom Saksassa, jossa Münchenin alioikeus (2004) teki päätöksen GNU GPL -lisenssien pohjalta. Brittiläisen Sitecomin saksalaisen tytäryhtiön Sitecom Germany GmbH:n myymä SiteCom WL-122-wlan-reititin asetettiin jakelukieltoon netfilter/iptables -projektin tekemän kieltopyynnön perusteella. Netfilter/iptables ohjelmisto oli lisensoitu eteenpäin GPL:n alla, eikä SiteCom maininnut markkinoinnissaan eikä jakelussaan, että yhtiön ohjelmistotuote sisälsi lisenssiehdon alaisia ohjelmistokomponentteja. Oikeudenkäynti tapahtui alioikeudessa Münchenissä, jossa haettiin väliaikaiskieltoa ohjelmiston myymiselle. Asian arvioinnissa tuomari päätyi tulokseen, että GPL -lisenssin ehdot ovat pääasiassa sitovia. Tuomioistuin sovelsi päätöksessään Saksan tekijänoikeuslakia ja Saksan kauppalakia. Seurauksena oikeus teki Sitecomille ennakkomääräyksen, joka kieltää Sitecomia myymästä tuotettaan ennen kuin kaikki GNU/GPL:n lisenssiehdot on täytetty. (Münchenin alioikeus, 2004)

Tapaus on erinomainen esimerkki siitä, että avoimen lähdekoodin lisenssit on hyvä ottaa yhtä vakavasti kuin tavalliset kaupalliset lisenssit. Tekijänoikeus ja yksinoikeuslisenssit antavat ohjelmistotuottajille vahvan kilpailuaseman, ja tekijänoikeuksien avulla yritys voi saavuttaa jopa määräävän markkina-aseman. Kilpailijoiden on vaikeaa päästä markkinoille ja tämä aiheuttaa kilpailun vähenemistä ja usein hidastaa myös uusien innovaatioiden syntyä. Tällaisissa tapauksissa myös valtiot ovat joutuneet puuttumaan tilanteeseen, jotta terve kilpailutilanne säilyisi. Lähdekoodin avoimuus tukee osaltaan vapaata kilpailua ohjelmistoliiketoiminnassa, estäen mahdollisten monopoliasemien syntymisen.

7. Yhteenveto

Teknologia kehittyä ja ohjelmistot mahdollistavat sen muokattavuuden. Yritykset ovat avanneet omia tuotealustojaan muille kehittäjille avoimiksi, mikä on kiihdyttänyt tuotekehitystä ja parantanut ohjelmistojen laatua. Lähdekoodi mahdollistaa käyttäjien omatoimisen ohjelman ylläpitämisen ja kehittämisen. Se on tarpeen ohjelman virheiden löytämisessä, korjaamisessa sekä yhteensopivien

ohjelmien suunnittelussa. Jokainen käyttäjä voi myös itse päättää korjauksen tai ominaisuuden liittämistä ohjelmaversioonsa.

Keskustelu jatkuu edelleen patenttirintamalla. Siirtyminen patenttijärjestelmään tarjoaisi kansainvälisen tavan hoitaa ohjelmistoinnovaatioiden suojaus. Huolta aiheuttaa kuitenkin tekijänoikeuden ja patenttisuojan yhdistelmä. Yhdistelmä on kilpailuoikeuden näkökulmasta liian vahva, ja on siten markkinoiden tehokkaan toiminnan kannalta ongelmallinen. Lisäksi patentointi voi aiheuttaa ongelmia yhteensopivuuden tavoittelussa. Yhteensopivuutta voidaan pitää edellytyksenä yhteiskunnan kannalta, koska se mahdollistaa terveen kilpailun ja tiedon jakamisen ilman keinotekoisia rajoituksia.

Avoimen lähdekoodin lisenssit sen sijaan saattavat kieltää ohjelmistopatenttien ja patenttilisenssien käytön. Avoin lähdekoodi ei ole ratkaisu kaikkiin ohjelmiin. Yritykset, joille maksetaan niillä olevasta osaamisesta, eivät katoa mihinkään. Aikataulujen tekeminen on vaikeaa, koska vapaaehtoisohjelmoijalta ei voida vaatia samalla tavalla työsuoritusta kuin palkatulta työntekijältä. Projektin pitää myös onnistua houkuttamaan ohjelmoijia osallistumaan siihen.

Kuten aikaisemmin on jo todettu, vaativat teknologian nopea kehitys ja liiketoiminnan kansainvälistyminen suuria haasteita myös juridiikan puolella. Muun muassa lähdekoodin lisenssien tuntemus on erityisen tärkeää, sillä hyvä juridinen valmistelu luo edellytykset ohjelmiston menestykselle.

Linuxin menestys on osoittanut että avoimellakin ohjelmistokehityksellä saadaan aikaan laadukkaita ja kilpailukykyisiä ohjelmia. On mielenkiintoista seurata mihin lähdekoodin muotoon ohjelmistojen kehitys keskittyy tulevina vuosina. Globaalista näkökulmasta avoimet ohjelmistot muodostuvat erittäin kiinnostaviksi. Köyhemmät maat rakentavat omaa IT-osaamista ja teollisuutta, ja resurssien puutteiden takia avoimen lähdekoodin ohjelmistot ovat heille varmasti varteenotettava vaihtoehto.

Viiteluettelo

Besaha, (2003) Bob Besaha, Bounty hunting in the patent base. *Comm. ACM* 46, 3 27-29.

Burnett R., Marshall S. (2003). *Web Theory: An Introduction*. London: Routledge.

DiBona C., Ockman S., Stone M. (1999). *Open Sources: Voices from the Open Source Revolution*. Available as <http://www.oreilly.com/catalog/opensources/book/toc.html>. Checked April 4, 2007.

Haarman, P. (1999). *Tekijänoikeus & lähioikeudet*. Jyväskylä: Gummerrus.

Hyvönen, E. (2003). *Ohjelmistoliiketoiminta*. Helsinki: WSOY.

- Lampola M. (1996). *Multimedian tuottamisen ja jakelun juridinen opas*. Helsinki: Tekes. Haettu 27.4.2007, osoitteesta: <http://www.tekes.fi/julkaisut/juridi/>.
- Lampola M., (2002). *Yliopostotutkimuksen kaupallinen hyödyntäminen*. Haettu 27.4.2007 osoitteesta: <http://www.sitra.fi/Julkaisut/raportti21.pdf>
- Münchenin alioikeuden päätös (2004). Available as http://www.jbb.de/judgment_dc_munich_gpl.pdf. Checked March 31, 2007.
- Nice Tuesday -tilaisuuden muistiinpanot (2007). Julkaisematon.
- OSI (2006). April 11. 2007, available as <http://www.opensource.org/history>.
- Open Source definition (2007). Available as <http://www.opensource.org/docs/definition.php>. Checked April 1, 2007.
- OpenOffice (2007). Haettu 1.4.2007, osoitteesta <http://fi.openoffice.org>.
- Opetusministeriö (2007). Kansainvälinen tekijänoikeus ja sopimukset. Haettu 27.4.2007, osoitteesta: http://www.minedu.fi/OPM/Tekijaenoikeus/kansainvaelinen_tekijaenoikeus_ja_sopimukset/?lang=fi.
- Opetusministeriö (2005). Tekijänoikeuslain muutokset voimaan vuodenvaihteessa. Haettu 27.4.2007, osoitteesta http://www.minedu.fi/OPM/Tiedotteet/2005/10/tekijanoikeuslain_muutokset_voimaan_vuodenvaihteessa?lang=fi.
- Opetusministeriö (2007). Tekijänoikeusjärjestelmä ja toimijat. Haettu 19.4.2007, osoitteesta http://www.minedu.fi/OPM/Tekijaenoikeus/tekijaenoikeusjaerjestelmae_ja_toimijat/?lang=fi.
- Patenttilaki. Haettu 17.5.2007, osoitteesta: <http://www.finlex.fi/fi/laki/ajantasa/1967/19670550?search%5Btype%5D=pika&search%5Bpika%5D=patenttilaki>
- Patentti- ja rekisterihallitus (2007). Haettu 19.4.2007, osoitteesta <http://www.prh.fi/fi/patentit/useinkysyttya.html#01>.
- Perens B. The open source definition, version 1.0 (1999). In Chris DiBona, Sam Ockmann & Mark Stone (eds.), *Open Sources: Voices from the Open Source Revolution*. Available as <http://www.oreilly.com/catalog/opensources/book/toc.html>. Checked April 8, 2007.
- Raymond E. (1997). *The Cathedral and the Bazaar*. Available as <http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/idex.html#catbmain>. Checked April 11, 2007.
- Rosen L. (2004). *Open Source Licensing Software Freedom and Intellectual Property Law*. Available as <http://www.rosenlaw.com/oslbook.htm>. Checked March 17, 2007.
- Stallman R., (1999). *The GNU Operating System and the Free Software Movement*. In: DiBona C, Ockman S., Stone M. *Open Sources: Voices from the Open Source*

- Revolution*. Available as <http://www.oreilly.com/catalog/opensources/book/toc.html>. Checked April 8, 2007.
- Stallman R. (1997). *Linux and the GNU Project*. Available as: <http://www.gnu.org/gnu/linux-and-gnu.html>. Checked April 19, 2007.
- Stallman R. (2002). *About the GNU project*. Available as: <http://www.gnu.org/gnu/thegnuproject.html>. Checked April 8, 2007.
- Stallman R. (2005). *Why "Free Software" is better than "Open Source"*. Available as <http://www.fsf.org/licensing/essays/free-software-for-freedom.html>. Checked April 4, 2007.
- Suomen laki (2007). tekijänoikeuslaki 8.7.1961/404. Haettu 19.5.2007 osoitteesta: <http://www.finlex.fi/fi/laki/ajantasa/1961/19610404>.
- Tekes (2007). Sanastoa. Haettu 19.4.2007 osoitteesta: <http://www.tekes.fi/rahoitus/yritys/juridi/sanastoa.html>
- Torvalds L., Diamond D. (2001). *Just for fun*. Texere: London.
- Välimäki M. (2002) *Avoimen lähdekoodin ohjelmistolisensseistä*. Haettu 19.5.2007, osoitteesta: http://www-old.hiit.fi/de/valimaki_dl_0502.pdf.
- Välimäki M. (2004). *A practical approach to the problem of open source and software patents*. Available as http://www.valimaki.com/org/os_patents.pdf. Checked April 11, 2007.
- Wikipedia (2007). Creative Commons. Haettu 28.4.2007, osoitteesta: http://fi.wikipedia.org/wiki/Creative_Commons.
- Wikipedia (2007). Patentti. Haettu 17.5.2007, osoitteesta: <http://fi.wikipedia.org/wiki/Patentti>.
- Wikipedia (2007). Wikipedia etusivu. Haettu 28.4.2007, osoitteesta: <http://fi.wikipedia.org/wiki/Wikipedia:Etusivu>.

Dokumentinhallintaohjelmat pienen organisaation näkökulmasta

Alexi Vuorenmaa

Tiivistelmä.

Digitaaliset dokumentit liittyvät erittäin olennaisesti organisaatioiden toimintaan ja toimintatapoihin. Dokumentinhallintaohjelmiston avulla tehostetulla dokumentinhallinnalla voidaan parhaimmillaan tuottaa lisäarvoa koko organisaatiolle. Dokumentinhallinnan toteutukset voidaan luokitella erilaisiin tasoihin, riippuen toteutuksen toiminnallisuudesta ja monimutkaisuudesta. Pienten organisaatioiden dokumentinhallintaan soveltuu parhaiten yksinkertainen asiakas-palvelin -mallin dokumentinhallintaohjelmisto. Pelkkä dokumentinhallintaohjelmiston käyttö ei sinällään tehosta organisaation dokumentinhallintaa, vaan tärkeintä olisi suunnitella dokumentinhallinnan rakenne ja sen käyttötarkoitus. Valmiin mallin perusteella on helpompi valita toteutukseen parhaiten soveltuva dokumentinhallintaohjelmisto. Tässä tutkielmassa on esimerkkeinä käytetty Motive Systemsin M-Files -ohjelmistoa ja Canonin ADOS-ohjelmistoa.

Avainsanat ja -sanonnat: Dokumentti, dokumentinhallinta, dokumentinhallintaohjelmisto.

CR-luokat: H.3, H.4

1. Johdanto

Paperidokumentit ovat olleet käytössä tallennusvälineinä jo vuosisatoja – digitaaliset dokumentitkin yleisesti käytössä jo useampia vuosikymmeniä. Yleisimmin tietokonetta työssä käytettäessä on kyse erilaisten dokumenttien luomisesta, hallinnasta, poistamisesta tai niiden hakemisesta. Digitaalisten dokumenttien hallintaan, käsittelyyn ja jakeluun on kehitetty useita ratkaisuja, jotka ovat toisiaan tukevia tai vaihtoehtoisia. Näitä ovat esimerkiksi Zantoutin ja Maririn [1999] mukaan ryhmätyöohjelmistot (Groupware), tiedonhallintajärjestelmät (Knowledge Management System) ja dokumentinhallintajärjestelmät (Document Management System, DMS). Tässä vertailututkielmassa käsittelemme ainoastaan dokumentinhallintaohjelmistoja.

Tutkielmani aiheena on dokumentinhallintaohjelmistojen vertailu erityisesti pienten organisaatioiden näkökulmasta. Tarkoitukseni on vertailla muutamia erilaisia valmiita, markkinoilla olevia dokumentinhallintaohjelmistoja ja arvioida niiden toiminnallisuutta sekä soveltuvuutta erityisesti pienten organisaatioiden vaatimukseen ja käyttötarpeisiin. Pienillä organisaatioilla tarkoitan tässä yhteydessä 3–10 käyttäjän organisaatioita. Oleellista on tässä yhteydessä määritellä pieni organisaatio myös sen käytössä olevien eri dokumenttien lu-

kumäärän ja organisaation toimintojen laajuuden perusteella. Tähän määrittelyyn palataan tarkemmin luvussa 2.

Kovin laajoihin käyttökokeiluihin ja -arviointeihin ei tässä tutkielmassa mennä, joten keskityn lähinnä valikoitujen ohjelmistojen ominaisuuksien, toimintojen ja vaatimusten vertailuun sekä ohjelmistojen sopivuuden arviointiin pienten organisaatioiden tarkoituksiin. Tietotekniikka-arkkitehtuurien osalta tämän tutkielman puitteissa käsitellään ainoastaan Microsoft Windows ja Office -työasema- ja palvelinympäristöä. Oletan, että tämä sovellusympäristö on kuitenkin selvästi yleisin juuri pienten organisaatioiden käytössä.

2. Määritelmiä

2.1. Dokumentit

Elorannan ja muiden [2001] mukaan sanalla dokumentti on monia merkityksiä, mutta pohjimmiltaan dokumentti ”sisältää tietoa jossain muodossa, missä sitä voidaan jakaa, levittää, tallentaa ja käsitellä”. Dokumentti koostuu dokumentin sisällöstä ja dokumenttia kuvaavista metatiedoista, joita ovat esimerkiksi dokumentin omistus- ja käyttöoikeus, historia- ja luontitiedot sekä tila. Tämän määritelmän perusteella kaikki tieto, joka on tallennettu luettavissa olevaan lähteeseen, muodostavat dokumentin. Tyypillisiä dokumentteja ovat esimerkiksi pöytäkirjat, lomakkeet, piirroksot, sähköpostit ja erilaiset asiakirjat, olivat ne sitten tallennettuna tietokantaan tai pöytälaatikkoon.

Zantout ja Marir [1999] toteavat, että kaikki organisaatioiden toiminta liittyy jollakin tavalla organisaation dokumentteihin. Organisaation toiminta on siis esimerkiksi dokumenttien tuottamista, muokkaamista, lukemista ja jakelua. Zantout ja Marir myös jaottelevat organisaation dokumentit sisäisiin ja ulkoiisiin dokumentteihin. Organisaation sisäisiä dokumentteja ovat muun muassa erilaiset muistiot, työaikataulut, myyntilaskelmat ja -ennusteet, sekä hallintoon liittyvät asiakirjat ja erilaiset koulutusmateriaalit. Organisaation ulkoisia dokumentteja sen sijaan ovat erilaiset esitykset, mainos- ja markkinointimateriaalit, tilaukset ja tarjoukset sekä tiedustelut, valitukset ja hinnastot.

2.2. Dokumentinhallintajärjestelmät

Eloranta ja muut [2001] esittelevät dokumentinhallinnan mallin, jossa dokumentinhallinta jaetaan kolmeen osaan: dokumenttien virtaus, dokumenttien elinkaari ja niiden jakamiseen liittyvät toiminnot. Näin ollen monipuolisen dokumentinhallintajärjestelmän tulisi pystyä toteuttamaan kaikki nämä ominaisuudet, alkaen dokumentin luomisesta sen työstämisen kautta jakeluun.

Zantout ja Marir [1999] määrittelevät dokumentinhallintajärjestelmien tärkeimmiksi seuraavat käyttäjän käytettävissä olevat toiminnot:

- suora dokumenttien muokkaaminen
- dokumenttien hakemistointi ja tallennus niiden löytämistä varten
- kommunikointi dokumenttien vaihdon kautta
- yhteistoiminta dokumenttien parissa
- dokumenttien virtauksen mallinnus ja automatisointi.

Suurin osa organisaatioiden tapahtumista ja toiminnoista on jossain muodossa dokumentoituina (emt.). Näin ollen dokumentit liittyvätkin erittäin olennaisesti organisaatioiden toimintatapoihin, ja dokumentinhallintajärjestelmät voidaan nähdä olennaisina organisaatioiden toimintatapoja tukevinä, tai jopa niitä määrittävinä, tekijöinä. Tehostetulla dokumentinhallinnalla voidaan siis tuottaa lisäarvoa koko organisaatiolle [Eloranta et al., 2001].

2.3. Pieni organisaatio

Tässä yhteydessä pienellä organisaatiolla tarkoitetaan alle kymmenen henkilön ja yli kahden henkilön muodostamaa yhtenäistä organisaatiota. Jotta dokumentinhallintajärjestelmällä voisi olettaa saavutettavan edes jonkinlaista hyötyä, oletetaan tässä yhteydessä pienen organisaation dokumenttien käsittely melko laajaksi. Toisin sanoen oletetaan, että organisaatiolla on useiden vuosien varrelta kertynyt lukuisia dokumentteja sekä useimmilla niistä olevan useita eri tekijöitä ja lukijoita. Arvioitu dokumenttien määrä voidaan olettaa olevan väliltä 1000–9000 dokumenttia, sisältäen erilaisia asiakirjoja, laskentataulukoita, esityksiä, grafiikoita ja kuvia, kaikki digitaalisessa muodossa tallennettuina. Lisäksi oletettakoon, että useita eri dokumentteja käytetään ja muokataan päivittäin, mutta toisaalta joitakin dokumentteja vain muutaman kerran vuodessa, mutta vuodesta toiseen. Oletetaan organisaation toimintaan liittyvän muun muassa tavallisimpien toimistorutiineiden hoitoa, kuten kirjanpitoa, laskutusta ja toiminnan suunnittelua.

Yllä esitetyillä oletuksilla saadaan luotua mielenkiintoinen skenaario, jossa siis pyritään annettuihin, pienelle organisaatiolle tyypillisiin lähtökohtiin löytämään sopivin mahdollinen dokumentinhallintajärjestelmä.

3. Dokumentinhallinnan erilaisia tasoja

Dokumentinhallinnan toteutukselle voidaan nähdä olevan erilaisia tasoja, riippuen toteutuksen toiminnallisuudesta ja monimutkaisuudesta. Seuraavassa on esitelty eri perustasot ja niiden toiminnallisuuksien eroja dokumentinhallinnassa.

3.1. Käyttöjärjestelmän kansiorakenne, nollasso

Ensimmäinen dokumentinhallinnan tasoista on nollasso, jossa siis ei ole varsinaista dokumentinhallintaohjelmistoa käytössä ollenkaan, vaan dokumentit on

järjestetty käyttöjärjestelmän mahdollistamalla hakemistorakenteella eritasoi-
siin kansioihin. Dokumentit ja kansiot voivat sijaita joko paikallisesti työase-
malla tai sitten yhteiskäytössä olevalla palvelimella. Tämä ei vaikuta pelkkiin
kansioihin perustuvan järjestelmän luokitteluun nollatason dokumentinhallin-
naksi. Kukin tiedosto voi sijaita vain yhdessä paikassa, vain yhden kansion
sisällä. Tiedostojen siirto, kopiointi, poisto, käyttöoikeuksien määrittely ja ja-
kaminen tapahtuvat suoraan käyttöjärjestelmän tarjoamien toimintojen avulla.
Samoin hakutoiminnot ovat käyttöjärjestelmän tarjoamia toimintoja.

Nollatason dokumentinhallinta on olettavasti yleisesti käytössä pienissä or-
ganisaatioissa. Varsinkaan alle neljän henkilön organisaatioissa monipuolisem-
paan dokumentinhallintaan ei yleensä ole edes tarvetta, muun muassa siksi,
että käyttäjät oppivat varsin helposti tuntemaan toistensa dokumenttien luokit-
telutyylin ja muut toimintatavat.

Nykyisissä käyttöjärjestelmissä kansiorakenteiden hallinta on tehty varsin
käyttäjätavalliseksi. Sen sijaan toiminnallisuuksissa on monia puutteita ver-
rattuna dokumentinhallintaohjelmistoihin. Versionhallintaa ei ole, vaan käyttä-
jän on itse huolehdittava siitä esimerkiksi tallentamalla uusia ilmentymiä ole-
massa olevista dokumenteista tai käyttämällä dokumenttimalleja. Käyttöjärjes-
telmän resurssienhallinnan avulla dokumentteja verkkolevyasemilla käytettä-
essä on myös esimerkiksi vahingossa tehtyjen poistojen palauttaminen ongel-
mallista, ellei tapauksesta riippuen jopa mahdotonta.

3.2. Työasemakohtainen dokumentinhallinta, taso I

Työasemakohtaisessa dokumentinhallinnan tasossa ohjelmisto on asennettu
paikallisesti yksittäisiin työasemiin, jotka voivat olla lähiverkon kautta yhtey-
dessä toisiinsa. Nollatasoon verrattuna dokumentinhallintaohjelmisto tuo muu-
tampia lisätoimintoja. Näitä ovat esimerkiksi yksittäisten dokumenttien version-
hallinta, eli mahdollisuus dokumentin tilan palauttamiseen muokkaushistoria-
sta, monipuolisemmat dokumenttien organisointimahdollisuudet, eli sama tie-
dosto voi sijaita useampien yläotsikoiden alla, sekä monipuolisempi dokument-
tien metatietojen käsittely ja käyttöoikeuksien hallinta.

Tässä tasossa tiedostoja sen sijaan ei ole keskitetty mitenkään, vaan kukin
käyttäjä hallitsee omia tiedostojaan ja niiden käyttöoikeuksien kautta doku-
menttien jakamista. Kukin käyttää siis itsenäisesti toimivaa dokumentinhallin-
taohjelmistoa omien tiedostojensa hallintaan. Tiedostojen jakaminen on toteu-
tettu joko dokumentinhallintaohjelmiston tarjoamien palvelujen avulla, sähkö-
postin välityksellä tai sitten nollatason mukaisesti jaettujen kansioiden avulla.

Nollatasoon verrattuna taso I tarjoaa käyttäjälle dokumentinhallintaohjel-
miston toteuttamat lisätoiminnallisuudet. Ohjelmasta riippuen näitä ovat esi-

merkiksi tiedostojen versionhallinta ja tiedostojen monipuolisempi kategorisointi sekä erilaisten metatietojen käyttö dokumenttien sisällön kuvauksessa.

3.3. Yksinkertainen asiakas-palvelin -mallin dokumentinhallinta, taso II

Asiakas-palvelin -mallin mukaisessa dokumentinhallintaohjelmistossa tiedostot sijaitsevat keskitetysti lähiverkon palvelimella ja niitä käytetään asiakasohjelman avulla työasemilta. Dokumenttivarasto sijaitsee siis palvelimella ja sitä hallinnoi dokumentinhallintaohjelmiston palvelinosa. Käyttäjät ovat yhteydessä tiedostoihin työasemaan paikallisesti asennetun dokumentinhallintaohjelmiston asiakasohjelman avulla. Edelliseen tasoon verrattuna tiedostojen jakaminen ja käyttö on monipuolisempaa, keskitetympää sekä turvatumppaa, koska tiedostot sijaitsevat yhdessä paikassa, jolloin asianmukaisten varmuuskopioidenkin tekeminen on keskitetympää ja automatisoitua.

Tässä tasossa kaikki organisaation dokumentit on siis tallennettu keskitetysti yhteen hallintajärjestelmään, joten kaikki dokumentit ovat määriteltyjen käyttöoikeuksien puitteissa suoraan jaettuja kaikkien käyttäjien kesken. Käyttäjät käyttävät yhteisesti samoja tiedostoja tai ainoastaan omiaan, mutta yhtä lailla kaikki dokumentit sijaitsevat palvelimella dokumentinhallintajärjestelmässä, joka huolehtii dokumenttien käyttöoikeuksista, jakamisesta, muokkauksesta ja versionhallinnasta.

Tasossa II on tasoon I verrattuna keskitetyn hallinnan tuomat edut yhdistettynä dokumentinhallintaohjelmiston toteuttamiin lisätoiminnallisuuksiin. Tasossa II siis yhdistyvät alempien tasojen parhaat ominaisuudet, eli nollatason yksinkertainen ja suora tiedostojen jakaminen lähiverkon kautta sekä tason I monipuolisemmat dokumentinhallintaominaisuudet.

3.4. Seuraavat tasot

En lähde tässä yhteydessä sen tarkemmin tarkastelemaan dokumentinhallinnan seuraavia tasoja, sillä ne eivät tämän tutkielman aiheen rajaukseen sovi. Taso III voisi esimerkiksi olla monimutkaisempi asiakas-palvelin -mallin dokumentinhallinta, jossa isomman organisaation eri yksiköillä on oma asiakas-palvelin -mallinen dokumentinhallintajärjestelmä ja lisäksi yksiköiden väliset järjestelmät on integroitu toisiinsa mahdollistaen suoran tiedonvälityksen. Tämän kaltaiset järjestelmät eivät näkemykseni mukaan ole kiinnostavia pienten organisaatioiden näkökulmasta, joten jätän kaikki seuraavat dokumentinhallinnan tasot tämän tutkielman tarkastelun ulkopuolelle.

4. Vertailussa kaksi dokumentinhallintaohjelmistoa

Valitsin tähän vertailuun kaksi suomenkielistä, toiminnoiltaan riittävän monipuolista mutta myös pienten organisaatioiden käyttötarpeet huomioivaa, koh-

talaisen tunnettua ja laajasti käytössä olevaan dokumentinhallintaohjelmistoa. Molemmat ohjelmistot voidaan asentaa sekä tasolle I että tasolle II.

4.1. Motive Systemsin M-Files

Kotimaisen Motive Systems Oy:n dokumentinhallintaohjelmisto M-Files soveltuu valmistajan mukaan ”hyvin myös pienten ja keskisuurten yritysten dokumenttienhallintaan” [M-Files, 2007]. M-Files-ohjelmisto koostuu muutamasta eri osasta. Seuraavassa valmistajan tietoihin perustuva [Motive Systems, 2006] lyhyt esittely ohjelmisto-osista:

- Asennusohjelma
- Selaa M-Filesia: peruskäyttäjän eniten käyttämä ohjelmisto-osa, jossa dokumentit näkyvät erilaisissa näkymissä
- Etsi M-Filesista: ohjelmisto-osan avulla voi tehdä tarkasti määriteltyjä hakuja dokumenttivarastoon.
- M-Files-asiakasohjelmiston asetukset
- M-Files-palvelin: ohjelmisto-osa, joka vastaa tiedon ja dokumenttien keskitetystä tallentamisesta ja jakamisesta.
- M-Files-palvelimen hallinta: työkalu, jolla säädellään M-Files-palvelimen asetuksia, hallitaan dokumenttivarastoa ja muokataan dokumenttivaraston rakennetta.
- Näytä tilatietoja: ohjelmisto-osa, jolla voi seurata tilatietoja tiedostojen siirrosta.
- WWW-käyttö: M-Filesia on mahdollista käyttää myös www-selaimen avulla normaalin Windows-käyttöliittymän (Selaa M-Filesia) lisäksi.
- Automaattiset päivitykset

Dokumentinhallinnan tasoa I vastaa asennus, jossa kaikki ohjelmiston osat on asennettu samalle työasemalle. Tasoa II vastaa asennus, jossa M-Files-palvelin ja M-Files-palvelimen hallinta on asennettu lähiverkon palvelimelle ja muut osat kuhunkin järjestelmää käyttävään työasemaan.

4.2. Canonin ADOS

Canon Europa NV:n, jota Suomessa edustaa Canon Oy, dokumentinhallintaohjelmisto on nimeltään ADOS. ADOS koostuu erillisistä moduuleista, joista on koostettu erilaisia kokonaisuuksia vastaamaan erilaisia käyttötarkoituksia. Martin [2007] mukaan ADOSista on saatavilla viittä eri versiota, joista tämän tutkielman puitteissa tarkasteluun sopivat ADOS Personal Edition yhdelle työasemalle, eli dokumentinhallinnan taso I, sekä ADOS WorkGroup Edition enimmillään 20 työasemalle, eli vastaa dokumentinhallinnan tasoa II. WorkGroup Edition:iin kuuluvat seuraavat moduulit [Martti, 2007.]:

- ADOS Navigator: peruskäyttäjän työkalu asiakirjojen arkistointiin ja haakuun, selaamiseen, katseluun, editointiin ja tulostamiseen. Sisältää Office Links, Shell Extention ja Viewer -osat.
- ADOS Capture Client: työkalu asiakirjojen suoraan skannaukseen, indeksointiin ja arkistointiin DR-skannereilla.
- ADOS Server: mm. uusien arkistojen, metatietokenttien ja näkymien luominen ja käyttöoikeuksien hallinta. Sisältää osat Administrator, Console, Document Loader ja Rights Server.

4.3. Vertailua

Taulukosta 1 voidaan huomata, ovat vertailut ohjelmistot toiminnoiltaan hyvinkin yhteneviä. Suurimmat erot löytyvät järjestelmävaatimuksista. Tämä on etenkin pienten organisaatioiden kohdalla merkittävä tekijä, sillä mahdollisesta resurssien vähydestä johtuen laitekanta saattaa olla jopa vanhentunutta, eikä näin ollen vastaa ohjelmistojen vähimmäisvaatimuksia.

Eräs merkittävä ero vertailtujen ohjelmistojen ominaisuuksissa on niiden käyttämässä tietokantajärjestelmässä. Martti [2007] ilmoittaa avoimesti Canonin ADOSin käyttävän Microsoftin MSDE-tietokantaa (Microsoft SQL Server Desktop Engine), kun taas M-Filesin osalta sen käyttämää tietokantaa ei ilmoiteta. Canonin ohjelmiston osalta sen käyttämän tietokannan ominaisuudet – niin haitat kuin hyödytkin – voidaan puolueettomasti selvittää, kun taas M-Filesin osalta se ei perustiedoilla ole mahdollista. Tästä syystä kyseinen ero ohjelmistojen välillä on merkittävä.

Olennaista olisi kuitenkin ohjelmistoa valittaessa vertailla sen sopivuutta juuri oman organisaation dokumenttien käyttöön ja niiden linkkausten hallintaan. Toisin sanoen, organisaation tulisi ensin suunnitella dokumenttien hallinnan rakenne ja dokumenttien linkkausten sekä käyttöoikeuksien hallinta perustuen organisaation toimintatapoihin ja käytäntöihin. Vasta huolellisen suunnittelun jälkeen tulisi valita sen toteutukseen parhaiten soveltuva dokumentinhallintaohjelmisto.

Ohjelmisto	<i>M-Files 3.0</i>	<i>Canon ADOS 6.x</i>
Valmistaja	Motive Systems Oy	Canon Europa / Canon Oy
suomenkielinen	kyllä	kyllä
Järjestelmävaatimukset:		
palvelin: käyttöjärjestelmä	Windows 2000, Windows XP, Windows Server 2003 tai	ADOS Workgroup Edition Server: Windows 2000 Server (SP 2 ,

Ohjelmisto	<i>M-Files 3.0</i>	<i>Canon ADOS 6.x</i>
	Windows Vista	Muisti: Minimi 1024 MB) Windows Server 2003 (Muisti : Minimi 1024 MB)
palvelin: muuta	Asennukseen tulee varata noin 100 megatavua tilaa asennuskohteesta. Dokumenttivaraston vaatima tila riippuu tallennettavien dokumenttien ja muiden kohteiden määrästä ja koosta.	Tietokantaohjelmisto: MSDE tai Microsoft SQL Server Storage: RAID, NAS, MOD, JUKEBOX, selainvaatimuksena väh. Microsoft Internet Explorer 5.5
työasema: käyttöjärjestelmä	Windows 2000, Windows XP, Windows Server 2003 tai Windows Vista	ADOS Personal Edition: Windows 2000 Professional (SP 2 , Muisti: Minimi 512 MB) Windows XP (Muisti: Minimi 512 MB) Tietokantaohjelmisto: MSDE tai Microsoft SQL Server ADOS Workgroup Edition Client: Windows XP (Muisti: Minimi 512 MB) Windows 2000 Professional (SP 2 , Muisti: Minimi 512 MB)
työasema: muuta		selainvaatimuksena väh. Microsoft Internet Explorer 5.5
Toiminnot		
dokumenttiarkiston selaus	kyllä	kyllä
dokumenttien haku	kyllä	kyllä
dokumentin versionhallinta	kyllä	kyllä
dokumentin käyttöoikeuksien hallinta	kyllä	kyllä
selainkäyttöliittymä	kyllä	ei, saatavilla erillisenä moduulina Standard Editioniin

Ohjelmisto	<i>M-Files 3.0</i>	<i>Canon ADOS 6.x</i>
liitettävyys MS Officeen	kyllä	kyllä
tuki paperiasiakirjojen skannaukseen	kyllä	kyllä

Taulukko 1: Ominaisuustaulukko

5. Loppuarviointi

Sujuva dokumentinhallinta tukee organisaation toimintaa, ja näin ollen dokumentinhallinnan pitäisi olla myös tärkeä osa organisaatiota ja sen toiminnan prosesseja. Dokumentinhallintaa tehostamalla voidaan tehostaa koko organisaation toimintaa, myös pienten organisaatioiden kohdalla. Sen sijaan pelkkä dokumentinhallintaohjelmiston käyttö ei sinällään kovinkaan paljon tehosta organisaation dokumentinhallintaa. Tärkeintä olisi suunnitella ensin dokumentinhallinnan rakenne ja sen käyttötarkoitus. Valmiin mallin perusteella on myös helpompi valita toteutukseen parhaiten soveltuva dokumentinhallintaohjelmisto. Dokumentinhallintaohjelmiston käyttöönotto ei ole vain ohjelmiston käyttöönottoprojekti muiden joukossa, sillä dokumentinhallintaohjelmiston käyttöönotolla voi olla merkittäviä vaikutuksia koko organisaation toimintatapoihin.

Suosittelavin pienille organisaatioille sopiva dokumentinhallinnan taso on mielestäni taso II, jossa yhdistyvät keskitetyn hallinnan tarjoamat edut sekä dokumentinhallintaohjelmiston tarjoamat laajemmat toiminnot dokumenttien käsittelyyn. Jos käyttäjiä on enemmän kuin kolme, ei mielestäni ole syytä harkita tason I mukaisia yksittäisiä dokumentinhallinnan toteutuksia. Jos tehostetulta dokumentinhallinnalta halutaan ainoastaan muutamia lisätoimintoja, kuten dokumenttien versionhallintaa, eikä olla valmiita suunnittelemaan perusteellisesti koko dokumentinhallinnan järjestelmää, ei mielestäni ole järkevää ottaa käyttöön dokumentinhallintaohjelmistoa. Versionhallintaan ja metatietojen käsittelyyn voi löytyä ratkaisuja muutenkin. Sen sijaan, jos ollaan valmiita suunnittelemaan koko organisaation dokumentinhallinta perusteellisesti, tuovat dokumentinhallintaohjelmistojen tarjoamat palvelut tehokkuutta koko organisaation toiminnalle.

Lähdeluettelo

[Eloranta et al., 2001] Eero Eloranta, Ari-Pekka Hameri and Mika Lahti, Improved project management through improved document management. *Computers in Industry* 45, 3 (2001), 231–243.

- [Zantout and Marir, 1999] Hind Zantout and Farhi Marir, Document management systems from current capabilities towards intelligent information retrieval: an overview. *International Journal of Information Management* **19**, 6 (1999), 471–484.
- [M-Files, 2007] *M-Files -ohjelmiston verkkosivut*. [<http://www.m-files.fi/>] Viitattu: 18.5.2007.
- [Motive Systems, 2006] *M-Files 3 Käsikirja*. Motive Systems Oy. Tampere, 2006. [http://www.m-files.fi/m-files/download/M-Files_3_0_user_guide_fin.pdf] Viitattu 18.5.2007.
- [Martti, 2007] Timo Martti, Canon ADOS 6.x built 30 Myynnin koulutusmateriaalia. Canon Oy, 19.4.2007. Helsinki.