

Eliisa Jauhiainen

**RASKE-MENETELMÄN SOVELTAMINEN: HAVAINTOJA
KAHDESTA JYVÄSKYLÄN YLIOPISTON
OPISKELIJAPROJEKTISTA**

Tietojenkäsittelytieteen

pro gradu -tutkielma

6.7.2005

Jyväskylän yliopisto

Tietojenkäsittelytieteiden laitos

Jyväskylä

TIIVISTELMÄ

Jauhiainen, Raita Eliisa Ursula

RASKE-menetelmän soveltaminen: Havaintoja kahdesta Jyväskylän yliopiston opiskelijaprojektista / Eliisa Jauhiainen

Jyväskylä: Jyväskylän yliopisto, 2005

134 s.

Pro gradu -tutkielma

Tässä tutkielmassa tarkastellaan tietojärjestelmätieteiden menetelmien piirteitä ja tutustutaan dokumenttien hallinnan kehittämisessä käytettävään RASKE-menetelmään. Menetelmällä tarkoitetaan ohjeistusta, jonka mukaan tietojärjestelmiä kehitetään. Tutkielmassa esitetään simpukankuorimalli, jonka mukaan menetelmiä voidaan analysoida. Tutkielman teoriaosuus perustuu kirjallisuuskatsaukseen ja tutkielman empiirisessä osassa tarkastellaan RASKE-menetelmän soveltamista Jyväskylän yliopiston tietojenkäsittelytieteiden laitoksen opiskelijaprojekteissa.

Tutkielman tavoitteena on kertoa miten RASKE-menetelmän soveltaminen onnistui valittujen opiskelijaprojektien tapauksissa. Tutkielman tuloksista havaitaan, että RASKE-menetelmän soveltaminen onnistuu sitä ennen tuntemattomilta. Menetelmän mukaiset mallit ovat myös helposti omaksuttavissa. Lisäksi opiskelijaprojekteissa RASKE-menetelmä koettiin hyödylliseksi menetelmäksi. Tulokset tukevat aikaisemman tutkimuksen oletusta siitä, että menetelmän käyttöönotto tietyllä sovellusalueella vaatii aina menetelmän soveltamista tapauskohtaisesti.

AVAINSANAT: menetelmä, rakenteinen dokumentti, asiakirja-analyysi

ABSTRACT

Jauhiainen, Raita Eliisa Ursula

Adapting RASKE methodology: Findings from two student projects in

University of Jyväskylä/ Eliisa Jauhiainen

Jyväskylä: University of Jyväskylä, 2005.

134 p.

Master's Thesis

This study is focused on information system methods and describes the RASKE methodology, which is used for document management development. A method may be characterized as a guideline, which provides rules to follow when developing information systems. This study first describes a shell model for analyzing methods. The empirical part of the study examines how RASKE methodology was used in two student projects in the University of Jyväskylä.

The aim of the study is to describe how the use of the RASKE methodology succeeded in selected student projects. The results of the case studies suggest that RASKE methodology can be adapted by someone who has not used it before. In addition, the RASKE methodology was considered useful in the selected projects. The results also support the finding of a previous study: the implementation of a method in certain domain requires some kind of appliance of the method.

KEYWORDS: method, structured document, document analysis

SISÄLLYS

1 JOHDANTO.....	6
2 MENETELMÄT TIETOJÄRJESTELMIEN KEHITTÄMISESSÄ.....	11
2.1 MENETELMÄT JA MALLIT	11
2.2 SIMPUKANKUORIMALLI	13
2.3 MENETELMIEN KÄYTTÄMISESTÄ	17
3 RASKE-MENETELMÄ.....	21
3.1 TAUSTAA.....	21
3.2 ASIAKIRJA-ANALYYSI	23
3.3 MALLINTAMINEN.....	26
3.4 RASKE-MENETELMÄN ARVIOINTIA SIMPUKANKUORIMALLIN MUKAAN.....	30
3.5 MALER & EL ANDALOUSSIN MENETELMÄN ARVIOIMINEN SIMPUKANKUORIMALLIN MUKAAN	32
3.6 RASKE-MENETELMÄN JA MALER & EL ANDALOUSSIN MENETELMÄN VERTAILUA...	41
4 OPISKELIJAPROJEKTIT	45
4.1 TAUSTAA.....	45
4.2 TAG2IT-PROJEKTI	48
4.3 TAG2IT-PROJEKTIN ETENEMINEN	52
4.4 XOOZOO-PROJEKTI.....	55
4.5 XOOZOO-PROJEKTIN ETENEMINEN	58
5 RASKE-MENETELMÄN SOVELTAMINEN OPISKELIJAPROJEKTEISSA	63
5.1 PROJEKTEISSA LAADITTUJEN MALLIEN VERTAILUA	63
5.2 PROJEKTIEN MALLIT VERRATTUNA RASKE-MENETELMÄÄN	74
5.3 YHTEENVETOA MALLEISTA.....	77

6 RASKE-MENETELMÄN KÄYTTÖ JA OMAKSUMINEN	
OPISKELIJAPROJEKTEISSA	80
6.1 RASKE-MENETELMÄN VALINTA JA MENETELMÄÄN TUTUSTUMINEN	81
6.2 HAASTATELTAVAT JA RASKE-MENETELMÄN OMAKSUMINEN	83
6.3 RASKE-MENETELMÄN MALLIT	86
6.4 RASKE-MENETELMÄ PROJEKTITYÖSKENTELYSSÄ	91
7 JOHTOPÄÄTÖKSIÄ.....	94
8 YHTEENVETO	101
LÄHDELUETTELO.....	105
LIITE 1. HAASTATTELUKYSYMYKSET.....	114
LIITE 2. TAG2IT-PROJEKTIN KONTEKSTIKAAVIO	119
LIITE 3. XOOZOO-PROJEKTIN KONTEKSTIKAAVIO	121
LIITE 4. TAG2IT-PROJEKTIN SYÖTE/TUOTOS-MALLI	123
LIITE 5. XOOZOO-PROJEKTIN SYÖTE/TUOTOS-MALLI.....	124
LIITE 6. TAG2IT-PROJEKTIN TILASIIRTYMÄKAAVIO	127
LIITE 7. XOOZOO-PROJEKTIN TILASIIRTYMÄKAAVIO.....	128
LIITE 8. TAG2IT-PROJEKTIAIHEEN KUVAUSLOMAKE.....	129
LIITE 9. XOOZOO-PROJEKTIAIHEEN KUVAUSLOMAKE.....	132

1 JOHDANTO

Dokumentti on ihmisen tulkittavaksi tarkoitettu tietoyksikkö (Sutton 1996). Dokumentteilla on merkittävä rooli organisaatioiden toiminnoissa, koska dokumentteihin on tallennettu organisaatioiden kollektiivinen tietämys. Nykyään organisaatioiden dokumentit ovat ainakin jossain vaiheessa digitaalisia ja toisinaan niitä myös säilytetään pysyvästi digitaalisessa muodossa. (Lehtinen 1998) Siten suurissa organisaatioissa organisaation toiminnan kannalta keskeiset tiedot on tallennettuina digitaalisten dokumenttien kokoelmiin (Salminen 1996).

Sisällönhallinta on erilaisten sisältökokonaisuuksien hallintaa, kuten dokumenttien tai websivustojen hallintaa (Honkaranta 2003). Tarkemmin määriteltynä sisällönhallinnasta voidaan erotella ainakin kolme lähestymistapaa, joita ovat webin sisällönhallinta, dokumenttien hallinta sekä rakenteisten dokumenttien hyödyntäminen (Honkaranta & Tyrväinen 2005).

Tämän tutkielman kannalta olennaisin osa sisällönhallintaa on dokumenttien hallinta sekä erityisesti rakenteisten dokumenttien hyödyntäminen. Suttonin (1996) mukaan *dokumenttien hallinta* on organisaation dokumenttien valvontaa. Päivärinta (2001) on väitöskirjassaan korostanut *elektronisten dokumenttien hallintaa (EDM)*, kun kyse on digitaalisessa muodossa olevien dokumenttien luomisesta, varastoinnista, organisoinnista ja muokkaamisesta. Menetelmiä ja teknikoita, joilla dokumentteja hallitaan, tarvitaan kuitenkin yhä enemmän (Lyytikäinen 2004, 13). Erityisesti organisaatioita hyödyntäisivät sellaiset dokumenttien hallintamenetelmät, joissa dokumenttien hallintaa kehitetään organisaation näkökulmasta teknologianäkökulman sijaan. (Päivärinta 2001) Tarve tällaisille menetelmille on olemassa, sillä dokumenttien hallinta on organisaation väline

liiketoiminnan prosessien tukemiseen teknisen edistyksellisuuden sijaan (Honkaranta 2003).

Perinteisesti tietojärjestelmien tutkimuksessa dokumentit on nähty jäsentymättömän tiedon yksiköinä, kun taas tietokantoihin tallennettu tieto on ollut niin sanottua jäsentynyttä tietoa. Tämän erottelun johdosta dokumenttien hallinnan tutkimus tietojärjestelmätieteiden alalla on pyrkinyt korostamaan sitä, että myös tämän niin sanotun "jäsentymättömän tiedon" hallinta organisaatioissa on tärkeää, koska tutkimusten mukaan jopa 80 - 90% organisaation tiedosta on tallennettu dokumentteihin. (Päivärinta 2001) Merkkaukielten avulla dokumentteihin saadaan kuitenkin lisättyä "tietoa tiedosta", jolloin dokumentteihin tallennetun tiedon hyödyntäminen tehostuu huomattavasti (Maler & El Andaloussi 1996).

Kun dokumenteille laaditaan sekä tietokoneen että ihmisen tulkittavissa oleva rakennemäärittely, voidaan puhua *rakenteisista dokumenteista*. Kun rakenteisia dokumenttistandardeja kehitetään, puhutaan *rakenteistamisesta*. (Tiitinen, Päivärinta, Salminen & Lyytikäinen 1997) Rakenteinen dokumentti on usein joko SGML- tai XML-dokumentti. (Salminen, Tiitinen & Lyytikäinen 1999)

SGML-kieli (Goldfarb 1990) eli Standard Generalized Markup Language on rakenteisten tekstidokumenttien määrittelemiseen ja esittämiseen käytetty kansainvälinen standardi. *XML-kieli* (Bray, Paoli, Sperberg-McQueen & Maler 2000) eli Extensible Markup Language kehitettiin SGML-kielestä tukemaan heterogeenista tietoa sisältävien dokumenttien hallintaa sekä helpottamaan eri sovelusten välistä kommunikaatiota. SGML-kieleen verrattuna XML-kieli on selvästi suppeampi ja se on suunniteltu Internetiä silmällä pitäen. (Salminen 2005)

XML-kielen avulla voidaan rakenteisten dokumenttien sisältämä tieto määritellä. XML-merkkauksen avulla osoitetaan dokumentin looginen rakenne ja nimeään loogisen rakenteen osat. (Salminen 2003) Looginen rakenne kuvataan dokumenttityypin määrittelyssä (DTD), jossa joukko elementtejä on määritelty hierarkkiseen järjestykseen toisiinsa nähden. XML-kielen nimetyillä *elementeillä* kuvataan sisällön rakenne ja elementteihin liittyvillä attribuuteilla voidaan lisätä elementteihin tietoa. (Honkaranta 2003) Esimerkiksi sisältö-nimisellä elementillä voi olla lapsielementteinään kappale-, lista-, ja taulukko-elementit. Näistä kappale-elementin attribuutteina voivat olla muun muassa normaali, kursivoitu ja lihavoitu-arvot, jotka siis määrittelevät sen, millaisesta tekstikappaleesta on kysymys.

Dokumenttien rakenteistamisesta on hyötyä organisaatiolle, sillä rakenteisten dokumenttien sisällöstä tulee monikäyttöistä, kun dokumentin sisältö on erotettu dokumentin ulkoasusta. Esimerkiksi yhdelle tekstisisällölle voidaan laatia useita eri ulkoasuja sen julkaisutavan mukaan. Myös laite- ja sovellusriippumattomuus dokumenttien hallinnassa lisääntyvät. (Tiitinen, Päivärinta, Salminen & Lyytikäinen 1997)

Menetelmä on systemaattisesti etenevä ja ennalta määrätty ohjeistus, jonka avulla jokin tietojärjestelmä tai sen osa kehitetään (Tolvanen 1998). Jotta rakenteisten dokumenttien eri mahdollisuuksia voitaisiin arvioida, tarvitaan arvioinnin helpottamiseksi menetelmiä, joilla tunnistetaan uusien järjestelmien ja sovellusten mahdolliset ongelmat, haasteet sekä mahdollisuudet. (Salminen, Tiitinen & Lyytikäinen 1999) Dokumenttien rakenteistamisprojekteissa käytettävän menetelmän tulisi sisältää asiakirja-analyysia. Asiakirja-analyysin tulisi kattaa sekä rakenteistamisen kohteena olevat dokumentit että dokumenttien hallinnan käy-

tänteiden tutkimisen ja kuvailun. (Salminen 2003, 916) Yksi tällainen on RASKE-menetelmä (Salminen 2003). RASKE-menetelmällä voidaan kuvata elektronisten dokumenttien hallinnan kohdeympäristöä ja sen avulla voidaan suunnitella siihen parannuksia (Lyytikäinen, Tiitinen & Salminen 2000).

Tässä tutkielmassa on tarkoituksena määrittää menetelmä tietojärjestelmätieteen näkökulmasta sekä esittää malli, jolla menetelmiä voidaan arvioida. Tutkielmassa esitellään RASKE-menetelmä sekä Maler & El Andaloussin menetelmä, jotka sisältävät asiakirja-analyysia. Tutkielman empiirinen osuus on tapaus-tutkimusta (case study) kahdesta Jyväskylän yliopiston opiskelijaprojektista. Yinin (1989) mukaan tapaus-tutkimus on empiirinen tutkimus, joka tutkii ilmiötä sen tosiasiallisessa esiintymiskontekstissa. Tapaus-tutkimuksen ohessa on usein suoritettava myös kirjallisuustutkimus. Kirjallisuustutkimus on keino päästä tutkimaan itse aihealuetta paremmin, vaikkei se olekaan tutkimuksen varsinainen tulos. Tässä tutkielmassa tapaus-tutkimusten kautta kartoitetaan, kuinka RASKE-menetelmää sovellettiin Jyväskylän yliopiston opiskelijaprojek-teissa lukukaudella 2004 - 2005. Tutkielman kirjallisia tietolähteitä ovat tieteelli-set artikkelit ja julkaisut, sekä opiskelijaprojektien laatimat raportit. Tutkimuk-sen empiirisessä osuudessa on suoritettu sekä opiskelijaprojektien tuottamien RASKE-menetelmää soveltavien mallien vertailua että projekteihin osallistu-neiden opiskelijoiden haastatteluja.

Tämän tutkielman tutkimusongelma voidaan esittää kysymyksen muodossa seuraavasti: Miten RASKE-menetelmän mallien soveltaminen onnistui Jyväsky-län yliopiston opiskelijaprojekteissa? Tutkimusongelman osaongelmiin haetaan vastauksia seuraavista kysymyksistä: Millainen on RASKE-menetelmä? Onko RASKE-menetelmä käytettävissä sitä ennen tuntemattomille? Ovatko RASKE-

menetelmän asiakirja-analyysin aikana laadittavat mallit helposti omaksuttavissa? Millaisia kokemuksia RASKE-menetelmän soveltamisesta opiskelijaprojekteista saatiin?

Tutkielman luvussa 2 tutustutaan tietojärjestelmätieteen menetelmiin ja niiden arviointiin simpukankuorimallin avulla, sekä selvitetään kuinka menetelmiä käytetään. Luvussa 3 perehdytään RASKE-menetelmään, erityisesti sen asiakirja-analyysivaiheeseen ja vaiheen malleihin. Luvussa myös esitellään RASKE-menetelmää simpukankuorimallin avulla. Luvussa 4 esitetään tutkielman kannalta olennaiset opiskelijaprojektit. Luvussa 5 kerrotaan tapaustutkimuksen kulku ja tiedonhankintatavat sekä RASKE-menetelmän soveltamisesta opiskelijaprojekteissa ja luvussa 6 analysoidaan opiskelijoiden haastatteluista saatuja tietoja, koskien asiakirja-analyysin malleja ja niiden soveltamista. Luku 7 kokoaa tapaustutkimuksen, sekä RASKE-menetelmän soveltamisesta että sen omaksumisesta opiskelijaprojekteissa ja luvussa 8 on koko tutkielman kokoava yhteenveto.

Tutkielman keskeisenä tuloksena on tietoa RASKE-menetelmän soveltuvuudesta sitä ennen tuntemattomille tietojärjestelmätieteilijöille sekä esimerkkejä siitä, kuinka RASKE-menetelmää voidaan soveltaa. Erityisesti tietoa saadaan RASKE-menetelmän aikana tehtävistä malleista, niiden laatimisen helppoudesta sekä mallien selkeydestä. Tutkielmasta voi olla hyötyä RASKE-menetelmää kehittäville tutkijoille, koska aikaisemmin menetelmää ovat soveltaneet vain ne tahot, jotka ovat olleet mukana menetelmän kehitystyössä. Mikäli RASKE-menetelmää tullaan joskus käyttämään yliopiston ulkopuolella, esimerkiksi yrityksissä, voivat tämän tutkimuksen tulokset olla hyödyllisiä niille, jotka toimivat dokumenttien hallinnan kehittämisprojekteissa.

2 MENETELMÄT TIETOJÄRJESTELMIEN KEHITTÄMISESSÄ

Tietojärjestelmätieteessä menetelmiä käytetään tietojärjestelmien analyysiin, suunnitteluun ja kehittämiseen. Menetelmiä alettiin käyttää 1970-luvun lopulla, jolloin luotiin ns. *vesiputousmalli* kuvaamaan tietokonesovellusten kehitystä. Vesiputousmallin käytössä oli kuitenkin rajoitteita ja näiden rajoitteiden purkamiseksi alettiin kehittää uusia menetelmiä. (Avison & Fitzgerald 2003)

Tässä luvussa kerrotaan mitä menetelmällä tarkoitetaan, mikä on menetelmän ja mallin suhde toisiinsa, mistä menetelmät koostuvat ja kuinka menetelmiä käytetään. Menetelmätietämyksen viitekehys perustuu Tolvasen (1998) esittämään malliin.

2.1 Menetelmät ja mallit

Menetelmät ovat tärkeässä roolissa tietojärjestelmien kehityksessä. Menetelmiä on kehitetty kahdesta eri syystä. Ensimmäinen menetelmien katsotaan antavan ohjeistuksen tietojärjestelmien suunnittelu- ja mallinnustyöhön. Toiseksi menetelmien katsotaan rajaavan tietojärjestelmäsuunnittelijoiden tarpeet. (Nance & Arthur 1988). Menetelmät siis tukevat tietojärjestelmien kehitystyötä, ja niiden käytöllä pyritään estämään järjestelmäsuunnittelun kaventumista pelkäksi ohjelmointityöksi (Eriksson & Penker 2000). Siksi menetelmät ovat merkittävä osa sekä tietojärjestelmien kehitystä että niiden tukimusta (Nance & Arthur 1988).

Kun tietojärjestelmää suunnitellaan ja kehitetään, järjestelmien toimintoja tai rakennetta kuvataan malleilla. *Mallit* ovat suunnittelutyölle asetettujen vaati-

musten mukaisia kuvauksia tietojärjestelmistä tai sen osista. (Erikkson & Penker 2000)

Menetelmä on jokin tapa järjestellä ajatuksia tai toimintoja. Se kertoo, kuinka jokin asia tehdään, milloin se tehdään ja miten se tehdään. (Erikkson & Penker 2000) Menetelmä koostuu joukosta ennalta määrättyjä tekniikoita, joille löytyy säännöt siihen, missä järjestyksessä ja millä tavalla näitä tekniikoita käytetään, jotta menetelmää käyttäen saavutettaisiin suunniteltava tulos (Tolvanen 1998).

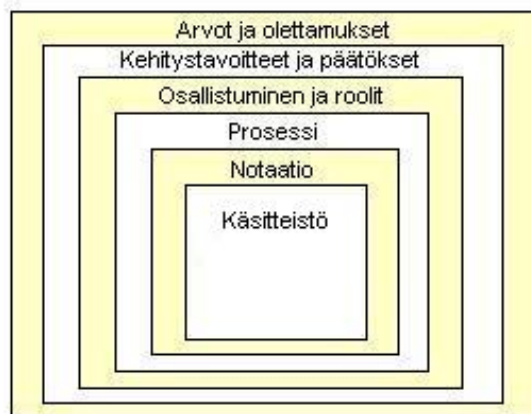
Mallit ovat osa menetelmiä. Menetelmä ei vain kuvaa sitä, kuinka malleja laaditaan, vaan myös sen, kuinka mallit menetelmässä jäsennetään. Mallien tarkoituksena on kuvata menetelmän seuraamisesta syntyviä tuloksia. Malli kuvataan *mallinnuskielen* (modelling language) avulla, joka määrittelee sen millainen malli on ja kuinka mallia käytetään. Kun menetelmässä on toimintaohjeita (processes) tuloksen aikaansaamiseksi, mallinnuskielestä ne puuttuvat. (Erikkson & Penker 2000)

Erilaisia menetelmiä on kehitetty paljon ja siten myös menetelmien merkintätapoja eli notaatioita on lukuisia. *UML-mallinnuskieli* eli Unified Modelling Language yhtenäistää useita eri mallinnuskieliä ja se on muodostunut mallinnuskielten standardiksi. (Erikkson & Penker 2000)

2.2 Simpukankuorimalli

Menetelmiä voidaan analysoida ja lajitella niiden *rakenteen*, *sisällön* ja *käytön* suhteen. Kun nämä kolme tekijää yhdistetään, voidaan puhua *menetelmätietämyksestä* (method knowledge). (Tolvanen 1998)

Tolvanen (1998) on esitellyt väitöskirjassaan simpukankuorimallin (shell model), jonne on koottu menetelmätietämyksen eri tasot. Mallin (KUVIO 1) muoto korostaa sitä, että menetelmät rakentuvat eri tekijöistä, jotka ovat toisiaan täydentäviä. Jokaista mallissa kuvattua tasoa tarvitaan, jotta voitaisiin puhua eheästä menetelmästä (Tolvanen 1998). Mallissa neliö kuvastaa menetelmätietämyksen osia ja neliöiden sisäkkäisyys niiden välisiä suhteita.



KUVIO 1. Simpukankuorimalli (Tolvanen 1998, 35)

Kukin simpukankuorimallin osa esitetään seuraavissa tekstikappaleissa mallin sisimmästä osasta alkaen. Esittely pohjautuu Tolvasen (1998) väitöskirjaan.

Käsitteistö

Kaikki menetelmät perustuvat johonkin *käsitteistöön* (conceptual structure). Käsitteistö pitää sisällään menetelmässä esiintyvät käsitteet, niiden rajoitteet sekä käsitteiden väliset suhteet. Menetelmän käsitteistö voi siis kattaa esimerkiksi ne symbolit, joita menetelmässä käytetään, tai tavat, joilla jokin prosessi kuvataan.

Menetelmien käsitteistöt eroavat yleensä paljon toisistaan. Tämä hajanaisuus johtuu muun muassa siitä, että menetelmiä käytetään eri kohdealueilla. Menetelmän käsitteistö onkin usein sovellus- tai toimialakohtainen. Menetelmän käsitteistö muodostaa perustan menetelmän muille osille. Siksi suuri osa menetelmien tutkimuksesta keskittyy pääasiassa juuri menetelmien käsitteistöjen tutkimukseen.

Notaatio

Menetelmän käsitteistö on esitettävissä vain käyttämällä jotakin mallinnustekniikkaa (modelling technique). *Mallinnustekniikoilla* muodostetaan malleja, jotka ovat aina jonkin notaation mukaisia. Tyypillisiä esimerkkejä mallinnustekniikoista ovat graafiset kaaviot, matriisit ja taulukot.

Notaatio tarkoittaa sääntöä, jonka mukaan menetelmän käsitteistöä mallinnetaan. Notaatio ottaa kantaa siihen, kuinka esimerkiksi kaaviossa esiintyvä luokka esitetään; suorakaiteella vai ympyrällä.

Prosessit

Prosessit ovat menetelmän menettelytapaohjeita. Prosessit määrittelevät sen, kuinka ja missä menetelmän vaiheessa eri mallinnustekniikoita käytetään, jotta saataisiin aikaan halutunlaisia, menetelmän mukaisia malleja.

Prosessien tulisi perustua menetelmän käsitteistöön. Jos menetelmässä esimerkiksi kuvataan tiedon siirtymistä toimijalta toiselle, tulee käsitteistöä löytyä käsitteet sekä tiedolle että siihen liittyville toimijoille. Toisaalta menetelmän käsitteistö voi rajoittaa menetelmän käyttämisprosessin etenemistä. Esimerkiksi jos käsitteistöä ei löydy käsitettä *periytyminen*, ei menetelmässä voida kuvata olion ominaisuuksien periytymistä millään kaaviolla.

Osallistuminen ja roolit

Useissa menetelmissä oletetaan, että menetelmää käyttävät tai sen vaikutuspiirissä ovat ainoastaan tietojärjestelmien suunnittelijat. Jotkin menetelmät pyrkivät kuitenkin kuvaamaan eri sidosryhmien osallistumista menetelmän käyttöön, sekä eri rooleja, joita menetelmän käyttöön kuuluu. Eri rooleja voivat esimerkiksi olla suunnittelijat, ohjelmoijat ja loppukäyttäjät. Suuri osa menetelmistä kuitenkin jättää roolit huomioimatta.

Ne menetelmät, joissa määritellään rooleja, on yleensä suunniteltu johonkin tiettyyn tietojärjestelmäkehitystyöhön, jossa loppukäyttäjien tai muun vastavan ryhmän osallistuminen on tärkeää suunnittelutyön tulokselle.

Kehitystavoitteet ja -päätökset

Menetelmän *kehitystavoitteet* (development objectives) kertovat millaisia toteutusratkaisuja menetelmässä pidetään hyvinä. Sen sijaan menetelmän *kehityspäätöksillä* kuvataan sitä, kuinka halutut kehitystavoitteet voidaan saavuttaa. Siten menetelmien kehitysvalinnat ja kehityspäätökset tulisivat olla läheisesti yhteydessä menetelmän prosesseihin, notaatioon sekä käsitteisiin. Esimerkkinä kehitystavoitteesta voisi olla *"dokumenttien sisältöjä sekä ulkoasuja tulisi yhtenäistää."* Vastaavasti kehityspäätös voisi olla *"organisaation kokousasiakirjat muutetaan XML-tiedostoiksi ja niille määritellään skeemat."*

Menetelmät voivat erota suuresti juuri kehitystavoitteidensa osalta. Kaksi samankaltaista, samalle sovellusalueelle kehitettyä menetelmää voivat olla lähes identtisiä keskenään, mutta niistä toinen keskittyy kuvaamaan muuttumatonta prosessia, kun taas toinen pyrkii muuttamaan sitä.

Arvot ja olettamukset

Menetelmät perustuvat aina johonkin olemassa olevaan olettamukseen siitä, mitä mallinnettava todellisuus on. Nämä olettamukset eivät kuitenkaan välttämättä tule menetelmästä selkeästi esille. Menetelmän kehitystavoitteiden ja arvojen välillä on yleensä eroavaisuus, joka on tärkeää tiedostaa. Kaksi eri menetelmää voi pyrkiä samaan kehitystavoitteeseen, mutta eri tavoin. Todellisuudessa useat menetelmät eivät määrittele selkeästi taustalla olevia olettamuksia. Monista menetelmistä olettamusten tunnistaminen uupuu kokonaan.

Yhteenveto

Simpukankuorimallin mukaan menetelmät perustuvat käsitteistöön, joka sisältää joukon erilaisia käsitteitä ja käsitteiden välisiä suhteita. Näitä käsitteitä käy-

tetään mallinnustekniikoilla, joilla kuvataan malleja jonkin notaation mukaan. Menetelmän käytön tulee perustua käsitteisiin ja menetelmän etenemisprosessi kuvaa, kuinka menetelmän mallit luodaan, kuinka niitä muokataan ja kuinka niitä käytetään menetelmän notaation mukaan. Eri sidosryhmät analysoivat ja korjaavat näitä käsitteitä sekä niiden ilmentymiä. (Tolvanen 1998)

Simpukankuorimalli painottaa menetelmän eri osien välisiä riippuvuuksia. Olennaisin riippuvuus eri menetelmän osien väliltä löytyy kuitenkin käsitteistön ja notaation väliltä, koska ilman tätä riippuvuutta sekä mallinnustekniikoiden kehitystyö että niiden käyttäminen olisi hyvin vaikeaa. (Tolvanen 1998)

2.3 Menetelmien käyttämisestä

Motivaatiot menetelmien käyttöönottamiselle vaihtelevat organisaatioittain, mutta yleisesti menetelmiä käytetään, jotta saataisiin aikaan parempia, loppukäyttäjien vaatimuksia vastaavia tuotteita sekä tehostettua järjestelmäkehitystä yhdenmukaistamalla työprosesseja. (Avison & Fitzgerald 2003) Yleisin tapa, jolla organisaatio valitsee käytettävän menetelmän, lienee ”yrityksen ja erheen” kautta. Organisaatiot voivat valita menetelmän myös konsultin suosituksesta tai epäsuorasti siten, että organisaatio valitsee ensin jonkin työvälineen, joka tukee jonkin tietyn menetelmän käyttöä. (Tolvanen 1998)

Vaikka menetelmiä on paljon, ei ole itsestään selvää, että menetelmät olisivat yleisesti käytettyjä tai edes hyödyllisiä tietojärjestelmien kehitystyössä. Empiiriset tutkimukset ovat osoittaneet, että useat yritykset eivät käytä menetelmiä lainkaan. (Tolvanen 1998) On esitetty useita syitä siihen, miksi menetelmien

systemaattinen käyttöönotto on kokenut takaiskuja. Yleisin syy lienee kuitenkin huono tuottavuus menetelmän käytöstä huolimatta. Toiseksi syyksi on epäilty menetelmien liiallista monimutkaisuutta. (Avison & Fitzgerald 2003) Menetelmä voidaan kokea tietyissä tapauksissa myös raskaiksi ja joustamattomiksi (Kalermo & Rissanen 2002)

Tolvasen (1998) mukaan menetelmien käyttäminen voi olla systemaattista, tilapäistä tai menetelmiä ei käytetä lainkaan. Yksi sovellusalue, jossa menetelmien käyttö on vähäistä, on web-pohjaisten sovellusten kehittäminen. Web-pohjaisten sovellusten kehittämisessä luotetaan usein vain muutamien ihmisten taitoihin ja kokemuksiin. Siten näiden sovellusten kehitystyö on usein luonteeltaan virheiden kautta oppimista. (Avison & Fitzgerald 2003)

Ennen kuin menetelmän käyttämisen tasoja voidaan luokitella, on määriteltävä mitä menetelmän käyttämisellä tarkoitetaan. Olennaisinta olisi pohtia sitä, miten tarkasti jotain menetelmää tulisi seurata, jotta voitaisiin sanoa, että menetelmä on käytössä.

Kuinka menetelmien käyttöä voidaan sitten tutkia? Yksi vaihtoehto menetelmien käytön tutkimiseen on tutkia empiirisesti menetelmien käyttöä todellisissa ympäristöissä, kuten jossain organisaatiossa. Menetelmien käyttöä voidaan myös tutkia keskittymällä mekanismeihin, jotka tukevat ns. paikallisten menetelmien (local method) kehittämistä ja käyttöä. (Tolvanen 1998)

Paikallisilla menetelmillä tarkoitetaan organisaation omaa muunnelmaa olemassa olevasta menetelmästä. Tutkimukset ovat osoittaneet, että paikalliset menetelmät ovat jopa suositumpia kuin niiden kaupalliset vastineet. Usein menetelmän

muokkaaminen tarkoittaa olemassa olevan menetelmän yksinkertaistamista tai sen yhdistelemistä toiseen menetelmään. Menetelmien muokkaaminen voi johtaa myös täysin uuden menetelmän kehittämiseen. (Tolvanen 1998)

Tolvasen (1998) mukaan paikallisten menetelmien kehitystyö jakautuu viiteen eri vaiheeseen. Näitä vaiheita ovat

1. menetelmien valinta
2. menetelmän muokkaus
3. sovelluksen valinta ja omaksuminen
4. uusien menetelmien esittely
5. menetelmän käyttö.

Ensimmäisessä vaiheessa organisaatiot valitsevat menetelmät, joita he aikovat käyttää. Toisessa vaiheessa valittuja menetelmiä muokataan tai organisaatio päätyy kehittämään tarpeisiinsa täysin uuden menetelmän. Kolmannessa vaiheessa organisaatio valitsee käyttöönsä sellaiset sovellukset, joiden avulla menetelmän käyttö onnistuu. Neljännessä vaiheessa uutta paikallista menetelmää testataan sekä sitä mahdollisesti pilotoidaan. Kun testaus on saatu päätökseen, organisaatio ottaa menetelmän ja siihen sopivat sovellukset käyttöönsä. Paikallisen menetelmän käyttöönotto ei kuitenkaan välttämättä noudata annettua viittä vaihetta sellaisenaan vaan osa vaiheista voi käytännössä tapahtua samanaikaisesti. (Tolvanen 1998)

Tolvasen (1998) mukaan paikallisten menetelmien käytössä keskeisenä tekijänä on organisaation tai projektin oppiminen menetelmän käyttöön. Kun menetelmän käyttö on opittu, sen soveltuvuudesta omaan organisaatioon tai projektiin saadaan tietoa ja tämän kautta menetelmä voi jalostua entisestään vastaamaan

niihin tarpeisiin, joihin menetelmää käytetään. Paikallisen menetelmän kehittäminen kuitenkin vaatii organisaatiolta merkittävästi resursseja, joka voitaisiin välttää käyttämällä jo olemassa olevia menetelmiä. Tästä huolimatta on kuitenkin havaittu, että tarve paikallisten menetelmien kehittämiseksi ja käyttöönotolle on niin suuri, että esimerkiksi suuret taloudelliset menoerät eivät ole paikallisten menetelmien käyttöönottoa organisaatioissa estäneet.

Paikallisten menetelmien lisäksi organisaatioissa voidaan ottaa käyttöön ketterä menetelmä (agile method). *Ketterällä menetelmällä* tarkoitetaan menetelmää, jonka painopiste on ohjelmointityön läpiviemisessä tunnollisen dokumentoinnin sijaan. Ketterät menetelmät pohjautuvat *Agile Manifestoon*, joka on kokoelma ketterän ohjelmistokehityksen arvoja ja periaatteita. Agile Manifesto esitettiin vuosituhannen vaihteessa vastatoimenpiteenä perinteisille ja ankarille ohjelmistokehitysmenetelmille ja prosessimalleille. Agile Manifesto sisältää 12 periaatetta, jotka ketterien menetelmien tulisi täyttää. (Kalermon & Rissanen 2002)

Kalermon ja Rissanen (2002) mukaan ketterä menetelmä sopii erityisesti projekteihin, joissa kehitetään uusia ja innovatiivisia ohjelmistotuotteita tiukkojen aikataulujen puitteissa. Ketterä menetelmä sopisi siten myös pienille organisaatioille, joiden liiketoiminta perustuu innovatiivisille ohjelmistotuotteille. Sen sijaan suuret organisaatiot käyttävät todennäköisemmin perinteisempiä menetelmiä. Esimerkkinä ketterästä menetelmästä on Extreme Programming -menetelmä.

3 RASKE-MENETELMÄ

Tehokas dokumenttien hyödyntäminen edellyttää dokumenttien standardointia (Lehtinen 1998). *Dokumenttien standardointi* tarkoittaa sopimusta säännöistä, jotka määrittelevät sen, miten tietoa dokumenteissa esitetään ja minkä käytänteiden mukaan dokumentteja laaditaan. Yhteisiä ja yhdenmukaisia sääntöjä tarvitaan, jotta saataisiin aikaiseksi tehokkaampia ja vakaampia tapoja jakaa dokumentteihin tallennettua tietoa. Jotta dokumentit ja dokumenttien hallinnan käytänteet saadaan yhtenäistettyä, tarvitaan syvällistä *asiakirja-analyysiä*. (Salminen, Tiitinen & Lyytikäinen 1999)

Tässä luvussa kerrotaan RASKE-menetelmästä, joka on kehitetty asiakirja-analyysin tueksi (Salminen 2003). Luvussa kerrotaan RASKE-menetelmän mukaisista malleista, joilla kuvataan kohdealueen rooleja, prosesseja ja dokumentteja. RASKE-menetelmää myös arvioidaan hyödyntäen luvussa 2 esitettyä simpukankuorimallia.

3.1 Taustaa

RASKE-projekti alkoi vuonna 1994 Jyväskylän yliopiston, eduskunnan ja valtioneuvoston yhteistyönä. Projektin tavoitteena oli asiakirjastandardien määrittäminen siten, että standardit soveltuisivat pitkäaikaiseen sähköiseen arkistointiin ja jossa asiakirjojen tietorakenteet olisi esitetty yhdenmukaisella tavalla. (Tiitinen, Salminen & Lyytikäinen 1997)

RASKE-projektissa kehitettiin olioperustainen *asiakirja-analyysi* rakenteistamisen tarpeisiin. Analyysin tarkoituksena oli auttaa kohdealueen käsitteiden loogisessa kuvauksessa sekä löytämään kohdealueen dokumenttien hallintaan liittyviä ongelmia. Syntyi *RASKE-menetelmä* ("RAkenteisten AsiakirjaStandardien KEhittämien"), joka vastasi dokumenttien standardoinnin tarpeisiin ottamalla huomioon mm. dokumenttien laatijoiden roolin dokumenttien elinkaareissa. Menetelmä jaettiin viiteen päävaiheeseen, joita olivat

- asiakirja-analyysi,
- suunnittelu,
- testaus ja evaluointi,
- toteutus sekä
- jatkuvan kehityksen käynnistäminen. (Lehtinen 1998, 10)

RASKE-projektin analysoituja kohdealueita olivat Suomen lainsäädännön valmisteluasiakirjat (1994 - 1995), EU-lainsäädäntöasiakirjat Suomessa (1996) sekä Suomen valtion talousarvioon liittyvät asiakirjat (1996). (Salminen, Lyytikäinen, Tiitinen & Mustajärvi 2001) RASKE-projekti päättyi vuonna 1998, mutta sitä seurasi useita muita projekteja, joissa eduskunta ja eräät ministeriöt toteuttivat SGML-ratkaisuja. (Salminen, Lyytikäinen & Tiitinen 2000)

RASKE-projektin käynnistyessä oli olemassa kaksi kansainvälistä standardia tiedon määrittämiseen ja esittämiseen rakenteisissa dokumenteissa. Nämä standardit olivat SGML (Goldfarb 1990) ja ODA (Open Document Architecture, ISO-8613 1989). Näitä standardeja verrattaessa päädyttiin ehdottamaan SGML-kielen käyttöönottoa. SGML-kieltä pidettiin soveliaampana sen helposti omak-

suttavan merkkauksen takia. (Salminen, Lyytikäinen, Tiitinen & Mustajärvi 2000)

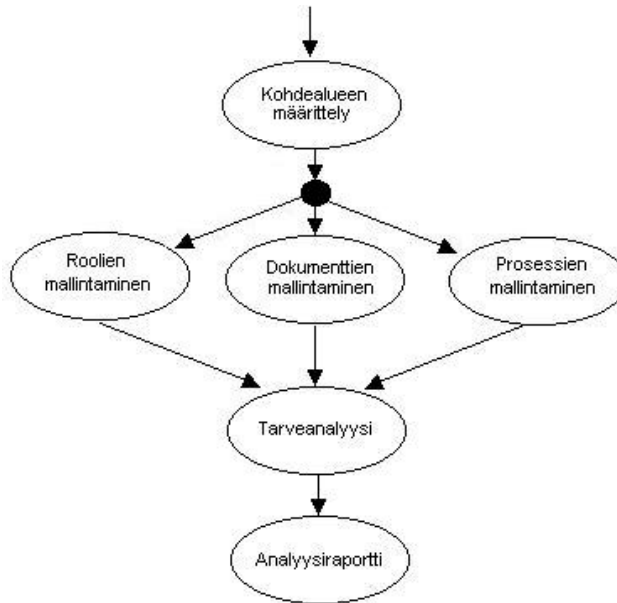
Asiakirjojen standardointia hankaloittivat muun muassa erimuotoisten dokumenttien suuret määrät. Myös se, että dokumentteja oli totuttu käsittelemään pelkästään paperidokumentteina, oli standardoinnin kannalta haastavaa. Lisäksi dokumenttien käyttäjät ja heidän tarpeensa dokumentteja koskien vaihteli suuresti. Siten asiakirjojen standardoiminen ei ollut haaste pelkästään teknisestä näkökulmasta vaan se vaikutti myös suuresti siihen, kuinka kohdealueen henkilöt työskentelivät. Siksi jo projektin alusta alkanut yhteistyö kaikkien asianomaisten kanssa oli välttämätöntä. (Salminen ym. 2000)

3.2 Asiakirja-analyysi

RASKE-menetelmän asiakirja-analyysi keskittyy kohdealueen kuvaukseen. Asiakirja-analyysin tarkoituksena on tarjota kuvausta sekä kohdealueen dokumenteista että niistä toiminnoista ja toimijoista, joita dokumenttien laatimiseen liittyy. Siten asiakirja-analyysi vaatii laajaa tietämystä dokumenttien luomisesta sekä niiden käyttämisestä. Asiakirja-analyysin keskeisimpänä tavoitteena on tarjota alustava ehdotus dokumenteille laadittavista standardeista. (Salminen, Lyytikäinen & Tiitinen 2000)

RASKE-menetelmässä asiakirja-analyysin vaiheen aikana tehtävä analyysiprosessi voidaan jakaa erillisiin vaiheisiin seuraavan kuvion (KUVIO 2) mukaisesti (Salminen, Lyytikäinen & Tiitinen 2000). Kuviossa ellipsit kuvaavat yksittäistä vaihetta ja nuolet kunkin vaiheen alkamisjärjestystä. Kohdealueen määrittelyn-

jälkeen roolien, dokumenttien ja prosessien mallintaminen voidaan suorittaa yhtäaikaista. Kuviossa tätä rinnakkaisuutta on kuvattu mustalla ympyrällä.



KUVIO 2. Analyysiprosessin vaiheet (Salminen, Lyytikäinen & Tiitinen 2000, 628)

Mallin mukaan analyysiprosessi alkaa kohdealueen määrittelyllä, jota seuraavat roolien, dokumenttien ja prosessien mallintaminen. Roolit, dokumentit ja prosessit voidaan määrittellä ja mallintaa samanaikaisesti. Lyytikäisen (2004) mukaan roolien mallintaminen kartoittaa kohdealueen toimijat, kun taas prosessien mallintaminen tuottaa kuvauksia kohdealueen toiminnoista. Tarveanalyysiin kirjataan ne vaatimukset, joita kohdealueen toimijoilla on. Tarveanalyysi voi olla esimerkiksi taulukko, johon on koottu dokumenttien käsittelytilanteisiin liittyviä kehittämistarpeita. Analyysiprosessin tuloksena on raportti, jossa koko analyysiprosessin aikana tuotettu dokumentaatio kootaan yhteen.

RASKE-projektin aikana ei ollut tarvetta mallintaa sovellusalueen tietojärjestelmiä, koska tavoitteena oli kehittää asiakirjoja ja niiden rakenteita, jotka olisivat tekniikka- tai ohjelmistoriippumattomia. Siksi järjestelmien mallintaminen erillisenä vaiheena on jätetty pois RASKE-menetelmän asiakirja-analyysi vaiheesta. Sovellusalueen järjestelmät ja tekniikat voidaan kyllä tarvittaessa kuvata kohdealueen määrittelyn aikana (ks. KUVIO 2, s. 24). (Lyytikäinen 2004)

Kun RASKE-menetelmää kehitettiin, asetettiin käytettäville malleille vaatimuksia. Näitä vaatimuksia olivat:

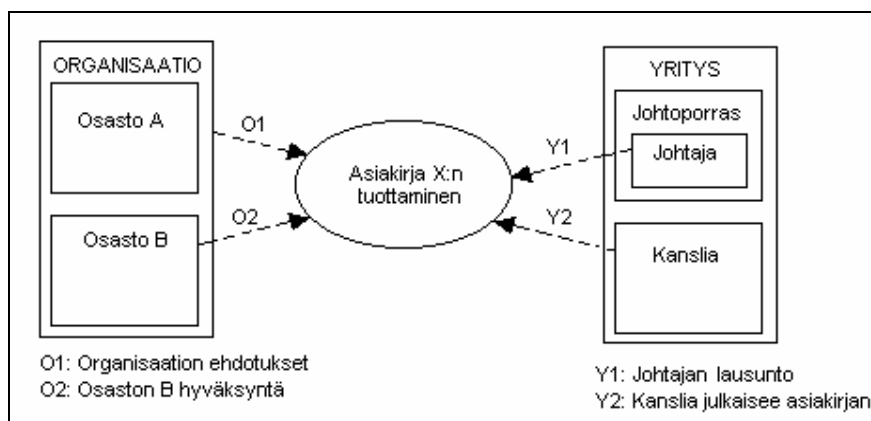
1. Mallien tulee kuvata todellista toimintaa eli sitä, miten työ todellisuudessa tehdään.
2. Mallien tulee olla graafisia ja mahdollisimman selkeitä. Siten niiden notaation tulisi olla yksinkertainen.
3. Malleja ei saa olla liikaa; yhden mallin tulee kuvata mahdollisimman monta olennaista tekijää; toiminnot, toimijat ja dokumentit.
4. Dokumenttien luomista ja käyttöä kuvaavien mallien tulisi erotella tiedon ja toimintojen vuo (Salminen, Lyytikäinen & Tiitinen 2000).

Vaatimusten pohjalta asiakirja-analyysin graafisten mallien pohjaksi valittiin ICN (Information Control Nets,) -mallit (Ellis 1979) sekä OOA (Object Oriented Analysis) -menetelmästä (Shlaer & Mellor 1992) löytyvät tilasiirtymäkaaviot (Salminen, Lyytikäinen & Tiitinen 2000). ICN-mallien notaatiolla mallinnettiin kohdealueen toimintoja ja tietolähteitä. Kohdealueen *toiminnoissa* (activities) tietoa tuotetaan ja sitä käytetään. *Tietolähteet* (resources) ovat tiedon säilytyspaikkoja, jonne tietoa tallennetaan tai josta tietoa saadaan. (Lyytikäinen, Tiitinen & Salminen 2000) Dokumenttien sisällön mallinnukseen päätettiin soveltaa

myös Maler & El Andaloussin (1996) menetelmää, josta kerrotaan tarkemmin luvussa 3.4. Lisäksi RASKE-menetelmässä sovellettiin olioajattelua. Dokumenttien katsottiin olevan ns. dokumenttiolioita. *Dokumenttiolio* määriteltiin olevan dokumenttien joukon abstraktio, johon sisältyi aina jokin sisältö. Sisältö taas rakentui hierarkkisista osista, jotka saattoivat olla esimerkiksi tekstiä, grafiikkaa tai numeerista dataa. (Lehtinen 1998)

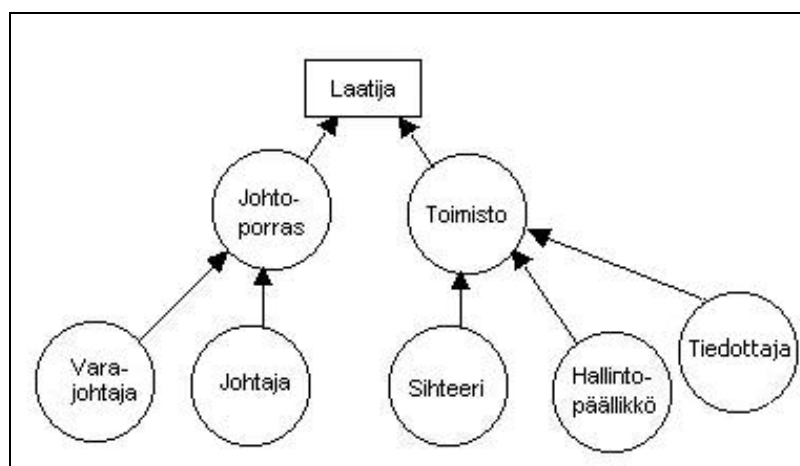
3.3 Mallintaminen

Asiakirja-analyysi aloitetaan kohdealueen yleiskuvauksella, joka mallinnetaan kontekstikaavioon (KUVIO 3). Kaaviossa suorakulmiot kuvaavat kohdealueen organisaatioita, organisaatioiden osastoja tai jopa yksittäisiä toimijoita. Toimijat voidaan ryhmitellä sisäkkäisten suorakaiteiden avulla, jolloin näiden toimijoiden hierarkkinen järjestys voidaan nähdä selkeästi. Organisaatioista ja/tai toimijoista lähtevät merkatut nuolet kertovat toimijoiden rooleista kohdealueella. Nuolet osoittavat mallinnuksen kohteena olevaan toimintoon, joka kuvataan kaavion keskellä olevalla ellipsillä. (Lyytikäinen 2004)



KUVIO 3. Kohdealueen yleiskuvas kontekstikaaviossa

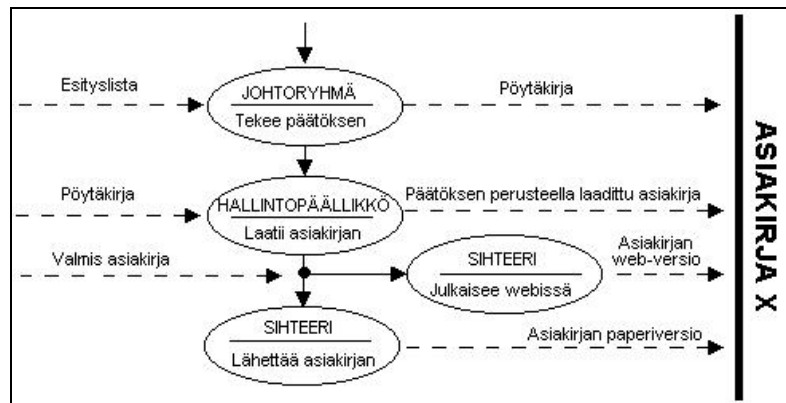
Kohdealueen yleiskuvauksen jälkeen voidaan aloittaa kohdealueen roolien, prosessien ja dokumenttien mallintaminen. Roolit jaetaan laatijoihin ja käyttäjiin. Laatijat ovat dokumenttien sisältöön vaikuttavia henkilöitä ja käyttäjät ovat niitä, jotka käyttävät dokumentteja työssään. Käyttäjät jaetaan edelleen sisäisiin ja ulkoisiin käyttäjiin. Kohdealueen roolit mallinnetaan roolitaulukkoon ja/tai roolimalliin (KUVIO 4). Roolimallissa roolit on merkitty ympyrällä ja rooleja yhdistävät nuolet osoittavat sen, mihin rooliluokkaan tai ryhmään kukin rooli kuuluu. (Salminen 2003) Esimerkiksi kuviossa 4 sihteerin rooli kuuluu toimistorooliluokkaan.



KUVIO 4. Roolimalli

Prosessien mallintaminen tuottaa kuvausta siitä, kuinka kohdealueen dokumentteja luodaan ja käytetään. Prosessit mallinnetaan dokumentin syöte- tai tuotos-malliin. Tarvittaessa nämä mallit voidaan yhdistää syöte/tuotos-malliksi (KUVIO 5). Mallissa ellipsi kuvaa prosessin aktiviteettia. Ellipsin yläosassa kerrotaan aktiviteetin suorittaja. Aktiviteettien väliset nuolet kertovat kunkin aktiviteetin alkamisjärjestyksen prosessissa. Jos nuolet alkavat yhteisestä mustasta ympyrästä, aktiviteetin voidaan suorittaa rinnakkain. Mallissa olevat katkoviiva

taas kuvaavat dokumenttivuota eli niitä syötteitä tai tuotoksia, joita aktiviteetteihin liittyy. Usein nämä syötteet ja tuotokset ovat dokumentteja. Syöte/tuotosmallin notaatio on lainattu ICN-malleista. (Lyytikäinen 2004)



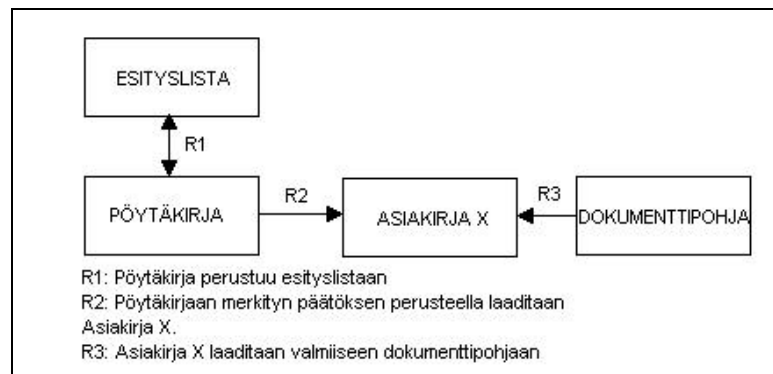
KUVIO 5. Prosessien kuvaaminen Syöte/tuotos-mallissa

Yksittäisen dokumenttiolion elinkaaren kuvaamiseen käytetään tilasiirtymäkaaviota (KUVIO 6). Oletuksena on, että dokumenttioliolla voi olla yhdellä hetkellä vain yksi tila. Tilasiirtymäkaaviossa suorakulmio kuvaa yhtä tiettyä dokumenttiolion tilaa ja nuoli kuvaa siirtymää tilasta toiseen. Jotta dokumenttiolio voisi siirtyä tilasta toiseen, tarvitaan jokin toiminto. Yhdessä tilassa tapahtuu kuitenkin vain yksi toiminto, mutta toiminto voi sisältää useita eri aktiviteetteja. Esimerkiksi kuviossa 5 yksi dokumenttiolion tilaan vaikuttava toiminto on "Hallintopäällikkö kirjaa dokumenttipohjaan tarvittavat tiedot". Tästä toiminnosta seuraa tila "Asiakirja X on laadittu". Tilasiirtymäkaavio on lainattu RASKE-menetelmään OOA-menetelmästä. (Lyytikäinen 2004)



KUVIO 6. Dokumenttiolion elinkaaren kuvaus tilasiirtymäkaaviossa

Dokumenttien suhteiden mallintaminen voidaan kuvata dokumenttisuhte (D-R) -kaaviolla (KUVIO 7), joissa suorakaiteet kuvaavat dokumenttiolioita ja nuolet näiden välisiä suhteita.



KUVIO 7. Dokumenttien mallintaminen D-R -kaaviossa

RASKE-menetelmässä dokumenttien osia, dokumenttikomponentteja, mallinetaan Elm-mallin notaation mukaan, joka on omaksuttu Maler & El Andaloussin menetelmästä. Elm-malliin palataan tarkemmin luvussa 3.5.

3.4 RASKE-menetelmän arviointia simpukankuorimallin mukaan

Erikssonin ja Penkerin (2000) mukaan järjestelmäkehitykseen kuuluu viisi vaihetta, joita ovat vaatimusanalyysi, analyysi, suunnittelu, ohjelmointi ja testaus. Järjestelmäkehitys aloitetaan vaatimusanalyysillä, jonka aikana kartoitetaan asiakkaan vaatimukset kehitettävälle tietojärjestelmälle. Vaatimusanalyysin jälkeen siirrytään analyysivaiheeseen, jonka aikana etsitään olioita ja luokkia, joita kehitettävään järjestelmään kuuluu. Suunnitteluvaiheessa analyysin tuloksia tarkastellaan teknisestä näkökulmasta ja tarvittaessa lisätään uusia luokkia, joita analyysivaiheessa ei vielä havaittu tarpeellisiksi. Ohjelmointivaiheessa suunnittelun tulokset muutetaan lähdekoodiksi ja testausvaihe testaa ohjelmointityön toimivuutta.

Tolvanen (1998) on väitöskirjassaan tutkinut järjestelmäkehitykseen liittyviä menetelmiä. Siten luvussa 2.2 esitelty simpukankuorimalli pyrkii kuvaamaan järjestelmäkehitykseen liittyvää menetelmätietämystä ja sen rakennetta. RASKE-menetelmä on kuitenkin kehitetty erityisesti sisällönhallinnan, kuten dokumenttien hallinnan tai rakenteisten dokumenttien kehittämistarkoituksiin (Lyytikäinen 2003, Honkaranta & Tyrväinen 2005). Luvussa 3.1 esitellyissä RASKE-menetelmän vaiheissa ei siten ole esimerkiksi ohjelmointivaihetta lainkaan, joka on taas olennainen osa järjestelmäkehitystä.

Päivärinnan (2001) mukaan dokumenttien hallinta on osa tietojärjestelmätutkimuksen kenttää ja dokumenttien hallintaan voidaan siten soveltaa yleisiä tietojärjestelmien kehittämismenetelmiä. Tutkija on väitöskirjassaan muodostanut lähestymistavan ja kehittämismenetelmän, jolla voidaan saada aikaan vaatimusmäärittely elektroniselle organisaation dokumenttien hallintajärjestelmälle. Muodostetun lähestymistavan perustana on ollut tietojärjestelmien menetelmiin kehitetty simpukankuorimalli. Vastavuoroisesti RASKE-menetelmää, joka on yksi dokumenttien hallinnan kehittämismenetelmä, voidaan arvioida simpukankuorimallin (ks. luku 2.2, s. 8) mukaan. Kuten luvussa 3.1 sanotaan, on RASKE-menetelmässä kaikkiaan viisi päävaihetta, joista asiakirja-analyysi on ensimmäinen. Koska tässä tutkielmassa RASKE-menetelmää tarkastellaan erityisesti sen asiakirja-analyysin kautta, arvioidaan RASKE-menetelmää simpukankuorimallin mukaan vain sen asiakirja-analyysin osalta. Tällöin siitä voidaan löytää esimerkiksi seuraavia osia:

Käsitteistö: Esimerkiksi toimija, dokumentti(olio) ja prosessi. Toimija toimii roolissa ja kuvattavan organisaation dokumenttien hallinnan prosessit koostuvat aktiviteeteista. Käsitteistöön kuuluvat myös asiakirja-analyysin aikana tehtävät mallit, joita muun muassa ovat: kontekstikaavio, roolimalli, syöte/tuotosmalli, tilasiirtymäkaavio ja komponenttikaavio.

Notaatio: Kontekstikaavio: suorakaide kuvaa organisaatiota ja ellipsi toiminnon kohdetta. Roolimalli: ympyrä kuvaa roolia ja nuoli roolin kuulumista johonkin rooliluokkaan. Syöte/tuotos-malli: Ellipsi kuvaa aktiviteettia ja katkoviiva dokumenttivuota input/output-mallissa. Tilasiirtymäkaavio: suorakaide kuvaa yhtä tilaa ja nuoli tilan siirtymistä. Komponenttikaavio: suorakaide kuvaa yksittäistä komponenttia ja viiva komponentin suhdetta toiseen komponenttiin.

Prosessit: Tiedonkeruuseen liittyviä prosesseja ovat mm. kohdealueen henkilöiden haastattelut sekä lähdedokumentteihin tutustuminen. Itse menetelmä etenee kuvion 2 osoittamalla tavalla.

Osallistuminen ja roolit: Sovellusalueen ihmisten osallistuminen dokumenttien hallinnan kehittämiseen sekä dokumenttianalyttikoiden ja järjestelmäkehittäjien roolit.

Kehitystavoitteet ja -päätökset: Tehdään muutoksia kohdealueen aktiiviteetteihin, rooleihin ja prosesseihin, jotta dokumenttien hallinta tehostuisi. (tavoite) Rakenteistetaan dokumentit. (päätös)

Oletukset ja arvot: Rakenteistaminen vaatii perusteellista asiakirja-analyysiä. Perusteelliseen asiakirja-analyysiin tarvitaan kohdealueen henkilöitä ja heidän tietämystään.

3.5 Maler & El Andaloussin menetelmän arvioiminen simpukankuorimallin mukaan

RASKE-menetelmään on otettu vaikutteita myös rakenteisten dokumenttien laatimiseen ja dokumenttien hallinnan kehittämiseen käytettävästä Maler & El Andaloussin menetelmästä. Seuraavaksi käydään Maler & El Andaloussin menetelmä lyhyesti läpi. Lisäksi Maler & El Andaloussin menetelmää tarkastellaan simpukankuorimallin avulla.

Maler & El Andaloussin menetelmä on kehitetty DTD-määritysten tuottamisen tueksi. DTD on oleellinen osa SGML- tai XML-dokumenttia. DTD:ssä määritellään dokumentin looginen rakenne, jonka osia kutsutaan elementeiksi. Elementtien lisäksi DTD:ssä voi olla elementtien ominaisuuksia määritteleviä attribuut-

teja tai entiteettejä. Entiteettien avulla voidaan viitata esimerkiksi kuvatiedoston. (Tiitinen, Päivärinta, Salminen & Lyytikäinen 1997)

Maler & El Andaloussin menetelmän tarkoituksena on tukea dokumenttien laatimista rakenteiseen muotoon tarjoamalla vaiheittain etenevän keinon DTD-määrittelyyn. (Maler & El Andaloussi 1996) Seuraavaksi Maler & El Andaloussin menetelmän eteneminen käydään läpi yleisellä tasolla. Teksti pohjautuu Maler & El Andaloussin (1996) teokseen *"Developing SGML DTDs. From text to markup"*.

Maler & El Andaloussin menetelmä jakautuu kolmeen päävaiheeseen, joita ovat

- dokumenttianalyysi
- mallinnusvaihe
- raportin laatiminen.

Menetelmän kahden ensimmäisen päävaiheen aikana käydään läpi kymmenen johdonmukaisesti etenevää alavaihetta (steps). Dokumenttianalyysin aikana näistä alavaiheista suoritetaan kolme ensimmäistä, jotka ovat

1. komponenttiehdokkaitten tunnistaminen
2. komponenttien luokittelu
3. tarpeiden varmistaminen vertailulla.

Ensimmäisessä alavaiheessa dokumenteista lähdetään tunnistamaan mahdollisia komponentteja. Komponentilla tarkoitetaan yksittäistä dokumentin rakennosaa. Esimerkki komponentista voisi olla dokumentin otsikko tai tekstikappale. Alavaiheen tarkoituksena on tunnistaa ja määrittellä kaikki ns. *semanttiset*

komponentit, joilla tarkoitetaan niitä dokumentin rakenneyksiköitä, jotka yksilöivät jotain tiettyä

- sisältöä (content-based components),
- rakennetta (structural components) tai
- esityksellistä muotoilua (presentational components).

Esimerkkinä sisältöä määrittelevästä komponentista voisi olla osoitetiedotkomponentti. Osoitteessahan hyvin usein esiintyy muun muassa kadunnimi, suuntanumero ja kaupunki. Rakennetta määrittelevä komponentti voisi esimerkiksi olla tekstikappale-, lista- tai taulukko-komponentti. Nämähän määrittelevät dokumentin sisällön rakennetta. Komponentit, jotka määrittelevät jotain esityksellistä muotoilua, liittyvät aina dokumentin ulkoasuun. Näitä komponentteja ovat esimerkiksi ne tekstinpätkät, joiden kirjasinlaji poikkeaa muusta dokumentin tekstistä.

Ensimmäisen alavaiheen aikana aletaan komponenteista laatia *komponenttilistaa*, johon kirjataan kunkin komponentin nimi, komponentin tarkempi määrittely, sekä kuvailua komponentista. On kuitenkin tärkeää pyrkiä pitämään komponenttilista yksinkertaisena. Esimerkiksi ylätunniste-komponentti voitaisiin määritellä komponenttilistaan seuraavalla tavalla:

Nimi: Ylätunniste

Määrittely: Dokumentin aloittava osuus, johon on kirjattu dokumentin tietoja.

Kuvailu: Ylätunnisteessa kerrotaan dokumentin nimi, sen laatija, organisaatio yhteystietoineen sekä dokumentin laatimispäivämäärä.

Toisessa alavaiheessa jatketaan komponenttien analysointia jaottelemalla niitä luokkiin ja alaluokkiin. Luokittelutapa on vapaasti valittavissa; luokittelu voi

perustua esimerkiksi komponenttien rakenteiden samankaltaisuuksiin. Esimerkiksi dokumentin otsikot niiden tasoista riippumatta (onko kyse pää- vai väliot-sikoista) voivat muodostaa yhden mahdollisen luokan. On tärkeää, että komponenttien luokittelu tehdään huolellisesti, koska luokittelun avulla voidaan huomata tarpeettomia komponentteja tai saada epäselväksi jääneet komponentit hahmottumaan selkeämmin. Perusteellinen luokittelu voi myös olla apuna myöhemmin tapahtuvaa DTD-määrittelyä varten. Ensimmäisen alavaiheen aikana laaditun komponenttilistan pohjalta laaditaan komponenttikohtaiset komponenttilomakkeet. Lomakkeille lisätään maininta komponentin luokittelusta. Vaihtoehtoisesti komponenttilomakkeet voidaan järjestellä laadittujen luokkien mukaan.

Kolmannen alavaiheen tarkoituksena on verrata komponenttilistaa jo olemassa oleviin, muissa projekteissa laadittuihin samankaltaisiin listoihin, mikäli tällaisia listoja on saatavilla. Se, miksi tällainen vertailu suoritetaan Maler & El Andaloussin menetelmässä vasta kolmantena eikä ensimmäisenä, johtuu siitä, että vertailu valmiisiin komponenttilistoihin saattaisi dokumenttianalyysin alkuvaiheilla vinouttaa näkemystä komponenttien luokittelusta oman työn tapauksessa. Samankaltaisten DTD-tiedostojen vertaileminen voi kuitenkin herättää ajatuksia sellaisista komponenteista, jotka ovat hyvin ilmeisiä myös omassa tapauksessa, mutta joita ei ollut vielä onnistuttu hahmottaa.

Maler & El Andaloussin menetelmän ensimmäisen vaiheen eli dokumenttianalyysin tuloksina ovat komponenttilista, komponenttilomakkeet, komponenttien luokittelu sekä käytettävien termien sanasto.

Toisen päävaiheen, mallinnusvaiheen, aikana suoritetaan loput seitsemän alavaihetta. Nämä vaiheet ovat

4. semanttisten komponenttien valitseminen
5. dokumenttihierarkian muodostaminen
6. tietoyksiköiden määrittely
7. data-tason komponenttien määrittely
8. oksien sitominen
9. yhteyksien luominen
10. suunnittelutyön kokoaminen.

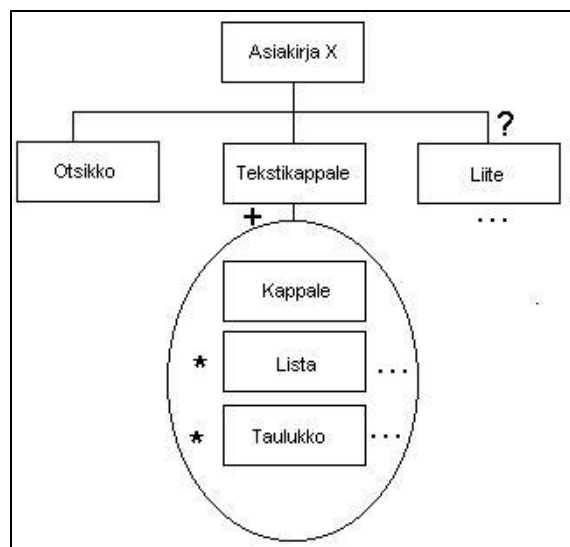
Neljännän alavaiheen aikana valitaan ne komponentit, jotka tullaan säilyttämään työn jatkuessa. Alavaiheen aikana komponenttilomakkeita tutkitaan huolellisesti. Kunkin komponentin kohdalta päätetään joko pitää se analyysin jatkuessa tai hylätä se. Tarkoituksena on, että jokaisen komponentin tarkoitus tulee voida perustella. Alavaiheen aikana olisi myös hyvä dokumentoida kunkin komponentin kohdalta se syy, miksi komponentti säilytettiin tai miksi se hylättiin. Tämä voidaan tehdä lisäämällä komponenttilomakkeelle kohdat "Hyväksytty" ja "Perustelut", joista ensimmäisen kohtaan kirjoitetaan joko "kyllä" tai "ei" ja jälkimmäisen kohtaan kirjataan syy komponentin säilymiselle tai hylkäämiselle. Esimerkiksi väliotsikko-komponentti voidaan merkitä lomakkeelle seuraavasti:

Nimi: Väliotsikko

Hyväksytty: Kyllä

Perustelut: Asiakirja X:ssä tulee olla aina yksi pääotsikko, joka aloittaa dokumentin. Mikäli Asiakirja X:n tekstisisältö on pitkä, tarvitaan väliotsikoita jäsentämään tekstiä loogisiin kokonaisuuksiin.

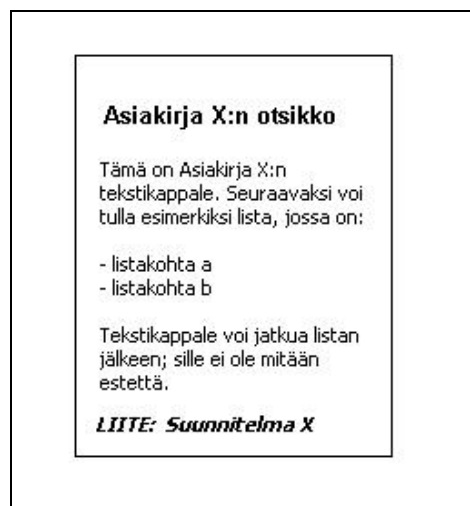
Kun komponenttilista on käyty läpi, siirrytään viidenteen alavaiheeseen, jonka aikana laaditaan listatuista komponenteista yhtenäinen kokonaisuus, joka kuvataan dokumenttihierarkiaan. Ensin komponenttilistasta erotellaan ne komponentit, jotka ovat jollain tapaa yhteydessä dokumentin hierarkiaan eli siihen järjestykseen, jossa komponentit dokumentissa esiintyvät. Siten dokumenttihierarkialla tarkoitetaan komponenttien suhteita toisiinsa. Dokumenttihierarkia kuvataan Elm-mallin notaation mukaan (KUVIO 8), jossa komponentit järjestetään ylhäältä alaspäin puumallin mukaisesti.



KUVIO 8. Dokumenttirakenteen kuvaus Elm-mallissa

Komponentteja merkitään Elm-mallissa suorakaiteilla, jotka nimetään komponenttikohtaisesti. Kuviossa 8 komponentteja ovat muun muassa otsikko, tekstikappale ja liite. Jos komponentit on kuvattu ellipsin sisälle, komponenttien esiintymisjärjestys voi olla mikä tahansa. Esimerkiksi kuviossa 8 Asiakirja X:n tekstikappale rakentuu kappaleesta, listasta ja taulukosta, jotka voivat esiintyä tekstikappaleessa missä järjestyksessä tahansa.

Kaaviossa symbolilla '?' ilmaistaan, että kyseinen komponentti voi puuttua. Symboli '+' kuvaa toistoa siten, että osa esiintyy vähintään kerran. Symboli '*' merkitsee myös toistoa, mutta nyt komponentin ei välttämättä tarvitse esiintyä lainkaan. Suorakaiteen vieressä tai alla olevat kolme pistettä kertovat, että kyseisen komponentin kuvaus jatkuu toisaalla, esimerkiksi erillisessä kaaviossa. Siten esimerkiksi Asiakirja X:n kohdalla liite-komponentti voi puuttua kokonaan. Asiakirja X:ssä ei myöskään tarvitse esiintyä listaa tai taulukkoa, mutta tarvittaessa niitä voi olla useampikin. Sen sijaan Asiakirja X:ssä tekstikappaleen tulee esiintyä ainakin kerran. Koska otsikko-komponentin kohdalla ei ole mitään symbolia, otsikko esiintyy Asiakirja X:ssä kerran ja vain kerran. Alla (KUVIO 9) on yksinkertainen hahmotelma siitä, miltä Asiakirja X voisi näyttää.



KUVIO 9. Asiakirja X.

Siirryttäessä kuudenteen alavaiheeseen, on dokumenttihierarkiat jo laadittu. On kuitenkin mahdollista, että komponenttilomakkeissa on vielä komponentteja, joita ei sisällytetty dokumenttihierarkiaan. Nämä komponentit ovat todennä-

köisesti luonteeltaan sellaisia, jotka jäsentävät dokumentin pääasiallista sisältöä eli liittyvät mahdollisesti dokumentin ulkoasuun. Nämä komponentit on jaettava kahteen eri ryhmään, joita ovat

- tietoyksiköt (information units) eli ns. IU-yksiköt
- data-tason komponentit (data level components).

Maler & El Andaloussin kuudennessa alavaiheessa etsitään juuri edellä mainittuja tietoyksiköitä. Tietoyksiköillä on usein oma "sisäinen rakenteensa" ja ne voivat olla rakenteeltaan monimutkaisia, toisin kuin data-tason komponentit. Kun tietoyksiköitä on löydetty, niiden sisältö ja sisäinen rakenne mallinnetaan esimerkiksi omiin pieniin puumalleihinsa (ks. KUVIO 8, s 37). Kaikki löydettyt tietoyksiköt kirjataan erilliseen tietoyksikkölistaan. Tietoyksiköt tulee myös olla nähtävissä komponenttilomakkeilla niiden komponenttien kohdalla, joihin ne liittyvät.

Seitsemännessä alavaiheessa palataan jälleen jo laadittuun komponenttilistaan. Kun kuudennessa alavaiheessa listalta poistettiin ne komponentit, jotka muodostivat tietoyksiköitä, tässä alavaiheessa listalta poimitaan ne komponentit, jotka muodostavat data-tason komponentteja. Data-tason komponentit kartoitetaan elementeiksi ja attribuuteiksi ja ne kuvataan tarkasti ns. elementtilomakkeisiin. Kahdeksanteen alavaiheeseen siirryttäessä alavaiheiden 4, 6 ja 7 listaukset kootaan viimein yhteen. Näitä listauksiahan olivat komponenttilistat, tietoyksikkö-lomake sekä elementtilomake. Kaikki luodut elementtilomakkeet tulee olla täytettyinä huolellisesti jokaisen elementin kohdalta.

Kun kahdeksan ensimmäistä alavaihetta on suoritettu, ei luotujen listausten ja lomakkeiden ulkopuolelle jääneitä komponentteja yleensä enää ole paljon. Mi-

käli tällaisia ongelmallisia komponentteja kuitenkin esiintyy, ne kootaan omaan listaansa. Siten siirryttäessä yhdeksänteen alavaiheeseen näiden ylijääneiden komponenttien lista on jo yleensä lyhyt. Ne komponentit, jotka listalta vielä löytyvät, ovat usein sellaisia, joiden avulla voidaan yhdistää tietoja toisiinsa. Tällaisia komponentteja kutsutaan linkkikomponenteiksi. Yhdeksännen alavaiheen tarkoituksena on tunnistaa linkkikomponentit sekä kuvata ja määritellä niiden tarkoitus. Tämän alavaiheen aikana dokumentin usein toistuvista sisällöistä voidaan myös määritellä *entiteettejä*, jotka ovat omia tekstistä tai muussa muodossa olevasta datasta rakentuvia kokonaisuuksia. Kun tämä on tehty, siirrytään viimeiseen eli kymmenenteen alavaiheeseen. Tässä alavaiheessa edellisten yhdeksän alavaiheen aikana tehty tuotettu dokumentaatio tarkastetaan ja dokumentaatioon voidaan tarvittaessa tehdä korjauksia.

Kolmannessa Maler & El Andaloussin menetelmän päävaiheessa laaditaan raportti, johon kerätään analyysin ja mallinnuksen aikana tuotettu dokumentaatio ja kootaan niistä looginen kokonaisuus.

Maler & El Andaloussin menetelmää arvioidessa simpukankuorimallia tarkastellen, voidaan menetelmästä löytää seuraavia osia:

Käsitteistö: Dokumentti, komponentti, IU-yksikkö, elementti, attribuutti, datatason elementti.

Notaatio: Dokumenttihierarkiamalli, (Elm-malli): komponentti kuvataan suorakaiteella, komponenttien väliset suhteet viivalla. Jos malli jatkuu esimerkiksi seuraavalla sivulla, jatkuvuutta kuvataan kolmella pisteellä. Hattara kuvaa määrittelemätöntä tekstiä.

Prosessit: Kolmen päävaiheen aikana käydään läpi kymmenen alavaihetta, joiden avulla dokumenteista etsitään, tunnistetaan ja luokitellaan erillisiä komponentteja.

Osallistuminen ja roolit: Kohdealueen ihmiset sekä dokumenttianalyttikot.

Kehitystavoitteet ja -päätökset: (tavoite) Toimiva DTD-tiedosto organisaation tarpeisiin.

Oletukset ja arvot: Rakenteisista dokumenteista on organisaatiolle hyötyä. Analyysivaiheessa kohdealueen ihmiset ja heidän tietämyksensä ovat tärkeitä.

3.6 RASKE-menetelmän ja Maler & El Andaloussin menetelmän vertailua

Sekä RASKE-menetelmä että Maler & El Andaloussin menetelmä sopivat dokumenttien rakenteistamistyön tueksi, sillä molemmat menetelmät sisältävät asiakirja-analyysivaiheen (Maler & El Andaloussin menetelmässä dokumenttianalyysi), jonka aikana pyritään kartoittamaan kohdealueen dokumenttityyppejä. Menetelmien asiakirja-analyysit painottavat kuitenkin eri asioita, jonka vuoksi menetelmistä on löydettävissä eroavaisuuksia.

Siinä, missä Maler & El Andaloussin menetelmä keskittyy dokumentteihin ja niistä löydettäviin komponentteihin, RASKE-menetelmästä löytyy keinot myös dokumenttien ulkopuolelle kohdistuvalle analyysille. RASKE-menetelmässähän keskitytään myös tarkastelemaan organisaatiota, jossa dokumentteja laaditaan. Siksi RASKE-menetelmä ei ole pelkästään dokumenttien rakenteistamiseen tarkoitettu menetelmä, kuten Maler & El Andaloussin menetelmä, vaan erityisesti dokumenttien hallinnan kehittämiseen tarkoitettu mene-

telmä. Tällaisten menetelmien määrä, jotka keskittyvät organisaation dokumenttien hallinnan kehittämiseen, on hyvin pieni. (Päivärinta 2001)

Luvuissa 3.4 ja 3.5 arvioitiin RASKE-menetelmää sekä Maler & El Andaloussin menetelmää simpukankuorimallin mukaan. Kun näitä arvioita verrataan keskenään, löydetään menetelmistä eroavaisuuksia myös menetelmien käsitteistön, notaation, prosessien, sekä kehitystavoitteiden ja -päätöksien suhteen. Kun Maler & El Andaloussin menetelmässä ydinkäsitteistö liittyy dokumentteihin ja dokumentin osiin (komponentit, elementit, attribuutit) niin RASKE-menetelmän käsitteistöön kuuluvat dokumentin lisäksi olennaisesti myös kohdealueen roolit ja prosessit. RASKE-menetelmässä dokumenttia ajatellaan myös dokumenttioliona, jolla on esimerkiksi aina yhdellä hetkellä jokin tietty tila. Maler & El Andaloussin menetelmän tapauksessa dokumentti on ylin käsite, jonka alakäsitteinä komponentit, elementit ja attribuutit ovat.

Menetelmien notaatioita tarkastelemalla huomataan, että Maler & El Andaloussin menetelmästä löytyy vain dokumenttihierarkian kuvaamiseen tarkoitettun Elm-mallin notaatio. Tämä malli notaatioineen löytyy myös RASKE-menetelmästä, mutta sen lisäksi RASKE-menetelmän notaatioon kuuluvat myös muut mallit, joiden notaatio poikkeaa Elm-mallin notaatiosta. Menetelmien prosessit poikkeavat myös toisistaan. RASKE-menetelmässä keskeisiä prosesseja ovat sovellusalueen ihmisten haastattelut sekä tutustuminen lähdedokumentteihin. Lisäksi RASKE-menetelmän asiakirja-analyysi on yksi prosessi, joka etenee omalla tavallaan (ks. KUVIO 2, s. 24). Maler & El Andaloussin menetelmän kohdalla prosesseja ovat menetelmän kolme päävaihetta, joiden aikana suoritetaan järjestyksessä 10 alavaihetta tietyssä järjestyksessä; yhden alavaiheen lo-

puttua alkaa seuraava alavaihe. RASKE-menetelmän asiakirja-analyysi vaiheessa alavaiheita voidaan suorittaa samanaikaisesti.

Menetelmien kehitystavoitteet ja -päätökset poikkeavat myös toisistaan. RASKE-menetelmässä yhtenä keskeisimpänä kehitystavoitteena on tehdä muutoksia kohdealueen aktiviteetteihin, rooleihin ja prosesseihin, jotta dokumenttien hallinta tehostuisi. Maler & El Andaloussin menetelmässä keskeisimpänä kehitystavoitteena on toimivan DTD-tiedoston tai -tiedostojen laatiminen organisaation tarpeisiin. Siten RASKE-menetelmä painottaa dokumenttien hallinnan osuutta, kun taas Maler & El Andaloussin menetelmä ei ota siihen juuri kantaa.

Menetelmien oletukset ja arvot ovat kuitenkin lähellä toisiaan; molemmat menetelmät tunnustavat kohdealueen ihmisten roolin sekä molemmissa menetelmissä oletetaan, että rakenteistamisesta on organisaatiolle hyötyä. Myös menetelmien osallistuminen ja roolit -taso on molemmissa menetelmissä sama; rooleihin kuuluvat sekä dokumenttianalyttikot että sovellusalueen ihmiset, joiden kanssa tehtävä yhteistyö on tärkeää.

Maler & El Andaloussin menetelmän mukainen mallintaminen on luonteeltaan staattista. Menetelmä tarjoaa johdonmukaisesti etenevät vaiheet, joita seuraamalla DTD-määritysten sekä SGML- tai XML-dokumenttien laatiminen ja käyttöönotto on mahdollista. Rakenteisia dokumentteja ja dokumenttien hallintaa kehitettäessä RASKE-menetelmää seuraamalla, mallintaminen on Maler & El Andaloussin menetelmän tapaan staattista muun muassa niiltä osin, joissa RASKE-menetelmä soveltaa Maler & El Andaloussin menetelmää. Erityisesti asiakirja-analyysin aikana suoritettava dokumenttikomponenttien mallintaminen on tällainen vaihe.

RASKE-menetelmän asiakirja-analyysi sisältää myös dynaamista mallintamista, jonka tarkoituksena on kuvata sovellusalueen muuttuvia tapahtumia. Esimerkiksi prosessimallit, kuten syöte/tuotos-malli, ja dokumenttiolion elinkaarta kuvaava tilasiirtymäkaavio, ovat luonteeltaan dynaamista mallintamista. Lisäksi RASKE-menetelmän analyysiprosessi mahdollistaa roolien, dokumenttien ja prosessien samanaikaisen määrittelyn ja mallintamisen (ks. KUVIO 2, s. 24), kun taas Maler & El Andaloussin menetelmässä yksi vaihe suoritetaan loppuun ennen seuraavan alkamista.

4 OPISKELIJAPROJEKTIT

Tässä luvussa kerrotaan kahdesta opiskelijaprojektista, joissa sovellettiin RASKE-menetelmää. Alaluvussa 4.1 esitellään opiskelijaprojektien taustat sekä malli dokumenttien hallinnan osa-alueista. Alaluvut 4.2 ja 4.3 sisältävät kuvaukset Tag2IT-projektin sovellusalueesta sekä projektin etenemisestä. Alaluvuissa 4.4 ja 4.5 kerrotaan vastaavasti XooZoo-projektin sovellusalue ja projektin eteneminen. Opiskelijaprojektien esittely perustuu projekteissa laadittuihin suunnitelmiin sekä raportteihin.

4.1 Taustaa

Projektin johtaminen -kurssi on pakollinen opintojakso lähes jokaiselle perustutkintoaan suorittavalle opiskelijalle Jyväskylän yliopiston Informaatioteknologian tiedekunnassa. Tavallisesti Projektin johtaminen -kurssille osallistuu kolmannen vuoden opiskelijoita, jotka ovat suorittaneet perusopintonsa kokonaisuudessaan sekä aineopintoja jonkin verran. (Vartiainen 2005)

Projektin johtaminen -kurssi on laajuudeltaan 15 opintopistettä (10 opintoviikkoa) ja sen kesto ajallisesti on kuusi kuukautta. Opintojakson aikana aiemmin opiskeltu teoria testataan käytännössä, kun opiskelijat saavat tehtäväkseen toteuttaa asiakkaan tilaaman työn. Lukukaudella 2001 - 2002 kurssin painopiste siirtyi projektityöstä projektin hallintaan. Projekteille tarjotut aiheet ovat vaihdelleet puhtaista koodausprojekteista erilaisiin kartoitusprojekteihin. (Vartiainen 2005)

Syksyllä 2004 Jyväskylän yliopiston informaatioteknologian tiedekunta tarjosi Projektin johtaminen -kurssille projektin aiheita, joista kaksi keskittyi yliopiston dokumenttien hallinnan kehittämiseen. Tiedekunnan toimisto koki tarvetta automatisoida tiedekuntaneuvoston kokousasiakirjoja sekä niiden laatimisprosesseja. Toiseksi tiedekunnan opinto-oppaan laatimisprosessi ja julkaiseminen koettiin monimutkaiseksi ja hankalaksi, joten oppaan laatimista ja koostamista haluttiin automatisoida mahdollisimman pitkälle. Molempien projektiaiheiden tapauksessa XML-kieli oli kiinnitetty projektiaiheen teknologiaksi. (Tietojenkäsittelytieteiden laitos 2004a : Tietojenkäsittelytieteiden laitos 2004b) Seuraavaksi kerrotaan lyhyesti projektiaiheiden taustat. Tarkemmat projektiaiheiden kuvaukset ovat tämän tutkielman liitteinä (Liite 8, Liite 9).

Tiedekunnan toimisto tuottaa vuosittain suuren määrän kokousasiakirjoja, joiden tietosisältö on hyvin toistuvaa. (Tag2IT 2005d) Toistuvaa tietosisältöä, kuten esimerkiksi esittelylistoissa esiintyvää lakitekstiä, kuten hallintojohtosääntöjä, ei nykyisellään pystytä käyttämään hyväksi suoraan eri dokumentteja tuottaessa. (Tietojenkäsittelytieteiden laitos 2004a) Kokousasiakirjojen laatimisprosessit ovat elektronisten asiakirjojen laatimista ja muokkaamista. Käytäntö on kuitenkin koettu ongelmalliseksi; dokumenttien ulkoasut vaihtelevat, kopiointin takia tekstiin jää virheitä sekä web-julkaisu intranetissa on koettu liian hankalaksi. (Tag2IT 2005d)

Tiedekunnan opinto-opas tuotetaan vuosittain ja sen laatimiseen osallistuu useita eri tahoja sekä tiedekunnan sisältä että muista tiedekunnista. Opinto-oppaan laatimisprosessi on kuitenkin koettu ongelmalliseksi useista eri syistä. Ensinäkin tiedekuntien tavat päättää tutkintorakenteista vaihtelevat; informaatioteknologian tiedekunnassa tutkintorakenteista ja -vaatimuksista päätetään

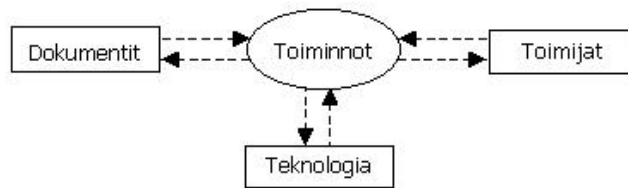
vuosittain, kun taas joissakin muissa tiedekunnissa ne voidaan päättää esimerkiksi kahden vuoden välein. Tämä hankaloittaa systemaattista opinto-oppaan laadintaa. (XooZoo 2005h)

Opinto-oppaan laatimisprosessissa on muitakin haasteita. Tiedekunnan sisällä ei ole yhteisten opintojen vastuuhenkilöä, joka koordinoisi tiedekunnan yleisten opintojen sisältöä ja aikataulua. Suunniteltavaa opetusohjelmaa jouduttiin myös tuomaan tiedekuntaneuvoston käsiteltäväksi useaan eri kertaan. Laadittujen Microsoft Word -asiakirjojen versio- ja muutoshallinta puuttuvat myös kokonaan. Koska yliopistolla ei ole olemassa yhteisiä julkaisumäärittämiä esimerkiksi tyylitiedostolle web-julkaisun kohdalla, oppaan yhtenäisen ulkoasun aikaansaaminen voi olla haastavaa. (XooZoo 2005h)

Molemmat tiedekunnan tarjoamat projekti-aiheet koskivat siis dokumenttien hallintaa ja sen kehittämistä tiedekunnan sisällä. Alaluvussa 4.2 esitellään Tag2IT-projekti, jonka tehtävänä oli kartoittaa tiedekunnan toimiston kokousasiakirjojen laatimista sekä suunnitella asiakirjoille uusi laatimistapa hyödyntäen XML-kieltä. Alaluvussa 4.3 kerrotaan tarkemmin, kuinka Tag2IT-projekti eteni Projektin johtaminen -kurssilla lukukautena 2004 - 2005. Alaluvussa 4.4 esitellään XooZoo-projekti, joka kartoitti tiedekunnan opinto-oppaan laatimisprosessia ja vastaavasti alaluvussa 4.4 kerrotaan, kuinka XooZoo-projekti eteni lukukautena 2004 - 2005. Alalukujen 4.2 ja 4.4 opiskelijaprojektien esittelyt seuraavat dokumenttien hallinnalle laadittua mallia, joka esitellään seuraavaksi.

Dokumenttien hallinnan sovellusalueelta voidaan erotella neljä eri osa-alueita, joita ovat dokumentit, toimijat (actors), toiminnot (activities) ja teknologia (technology) (Salminen 2003). *Toimijat* suorittavat sovellusalueen toimintoja.

Toimijoita ovat siis sovellusalueen ihmiset ja organisaatiot. *Toiminnot* ovat organisaation sisäisiä tai organisaatioiden välisiä prosesseja ja *teknologiaa* käytetään toiminnoissa. Dokumenteista, toimijoista ja teknologiasta välittyy aina informaatiota johonkin toimintoon. Alla olevassa kuviossa informaatiota on kuvattu nuolilla. (RASKE 2005)



KUVIO 10. Dokumenttien hallinnan osa-alueet (Salminen 2003, 917)

Tässä tutkielmassa tapaustutkimuksen analyysiyksikköinä ovat Tag2IT- ja XooZoo-projektit, joiden asiakkaana toimi Jyväskylän yliopiston informaatio-tekniikan tiedekunta (Tag2IT 2004a : XooZoo 2005a). Seuraavaksi esitellään projektien sovellusalueet dokumenttien hallinnan osa-alueiden mukaisesti.

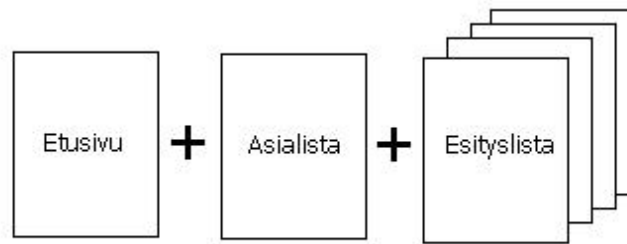
4.2 Tag2IT-projekti

Tässä luvussa esitellään Tag2IT-projektin osa-alueet dokumentteineen, toimijoineen, toimintoineen ja teknologioineen luvussa 4.1 esitetyn dokumenttien hallinnan mallin mukaisesti (ks. KUVIO 10). Esittely perustuu Tag2IT-projektissa laadittuihin suunnitelmiin sekä raportteihin.

Sovellusalueen dokumentit

Tag2IT-projektin sovellusalueena olivat tiedekunnan kokousasiakirjat ja niiden rakenteistaminen (Tag2IT 2005e). Kokousasiakirjoilla tarkoitetaan tiedekunta-

neuvoston jäsenille lähetettävää kokouksen esittelylistaa ja sen pohjalta laadittavaa kokouksen pöytäkirjaa. *Esittelylista* (KUVIO 11) muodostuu etusivusta, joka kattaa kokouskutsun, asialistasta, joka on kokouksessa käsiteltävien asioiden luettelo, sekä yksittäisistä esityslistoista. Kun nämä osat liitetään yhteen, muodostuu valmis esittelylista. Esittelylistan sivumäärä voi siis vaihdella, mutta keskimäärin se kattaa noin 10 - 15 sivua. (Tag2IT 2005d)



KUVIO 11. Esittelylistan rakenne (Tag2IT 2005d, 47)

Sovellusalueen toimijat

Tiedekuntaneuvoston kokouksen esittelylistaan saadaan tietoa sekä tiedekunnan sisältä että ulkopuolelta. Esimerkiksi informaatioteknologian tiedekunnan laitokset toimittavat opiskelijoiden tutkinnonhakulomakkeet tiedekunnan toimistoon. Lisäksi laitokset ilmoittavat toimistolle tulevista väitöstutkimuksista. Tiedekunnan ulkopuolelta tietoa saadaan esimerkiksi virkojen hakijoilta sekä opiskelupaikan hakijoilta. (Tag2IT 2004c).

Esittelylistan koostamisesta ja lähettämisestä tiedekuntaneuvoston jäsenille vastaa tiedekunnan hallintopäällikkö. Hallintopäällikkö sekä valmistelee kokouksessa käsiteltävät asiat että toimii kokouksen esittelijänä. Opintoihin liittyvien asioiden esittelijänä toimii opintoasiainpäällikkö. (Tag2IT 2005d) Esittelylistan hyväksyy tiedekunnan dekaani (Tag2IT, 2004c). Kun tiedekuntaneuvoston ko-

kous on pidetty, tiedekunnan toimistossa laaditaan esittelylistan pohjalta kokouksen pöytäkirja. (Tag2IT 2005d)

Sovellusalueen toiminnot

Esittelylistan laatimisprosessi alkaa kokouksen valmistelulla tiedekunnan toimistossa. Hallintopäällikkö, opintoasiainpäällikkö sekä osastosihteeri kokoavat tietoja ehdotetuista nimityksistä, valmistumisista sekä muista kokoukseen vietävistä asioista. Kukin kokoukseen vietävä asia (esittelylistassa esitys) laaditaan yhteen Microsoft Word -dokumenttiin ja nämä dokumentit tallennetaan verkkoasemalle. Hallintopäällikkö kokoaa erillisistä asiakirjoista yhtenäisen kokonaisuuden, joista muodostuu esittelylista. Dekaanin tarkistaa esittelylistan ja hyväksyy sen sellaisenaan tai muutoksin. Tämän jälkeen esittelylista julkaistaan paperiversiona ja jaetaan tiedekuntaneuvoston jäsenille. Esittelylistan jakelu tapahtuu yliopiston sisäisen postin kautta, mutta myös osastosihteeri voi hoitaa sen. (Tag2IT 2004c)

Kun tiedekuntaneuvoston kokous on ohi, hallintopäällikkö ja opintoasiainpäällikkö muokkaavat esittelylistaa muistiinpanojen pohjalta. Tämän jälkeen erilliset dokumentit taas kootaan, joista muodostuu pöytäkirja. Dekaanin tarkastaa ja hyväksyy pöytäkirjan sellaisenaan tai muutoksin, jonka jälkeen tiedekunnan toimisto julkaisee siitä paperiversioon. Osa pöytäkirjan sisällöstä viedään tiedekunnan www-sivuille. (Tag2IT 2004c)

Sovellusalueen teknologia

Tiedekunnan toimisto laatii kokousasiakirjat Microsoft Word -tekstinkäsittelyohjelmalla. Kun seuraavaa tiedekuntaneuvoston kokousta ale-

taan valmistella, verkkoaseman hakemistorakenteeseen luodaan uusi kansio, jonne uuden esittelylistan osat (ks. KUVIO 11, s 49) talletetaan.

Tag2IT-projektin tavoitteena oli laatia tarvittavat määrittelyt, dokumenttipohjat, ja muunnosohjelmat niin, että esittelylistojen ja pöytäkirjojen laatimisessa siirryttäisiin käyttämään XML-kieleen perustuvia sisältömäärittelyksiä sekä muunnos- ja tyylimäärittelyksiä. Tavoitteena oli suunnitella ja toteuttaa uuden XML-muotoisen tiedon käsittelyyn pohjautuva toimintaprosessi. (Tag2IT 2005e) Tag2IT-projekti päättyi siirtämään tiedekunnan toimiston kokousasiakirjat Microsoft Word -tekstinkäsittelyohjelman asiakirjoista Microsoft InfoPath -lomakkeille. (Tag2IT 2005a)

InfoPath-lomakkeet tallentavat niihin syötetyn tiedon XML-muotoon. XML-kieli tukee monikanavajulkaisua, eli yhdelle tietosisällölle on mahdollista laatia erilaisia ulkoasuja esimerkiksi web- ja paperijulkaisu. Siten Tag2IT-projektissa pyrki mahdollistamaan InfoPath-lomakkeille syötetyn sisällön tuottamisen PDF (Portable Document Format) -dokumenteiksi ja HTML (Hypertext Markup Language) -tiedostoiksi. InfoPath-lomakkeille haluttiin kuitenkin laatia valmis rakennemäärittely, jonka johdosta projekti laati DTD-määrittelyksiä. DTD-määrittelysten laatimisessa käytettiin Altova XmlSpy -ohjelmaa (Tag2IT 2005d). Suunnittelun työvälineenä käytettiin myös Altova Stylevision -ohjelmaa, jonka avulla oli mahdollista luoda tarvittavat tyyli-tiedostot. Muunnosten suunnittelussa pyrittiin löytämään sellainen ratkaisu, jonka avulla muunnos tapahtuisi mahdollisimman vaivattomasti sekä HTML- että PDF-muotoon. (Tag2IT 2005a)

Tag2IT-projekti sai tiedekunnalta käyttöönsä työasemia, joiden käyttöjärjestelmänä oli Microsoft Windows XP. Jokaiselle koneelle on asennettu Microsoft

Office 2003 paketti. Käytössä oli myös CorelDraw piirto-ohjelma, Altova XmlSpy XML -editori ja MS Project -sovellus. (Tag2IT 2004a)

4.3 Tag2IT-projektin eteneminen

Tag2IT-projektin projektiryhmään kuului viisi tietojärjestelmätieteiden opiskelijaa. Projektin asiakkaana toimi Jyväskylän yliopiston informaatioteknologian tiedekunnan toimisto. (Tag2IT 2005e)

Tag2IT-projekti jaettiin viiteen vaiheeseen, joita olivat

1. käynnistysvaihe
2. kartoitusvaihe
3. toteutusvaihe
4. testausvaihe ja
5. päätösvaihe (Tag2IT 2004a).

Projektin käynnistysvaiheessa suunniteltiin ja puitteet projektin aloittamiselle. Kartoitusvaiheen aikana projektiryhmän tehtävänä oli määrittellä ja suunnitella projektin toteutusvaiheessa tehtävän työn perusta. (Tag2IT 2004a) Kartoitusvaiheessa projektiryhmä sovelsi RASKE-menetelmää sovellusalueen kartoittamisessa. Ryhmä jakoi RASKE-menetelmän asiakirja-analyysin aikana laadittavat mallit projektiryhmän kesken siten, että osa ryhmästä mallinsi rooleja, osa laati prosessikuvauksia ja osa kartoitti dokumenttikomponentteja. (Tag2IT 2004b) Kartoitusvaiheen aikana tuotettiin raportti asiakasorganisaation sekä kokousasiakirjojen laatimisprosessin nykytilasta RASKE-menetelmää mukailleen. Raporttiin lisättiin myös kuvausta tavoitetilasta. (Tag2IT 2004a)

Toteutusvaiheessa projektiryhmä valitsi loppukäyttäjän sovellukseksi Microsoft InfoPath-ohjelman. Ohjelma valittiin, koska se oli jo valmiiksi asennettuina tiedekunnan toimiston työasemille. Lisäksi valintaan vaikutti ohjelman samankaltaisuus Microsoft Word -tekstinkäsittelyohjelman kanssa, jolla tiedekuntaneuvoston kokousasiakirjat nykyisessä prosessissa laaditaan. Koska molemmat ohjelmat ovat osa Microsoft Office -tuoteperhettä, on niissä havaittavissa myös samankaltaisuuksia, voivat helpottaa asiakkaan siirtymistä vanhasta kokousasiakirjojen laatimistavasta uuteen. Erityisesti InfoPath-ohjelman valintaan vaikutti se, että laadittuihin lomakkeisiin syötetty informaatio tallentuu XML-kielen muotoon, joka mahdollistaa XML-tiedon muuntamisen sekä HTML- että PDF-muotoon. (Tag2IT 2005a)

Projektin toteutusvaiheen aikana suoritettu suunnittelu jaettiin viiteen eri osaan, joita olivat

- yleisrakenteen suunnittelu
- DTD-määrittelyjen suunnittelu
- käyttöliittymäsuunnittelu
- monikanavajulkaisun suunnittelu ja
- muunnosten suunnittelu (Tag2IT 2005a).

Yleisrakenteen suunnittelu toteutettiin sekä projektiryhmän sisäisesti että asiakkaan ja teknisen asiantuntijan kanssa pidetyissä katselmointitilaisuuksissa (Tag2IT 2005a). Katselmoinnit edelsivät virallisia johtoryhmän kokouksia, mutta ne olivat luonteeltaan epäformaalimpia. Siinä missä johtoryhmän kokoukset keskittyivät projekteissa laadittujen suunnitelmien ja raporttien hyväksymiseen, katselmoinnit järjestettiin itse tuotoksen takia. Katselmointien aikana käydyissä keskusteluissa projektiryhmän, asiakkaan ja teknisen asiantuntijan kesken poh-

dittiin tehtävän toteuttamista. Lisäksi katselmointien aikana projektiryhmä kertoi asiakkaalle tekemästään työstä sekä esitteli laadittuja InfoPath-lomakkeita. Katselmointien rooli itse työn etenemisessä ja kehittämisessä oli projektille merkittävä. (Tag2IT 2005g : Tag2IT 2005h)

Skeemasuunnittelu toteutettiin osittain yhdessä teknisen asiantuntijan kanssa. Varsinainen suunnittelu oli DTD-määrytyksien laatimista. Laaditut DTD:t konvertoitiin jälkepäin XML-skeemoiksi. Työvälineenä tässä käytettiin Altova XmlSpy-ohjelmaa. Skeemasuunnittelussa täytyi erityisesti huomioida XML-skeemojen yhteensopivuus InfoPath-lomakkeiden kanssa. Tämä asetti haasteita skeemasuunnitteluprosessille, sillä InfoPath-ohjelman tarjoamat kentät ovat aina elementtikohtaisia; kunkin elementin sisältö voidaan syöttää sen ilmentämään lomakekenttään. Kentät, kuten tekstilaatikat, pudotusvalikot tai toistuvat taulukot olivat käytettävissä sen mukaan, kuinka yksittäisen elementti on määritelty DTD:ssä. Mikäli elementti saa esiintyä vain kerran, InfoPath-ohjelma tarjoaa sille erilaisia kenttävaihtoehtoja kuin esimerkiksi sellaiselle elementille, joka saa toistua tai joka voi puuttua. (Tag2IT 2005a)

Testausvaiheessa projektiryhmä testasi toteutusvaiheessa tehdyn työn toimivuutta. Vaihe jakaantui testaus- ja jatkokehitysalavaiheisiin. Projektin päätös-vaihe käsitti loppuraportin laadinnan. (Tag2IT 2004a)

Tag2IT-projektin tulos muodostuu DTD- ja skeematiedostoista, InfoPath-ohjelmalla tehdyistä lomakkeista ja Stylevision-ohjelmalla tehdyistä muunnos- ja tyyli-tiedostoista. Toteutetun kokonaisuuden toimintaperiaate on seuraava: esittelylistan (tai pöytäkirjan) osat täytetään InfoPath-lomakkeella. Tiedostot tallentuvat XML-muotoon, jonka jälkeen ne kootaan yhtenäiseksi esittelylistaksi

tai pöytäkirjaksi XSLT (The Extensible Stylesheet Language Transformations)-kielellä koodattua muunnosta hyödyntäen. Esittelylista tai pöytäkirja käännetään muunnos- ja tyylitiedostojen avulla HTML-tiedostoksi (ja jatkossa PDF-tiedostoksi). (Tag2IT 2005e)

Toteutusvaiheessa laaditut kuvaukset ja määrittelyt olivat:

- DTD-määrittelyt ja niistä muodostetut XML-skeemat kokousasiakirjojen osille. Näitä laadittiin 16 kappaletta.
- InfoPath-lomakkeet esittelylistan/pöytäkirjan eri osille, joita laadittiin 14 kappaletta.
- Esitetyt lomakkeet XML-tiedostoina kustakin InfoPath-lomakkeesta.
- HTML-muunnostiedostot eri lomakkeille. (Tag2IT 2005d)

DTD:t määriteltiin siten, että ne kattoivat sekä esittelylista- että pöytäkirjaversi-
on. (Tag2IT 2005d)

4.4 XooZoo-projekti

Seuraavissa alaluvuissa esitellään XooZoo-projektin osa-alueet dokumentteineen, toimijoineen, toimintoineen ja teknologioineen dokumenttien hallinnan mallia mukaillen (ks. KUVIO 10, s. 48). Esittely perustuu XooZoo-projektissa laadittuihin suunnitelmiin sekä raportteihin.

Sovellusalueen dokumentit

XooZoo-projektin tarkoituksena oli kehittää informaatioteknologian tiedekunnan opinto-opasta ja sen vuosittaista laatimisprosessia. (XooZoo 2005a) Tiedekunnan opinto-opas sisältää tietoa muun muassa tiedekunnasta ja tiedekunnan

laitoksista, tutkintorakenteista ja opinnoista kurssikuvauksineen. Painetun opinto-oppaan laajuus on noin 150 sivua. (Jyväskylän yliopisto 2004)

Opinto-oppaaseen saadaan sisältöä useista eri lähteistä. Kurssikuvaukset kopioidaan Korppi-opintotietojärjestelmästä, esittelytekstejä tiedekunnasta, sekä tietoa opinnoista ja opintorakenteesta kahdelta tiedekunnan laitokselta ja laitoksilla toimivilta useilta henkilöiltä. (Tietojenkäsittelytieteiden laitos 2004b)

Sovellusalueen toimijat

Opinto-oppaan laatimiseen osallistuu informaatioteknologian tiedekunnassa lukuisia eri tahoja, kuten amanuenssit, laitoksen opetushenkilöstö, osastosihteerit sekä tekniset käsittelijät. Keskeisessä osassa on opintoasiainpäällikkö, joka koostaa opinto-oppaan. Ennen kuin tiedekuntaneuvosto on hyväksynyt oppaaseen tulevan sisällön, sisältöä valmistellaan hyväksyttäväksi muun muassa amanuenssien, opetushenkilöstön ja opintoasiainpäällikön voimin. Opinto-oppaan www-julkaisun valmistelee tekninen käsittelijä. Opinto-oppaan painojulkaisusta vastaavat opintoasiainpäällikkö sekä tiedekunnan tiedottaja. Opinto-oppaaseen saadaan sisältöä myös tiedekunnan ulkopuolelta, kuten opiskelijapalveluista ja muista tiedekunnista. (XooZoo 2005h)

Sovellusalueen toiminnot

Opinto-oppaan laatimisprosessi alkaa laitosneuvoston tutkintorakenteen ja vaatimusesitysten hyväksymisellä. Kun laitosten opetusohjelmat ovat valmiita, opintoasiainpäällikkö valmistelee niiden pohjalta tutkintorakennevaatimukset. Opintoasiainpäällikön laatimat esitykset hyväksytään tiedekuntaneuvostossa, jonka jälkeen opintoasiainpäällikkö valmistelee opinto-oppaan yleisen sisällön, jonka tiedekuntaneuvoston tulee hyväksyä. Kun opetusohjelma on hyväksytty

sekä opintoasiainpäällikkö että opiskelijapalvelut valmistelevat oppaasta julkaisun, josta tekninen käsittelijä valmistelee www-julkaisun ja opintoasiainpäällikkö yhdessä tiedottajan kanssa toimittaa opinto-oppaan painettavaksi. Kun opinto-opas on saatu painettua sekä opintoasiainpäällikkö että osastosihteeri hoitavat oppaan jakelun. (XooZoo 2005h)

Sovellusalueen teknologia

Informaatioteknologian tiedekunnan opinto-opas laadittiin Microsoft Word -tekstinkäsittelyohjelmaa käyttäen. Koska oppaan sisällöntuottajia on useita ja heidän laatimansa tekstit poikkesivat tyyleiltään toisistaan, oli yhtenäisen ulkoasun laatiminen opinto-oppaalle työlästä. Lisäksi opinto-oppaaseen päätyvät kurssikuvaukset kopioitiin Korppi-opintotietojärjestelmästä Microsoft Word -tekstinkäsittelyohjelmaan ja siksi kurssikuvauksia jouduttiin muokkaamaan ennen kuin ne olivat yhtenäisiä muun oppaan sisällön kanssa. Opinto-oppaan laatimisprosessia haluttiin siten keventää ja yhtenä ratkaisuna ehdotettiin oppaan siirtämistä XML-muotoon. (Tietojenkäsittelytieteiden laitos 2004b) Kun XooZoo-projekti lähti toteuttamaan opinto-oppaan siirtämistä XML-muotoon, yhdeksi kehitystyökaluksi valittiin Altova XmlSpy -ohjelma, jonka avulla esimerkiksi DTD-määritysten laatiminen onnistuu. (XooZoo 2004e).

Yliopisto tarjosi XooZoo-projektille neljä työasemaa sekä kannettavan tietokoneen perusohjelmistoineen. XML-dokumenttien ja DTD-määritysten tekemiseen ryhmä sai käyttöön edellä mainitun Altova XmlSpy-ohjelman. Lisäksi projektilla oli käytössä Near&Far-ohjelmisto DTD-määritysten laatimiselle. XooZoo-projektin tehtävänä oli myös kartoittaa opinto-oppaan laatimisprosessia. Prosessikuvausten mallintamiseen ryhmällä oli käytössään Microsoft Visio -ohjelma. (XooZoo 2005g)

Projektin tavoitteena oli tukea opinto-oppaan web-julkaisua paperijulkaisun lisäksi. Myös oppaan saattaminen PDF-muotoon oli yksi ensisijaisista tavoitteista. Projektin aikana ei päätetty tiettyä sovellusta, jolla esimerkiksi PDF-julkaisun laatiminen suoritettaisiin. Sovellukselle asetettiin vain vaatimuksia; sovelluksen haluttiin olevan hinnaltaan edullinen ja asiakkaalle (opintoasiainpäällikölle) helppokäyttöinen. (XooZoo 2005k)

4.5 XooZoo-projektin eteneminen

XooZoo-projektiin kuului kuusi tietojärjestelmätieteiden opiskelijaa (XooZoo 2005g). XooZoo-projektin tehtävänä oli kartoittaa opinto-oppaan nykyinen laatimisprosessi yleisellä tasolla. Lisäksi eri käyttäjäryhmien mielipiteet oppaan sisällöstä ja käytöstä selvitettiin, mutta projektin aikana ei tuotettu valmiita XML-oppaita eri käyttäjäryhmiä varten. (XooZoo 2005a)

XooZoo-projektin vaiheet olivat seuraavat:

1. käynnistysvaihe
2. opinto-oppaan laatimisprosessin kartoitus
3. opinto-oppaan loogisen rakenteen kartoitus
4. opinto-oppaan sisällön ja käytön kartoitus
5. opinto-oppaan rakenteen määrittäminen ja mallidokumenttien laatiminen XML-kielillä
6. päättämisen vaihe. (XooZoo 2005a)

Käynnistysvaiheessa laadittiin projektisuunnitelma ja siihen sisältyvät liitedokumentit (XooZoo 2004a). Käynnistysvaiheen jälkeen XooZoo-projektissa alkoi

opinto-oppaan laatimisprosessin kartoitusvaihe. Yleisesti vaihe sisälsi opinto-oppaan tuottamista koskevaa kartoitusta. (XooZoo 2005a) Laatimisprosessin kartoitusvaiheen tarkoituksena oli kartoittaa yliopiston opinto-oppaan laatimisprosessia ja sen ongelmakohtia, sekä tehdä ehdotus laatimisprosessista tulevaisuutta varten. Laatimisprosessin kartoitusvaiheen työmenetelminä olivat prosessiin osallistuvien henkilöiden haastattelut. XooZoo-projektin työn etenemistä tukivat myös työpajat, joissa opinto-opasta kartoitettiin projektiryhmän, teknisen asiantuntijan sekä asiakkaan kesken (XooZoo 2004e). Vaiheen keskeisiä tavoitteita olivat

- Tunnistaa opinto-oppaan tuottamiseen osallistuvat henkilöt ja näiden roolit.
- Tutustua prosessien mallinnusmenetelmiin.
- Haastatella opinto-opasta tuottavia henkilöitä ongelmakohtien havaitsemiseksi.
- Analysoida haastattelut, tuottaa niistä raportti sekä tehdä parannusehdotuksia laatimisprosessiin. (XooZoo 2005b, 4-5)

Opinto-oppaan loogisen rakenteen kartoitusvaiheen tarkoituksena oli suunnitella informaatioteknologian tiedekunnan opinto-oppaan looginen rakenne ja määritellä sen sisältökomponentit myöhempää oppaan rakenteistamista varten. Tavoitteena oli tehdä yhteinen raportti opinto-oppaan kartoitus-vaiheen kanssa. (XooZoo 2005c) Kuten Tag2IT-projektissa, myös XooZoo-projektiryhmä jakoi RASKE-menetelmän mallien laatimista projektiryhmän jäsenten kesken. Osa projektiryhmästä laati roolitaulukkoa, osa mallinsi kohdealueen prosesseja ja osa suoritti dokumenttien mallintamisen komponenttitaulukkoon ja komponenttikaavioihin. (XooZoo 2004b : XooZoo 2004c : XooZoo 2004d)

Kokonaisuudessaan loogisen rakenteen kartoittaminen oli XooZoo-projektin kannalta erittäin merkittävä vaihe. Vaiheen aikana projektiryhmä havahtui tarkastelemaan työtapojaan; ryhmän sisällä huomattiin tehdyn päällekkäistä työtä toisen vaiheen kanssa. Vaiheen aikana laaditut tuotokset eivät vastanneet alkuperäisiä vaatimuksia, joten projektiryhmän oli tehtävä uusia päätöksiä ja kehitettävä työskentelytapojaan. Nämä vastoinkäymiset johtuivat projektin toimeksiannon vaativuudesta ja ryhmän kokemattomuudesta dokumenttien rakenteistamisessa. (XooZoo 2005c)

Opinto-oppaan sisällön ja käytön kartoitusvaiheessa kartoitettiin tiedekunnan opinto-oppaan sisältöä (XooZoo 2005d). Kartoituksen apuna käytettiin Jyväskylän yliopiston Korppi-opintotietojärjestelmää, jonka kautta suoritettiin kysely (XooZoo 2005g). Kyselyn kohderyhmänä käytettiin vuoden 2004 aikana aloittaneita opiskelijoita sekä vuosien 2000–2003 aikana aloittaneita opiskelijoita. Kyselyä testattiin omassa projektiryhmässä, jonka jälkeen valmis kysely laitettiin asiakkaan ja ryhmän ohjaajan testattavaksi.

Vaiheen tuloksena koostettiin tulosraportti eri käyttäjäryhmien tarpeista niin sisällön kuin julkaisukanavankin suhteen. Vaiheen tavoitteena oli selvittää opinto-oppaan sisällön tarkoituksenmukaisuutta, sekä tutkia, millaista tietoa mahdollisesti kaivataan lisää, ja mikä nykyisessä oppaassa on hyvää tai tarpeetonta. Lisäksi tarkoituksena oli selvittää, mitä opinto-oppaan sisältämiä tietoja tarvitaan painettuna oppaana, vai riittääkö ainoaksi julkaisukanavaksi tiedekunnan ja laitosten verkkosivut. (XooZoo 2005d) Vaiheen aika tuotettiin opinto-oppaan sisällön ja käytön kartoitus –raportti.

Opinto-oppaan rakenteen määrittäminen ja mallidokumenttien laatiminen XML-kielellä - vaiheen tarkoituksena oli luoda skeemamäärittelyt opinto-oppaan sisällöstä ja testata samaan aikaan skeemoja opinto-oppaan nykyisen sisällön avulla. (XooZoo 2005e) Vaiheen tuloksiksi saatiin DTD-määrittelyt opinto-oppaan sisällöstä. Vaiheen tuloksia olivat myös DTD-määrittelyjen avulla tehdyt mallidokumentit. (XooZoo 2005e)

Päätämisvaiheessa tarkoituksena oli kartoittaa niitä toimenpiteitä, joita tätä projektia hyödyntävien tahojen on kannattavaa toteuttaa. Jatkokehityksessä myös pohdittiin millaisia menetelmiä tai neuvoja projektin pohjalta voidaan antaa jatkokehityksestä vastaaville tahoille. (XooZoo 2005f)

Jo alkuvaiheessa XooZoo-projektin aiheen rajaaminen sekä projektin kokonaisuuden hahmottaminen oli vaikeaa, koska toimeksianto koettiin haastavaksi ja laajaksi. Opinto-oppaan rakenteen osalta projekti sai määritettyä opinto-oppaalle DTD-määrittelyt ja tuotettua määrittelyä vastaavat opinto-oppaan mallidokumentit. (XooZoo 2005g) DTD-määrittelyä XooZoo-projekti laati seuraaville kokonaisuuksille:

- opinto-oppaan julkaisutiedot
- opinto-oppaan yksittäisten sisältöosien käsittelyyn liittyvien tiedoille oma DTD-määrittely, meta.dtd
- opinto-oppaan tekstisisällöille luku.dtd
- Korppi-järjestelmästä saataville opintojaksokuvauksille opintojaksot.dtd
- tiedekunnan tutkintojen merkkaukselle tutkinot.dtd sekä
- laitosten, henkilökunnan yms. yhteystiedoille yhteystiedot.dtd. (XooZoo 2005i)

XML-mallidokumentteja XooZoo-projekti laati lähinnä laatimiensa DTD-määrittysten testaamiseksi. Yhteensä XML-mallidokumentteja toteutettiin 15 erilaista. (XooZoo 2005i)

5 RASKE-MENETELMÄN SOVELTAMINEN OPISKELIJAPROJEKTEISSA

Tässä luvussa käsitellään tutkielman empiirisen osuuden ensimmäinen vaihe, jossa vertaillaan projektien asiakirja-analyysia ja sen tuloksia. Lähteinä ovat opiskelijaprojektien keskeisimmät raportit, joista RASKE-menetelmän asiakirja-analyysiä soveltavat mallit on löydettävissä. Nämä raportit ovat Tag2IT-projektin Kartoitustyön raportti (Tag2IT 2004c) sekä XooZoo-projektin Opintooppaan laatimisprosessin kuvaus & sisältö komponentit -raportti (XooZoo 2005h). Projekteihin osallistuneita opiskelijoita myös haastateltiin asiakirja-analyysia ja erityisesti analyysin aikana laadittavia malleja koskien. Haastatteluiden tuloksia arvioidaan luvussa 6. Alaluvussa 5.1 vertaillaan projektien laatimia malleja toisiinsa ja alaluvussa 5.2 malleja verrataan RASKE-menetelmän notaatioon. Alaluvussa 5.3 kootaan malleista yhteenveto.

5.1 Projekteissa laadittujen mallien vertailua

Sekä Tag2IT- että XooZoo-projekti laativat RASKE-menetelmän mukaisia malleja kartoittaessaan sovellusalueen rooleja, prosesseja ja dokumentteja. Projektit kokosivat asiakirja-analyysin tulokset raportteihinsa. (Tag2IT 2004c : XooZoo 2005h) Seuraavaksi projekteja verrataan suhteessa laadittuihin malleihin ja niiden sisältöön.

Tag2IT-projektissa sovellusalueen kartoitusvaiheen tuloksina olivat seuraavat mallit:

- kontekstikaavio sekä esittelylistan että pöytäkirjan laatimiselle

- roolitaulukko
- syöte/tuotos-malli sekä esittelylistan että pöytäkirjan laatimiselle
- tilasiirtymäkaavio sekä esittelylistan että pöytäkirjan laatimiselle
- komponenttitaulukko sekä esittelylistalle että pöytäkirjalle
- komponenttikaaviot sekä esittelylistalle että pöytäkirjalle. (Tag2IT 2004c : Tag2IT 2005d)

XooZoo-projekti laati RASKE-menetelmän asiakirja-analyysia soveltaessaan seuraavat mallit:

- kontekstikaavio opinto-oppaan laatimiselle
- roolitaulukko
- syöte/tuotos-malli opinto-oppaan laatimisesta sekä tiedekunnan että tietojenkäsittelytieteiden laitoksen näkökulmasta
- tilasiirtymäkaavio opinto-oppaan laatimisesta sekä tiedekunnan että tietojenkäsittelytieteiden laitoksen näkökulmasta
- komponenttitaulukko
- komponenttikaaviot. (XooZoo 2005h)

Vertaamalla Tag2IT- ja XooZoo-projektien laatimia malleja (TAULUKKO 1, s. 65) voidaan huomata, kuinka projektiaiheiden ja siten projektien sovellusalueiden laajuus erosi projektien välillä. Tag2IT-projektin tapauksessa sovellusalue oli suhteellisen pieni; projekti keskittyi kuvaamaan tiedekunnan toimiston tuottamia esittelylistoja ja pöytäkirjoja. XooZoo-projektin tehtävänä oli mallintaa tiedekunnan opinto-opas, johon liittyi luonnollisesti enemmän kohdealueen toimijoita ja toimintoja.

TAULUKKO 1. Opiskelijaprojekteissa laadittuja RASKE-malleja

Malli	Tag2IT	XooZoo	Huom.
Kontekstikaavio	2 kpl	1 kpl	
Roolitaulukko	1 kpl	1 kpl	
Syöte/tuotos-malli	2 kpl	2 kpl	Toinen XooZoo-projektin malleista kuvasi tulevaisuudessa tapahtuvaa prosessia.
Tilasiirtymäkaavio	2 kpl	2 kpl	
Komponentti- taulukko	2 kpl	1 kpl	Tag2IT laati taulukot erikseen esittelylistalle ja pöytäkirjalle, jotka oli eritelty vielä pienempiin osiin komponenttien mukaan.
Komponentti- kaaviot	22 kpl	11 kpl	XooZoo mallinsi opinto-oppaan oleellisimmat komponentit raporttiinsa. Tag2IT kuvasi esittelylistan (11 kaaviota) ja pöytäkirjan (11 kaaviota) rakenteen kokonaan.

Kontekstikaavio

Molemmissa projekteissa toimintaympäristön kuvaus mallinnettiin kontekstikaavioon. Tag2IT-projekti laati kaksi erillistä kontekstikaaviota; yhden esittelylistan (Liite 2) ja yhden pöytäkirjan laatimiselle. XooZoo-projekti kuvasi toimintaympäristönsä yhteen kontekstikaavioon (Liite 3), jonka mallinnuksen kohteena oleva toiminto oli opinto-oppaan laatiminen. Tag2IT-projekti mallinsi erikseen tiedekunnan laitokset ja niiden sisältämät keskeiset roolit, kuten esimerkiksi laitosneuvostot ja laitosten johtajat. XooZoo-projekti mallinsi tiedekunnan sisältämät roolit ilman laitoskohtaista jaottelua.

Molempien projektien kontekstikaavioissa oli kohdealueella mallinnettuina tietolähteinä myös tiedekunnan ulkopuolisia tahoja. Nämä tahot mallinnettiin kaavioon nimeämällä tietolähdettä otsikolla ”Muut”. Tag2IT-projektin sovel-
lusalueella tiedekunnan ulkopuolisia toimijoita olivat opiskelupaikan tai virka-

paikan hakijat ja XooZoo-projektissa tiedekunnan ulkopuolelta tuleva tieto saatiin opiskelijapalveluilta, yliopiston hallinnolta tai muilta tiedekunnilta.

Roolitaulukko

Roolikuvaukset laadittiin molemmissa projekteissa roolitaulukkoihin. Molempien projektien tapauksessa roolit jaoteltiin RASKE-menetelmän mukaisesti laatijoihin ja käyttäjiin.

Tag2IT-projekti jaotteli laatijat viiteen eri ryhmään, joita olivat tuottaja, kokoaja, yhdistäjä, tarkastaja ja julkaisija. Julkaisija jaettiin vielä kahteen alarooliin, joita ovat tulostaja ja jakelija. Hallintopäällikkö toimi jokaisessa edellä mainitussa laatijan roolissa. Hallintopäällikkö muun muassa tuottaa suuren osan esittelylistan esityksistä, tarkastaa esittelylistan sekä vastaa sen yhdistämisestä. Opintoasiainpäällikkö toimi tuottajan, kokoajan ja julkaisijan rooleissa. Opintoasiainpäällikkö tuottaa ne esitykset, jotka kuuluvat opintoasioihin. Näitä ovat muun muassa väitöstutkimuksen arvostelu ja opiskelijavalintaperusteiden esitleminen. Osastosihteeri toimi tuottajan, yhdistäjän ja julkaisijan rooleissa. Osastosihteeri esimerkiksi kirjoittaa esittelylistaan valmistuneiden opiskelijoiden tiedot sekä osallistuu esittelylistan jakeluun. Muita laatijoita olivat dekaani, joka tarkastaa esittelylistan ja hyväksyy sen, yliopiston sisäinen posti, joka osallistuu esittelylistan jakeluun, sekä Copy Shop, joka tulostaa esittelylistan ja pöytäkirjan paperiversion.

XooZoo-projektin roolitaulukossa laatijoita oli yhdeksän erilaista, joita olivat tuottaja, valmistelija, yhdistäjä, koostaja, asiatarastaja, hyväksyjä, tekninen käsittelijä, jakelija ja koordinoija. Tuottajarooli oli XooZoo-projektin roolitaulukossa laajin; opinto-oppaan tuottamiseen osallistuu 22 eri tuottajaa yliopiston

kielikeskuksesta aina laitosten amanuensseihin. Opintoasiainpäällikkö toimi neljässä eri laatijaroolissa. Opintoasiainpäällikkö toimi opinto-oppaan valmistelijana, yhdistäjänä, koostajana ja jakelijana. Opinto-oppaan koostajan roolissa toimii pelkästään opintoasiainpäällikkö. Jakelijaroolissa toimi opintoasiainpäällikön lisäksi osastosihteeri.

Roolitaulukoissa mallinnettiin laatijoiden lisäksi myös käyttäjät. Käyttäjiä Tag2IT-projekti mallinsi kuusi erilaista; kolme ulkoista ja kolme sisäistä käyttäjää. Ulkoisia käyttäjiä olivat web-sivujen vierailija, nimityksen selvittäjä ja muu kokouksesta kiinnostunut henkilö tai taho. Sisäisiä käyttäjiä olivat tiedekunta-neuvoston jäsenet, joille esittelylista toimitetaan, toimiston työntekijät sekä muut sisäiset käyttäjät.

Myös XooZoo-projekti jaotteli käyttäjät ulkoisiin ja sisäisiin käyttäjiin. Yhteensä roolitaulukossa oli käyttäjiä 11 kappaletta. Näistä kaksi oli ulkoisia käyttäjiä ja yhdeksän sisäisiä käyttäjiä. Ulkoiset opinto-oppaan käyttäjät olivat www-sivuilta löytyvän opinto-oppaan selaajat sekä muut käyttäjät. Sisäisiä käyttäjiä olivat muun muassa opiskelijat, jotka on jaoteltu viiteen eri ryhmään pääaineopiskelijoista jatko-opiskelijoihin. Opiskelijoiden lisäksi sisäisiä käyttäjiä olivat opetushenkilöstö, hallintohenkilöstö, opinto-ohjaaja sekä muiden tiedekuntien ja yliopiston yksiköiden henkilöstö.

Syöte/tuotos-malli

Molemmissa projekteissa kohdealueen prosesseja kuvattiin syöte/tuotomalleissa. Tag2IT-projekti mallinsi erikseen esittelylistan (Liite 4) ja pöytäkirjan laatimisprosessit. XooZoo-projektissa laadittiin myös kaksi syöte/tuotos-mallia, joissa kuvattiin koko opinto-oppaan laatimisprosessia. Toisessa mallissa laati-

misprosessi kuvattiin tiedekunnan näkökulmasta (Liite 5) ja toisessa tietojenkäsittelytieteiden laitoksen näkökulmasta.

Tag2IT-projektin kohdalla sekä esittelylistalle että pöytäkirjalle laaditussa syöte/tuotos-malleissa oli neljä aktiviteettia. Esittelylistan laatimisessa (Liite 4) aktiviteetteja olivat valmistelu, kokoaminen, tarkastus ja julkaisu. Valmisteluaktiviteetin syötteitä olivat esittelylistaan tulevat esitykset, kuten työsuunnitelmat ja oikaisupyynnöt. Valmistelun tuotoksena on hallintopäällikön ja opintoasiainpäällikön laatimat esittelylistaan tulevat esitykset erillisinä asiakirjoina. Valmistelun jälkeen suoritetaan kokoaminen, jonka tuotoksena saadaan erillisistä asiakirjoista koottu esittelylista. Kokoamisen jälkeen esittelylistan tarkastaa dekaani. Tarkastusaktiviteetin tuotoksena on siten tarkastettu esittelylista. Viimeinen aktiviteetti on julkaisu, jonka tuotoksena on esittelylistan paperiversio.

Pöytäkirjan laatimiseen kuuluvat aktiviteetit olivat tuottaminen, kokoaminen, tarkastus, web-version sekä paperiversion julkaisu. Pöytäkirjan laatiminen alkaa tuottaminen-aktiviteetista, jonka syöteenä on esittelylista muistiinpanoineen. Tuottamisen tuotoksena saadaan muokatut asialistat. Kun pöytäkirjan asialistojen tuottaminen on valmis, listat kootaan. Kokoaminen-aktiviteetin tuotoksena on valmis pöytäkirja. Pöytäkirjan tarkistaa dekaani, jonka jälkeen pöytäkirja voidaan julkaista sekä paperiversiona että www-ympäristössä.

XooZoo-projektin syöte/tuotos-mallissa (Liite 5) opinto-oppaan laatimisen aktiviteetteja oli 12 kappaletta. Opinto-oppaan laatiminen alkaa tutkintorakennemuutoksen hyväksymisellä. Tämän aktiviteetin tuotoksena ovat laitoskohtaiset tutkintorakenne ja -vaatimusesitykset. Näiden pohjalta opintoasiainpäällikkö

valmistelee molempien laitosten tutkintorakennevaatimukset, jonka tuotoksena saadaan esitykset tutkintorakenteista tiedekuntaneuvostoon. Kolmannessa aktiviteetissa tiedekuntaneuvosto päättää tutkintorakennevaatimuksista. Tämän aktiviteetin jälkeen XooZoo-projektin syöte/tuotos-malliin on laadittu aktiviteetti, joka ei ole mukana opinto-oppaan nykyisessä laatimisprojektissa. Tässä ylimääräisessä aktiviteetissa, jota on mallissa kuvattu värillisellä ellipsillä, erillinen opasryhmä koordinoi opintoja ja opetusohjelmia sekä laatisi aikataulun opinto-oppaan laatimiselle. Aktiviteetin tuotoksena olisi ohjeistus opinto-oppaan laatimiselle tutkintorakenteen pohjalta. Tämän jälkeen voitaisiin suorittaa kaksi eri aktiviteettia; laitosten amanuenssit valmistelisivat laitosten osuudet opinto-oppaan sisällöstä saamaan aikaan, kun opintoasiainpäällikkö valmistelee opinto-oppaan yleistä sisältöä. Aktiviteettien tuotoksina olisivat ehdotukset oppaan sisällöstä.

Kun ehdotukset opinto-oppaan sisällöstä ovat valmiita, opintoasiainpäällikkö valmistelee niiden pohjalta esityksen tiedekuntaneuvoston hyväksyttäväksi. Kun tiedekuntaneuvosto on hyväksynyt opinto-oppaan sisällön, on tuotoksena hyväksytty opinto-opas mahdollisine muutoksineen. Kun opas on hyväksytty, valmistellaan oppaan julkaisua, jonka jälkeen tekninen käsittelijä valmistelee www-julkaisun ja opintoasiainpäällikkö tiedottajan kanssa painojulkaisun. Viimeisenä aktiviteettina syöte/tuotos-mallissa oli oppaan jakelu, jonka tuotoksena opas on jaettu sitä käyttäville tahoille.

XooZoo-projekti kuvasi syöte/tuotos-mallissaan siis sellaisia aktiviteetteja, jotka eivät ole mukana opinto-oppaan nykyisessä laatimisprosessissa. Tällaisia toimintoja oli mallissa kaksi kappaletta; neljäs ja viiden aktiviteetti. Neljännessä aktiviteetissa aktiviteetin suorittajana toimisi opasryhmä, jonka tarkoituksena

olisi tutkintorakenteen pojalta tuottaa ohjeistus oppaan laatimiselle. Viidennessä aktiviteetissa laitosten amanuenssit valmistelisivat laitosten omat osuuden opinto-oppaan sisällöstä. Näitä ehdotuksia tulevaisuuden laatimisprosessin aktiviteeteiksi kuvaavia ellipsejä on merkitty värillisellä ellipsillä valkoisen sijaan.

XooZoo-projektin toinen syöte/tuotos-malli kuvasi edellä mainitun viidennen aktiviteetin etenemistä, eli sitä, kuinka laitosten amanuenssit laatisivat opinto-oppaan sisältöä. Tässä mallissa aktiviteetteja oli kahdeksan kappaletta. Ensin laitoksen amanuenssi valmistelee oppaan sisältöä ja laatii ehdotuksen laitoksen neuvostoon. Laitos neuvosto hyväksyy laitoksen oman osuuden oppaan sisällöstä, jonka jälkeen amanuenssi yhdessä laitoksen opetushenkilöstön kanssa valmistelee laitospöytäkirjat kurssikuvaukset ja aikataulut. Tämän kolmannen aktiviteetin tuotoksena olisi tuloste kurssitiedoista. XooZoo-projekti mallinsi tämän kolmannen aktiviteetin aikana suoritettavat viisi aktiviteettia samaan syöte/tuotos-malliin. Kaikki nämä aktiviteetit liittyvät kurssitietojen muokkaamiseen laitoksen sisällä.

Tilasiirtymäkaavio

Projektit mallinsivat kartoitustyössään myös dokumentin elinkaarta kuvaavia tilasiirtymäkaavioita. Tag2IT-projekti päätyi laatimaan tilasiirtymäkaavioitakin kaksi kappaletta; yhden esittelylistalle (Liite 6) ja yhden pöytäkirjalle. Tiloja molemmissa tilasiirtymäkaaviossa oli viisi kappaletta. Ensimmäisessä kaaviossa kuvataan opinto-oppaan laatimisprosessia (Liite 7) eli koko oppaan elinkaarta. Toisessa tilasiirtymäkaaviossa kuvattiin oppaaseen tulevien tekstien tilaa. Molemmissa kaavioissa tiloja on kahdeksan.

XooZoo-projekti oli myös lisännyt tilasiirtymäkaavioiden yhteyteen tavoiteaikataulut, joiden aikana yksittäinen opinto-oppaan laatimiseen tai oppaaseen päätyvän tekstin tila tapahtuu. Opinto-oppaan laatimisprosessia kuvaavaan tilasiirtymäkaavioon oli kuukausien lisäksi lisätty ylöspäin suuntautuvia nuolia; nämä nuolet mallintavat mahdollisesti tapahtuvia muutoksia tai hylättyjä tutkintorakenteita, joiden takia joudutaan palamaan edelliseen tilaan.

Dokumenttimallit

Dokumenttikomponentit mallinnettiin molemmissa projekteissa sekä komponenttitaulukoihin että komponenttikaavioihin. Tag2IT-projekti laati komponenttitaulukot esittelylistalle ja pöytäkirjalle erikseen, kun taas XooZoo-projektilla komponenttitaulukoita on yksi ja se on laadittu koko opinto-oppaalle.

Tag2IT-projektin komponenttitaulukot olivat XooZoo-projektin vastaavaan taulukkoon verrattuna huomattavasti yksinkertaisempia; kun Tag2IT-projektin laatimat komponenttitaulukot sisältävät vain kaksi saraketta, joista ensimmäisessä on dokumenttikomponentin nimi ja jälkimmäinen on varattu komponentin sisältämän tiedon kommentoimiselle (TAULUKKO 2, s. 73), sisälsi XooZoo-projektin komponenttitaulukko jopa kuusi sarakkeita (TAULUKKO 3, s. 73). Tag2IT-projektin komponenttitaulukko oli myös jaoteltu pienempiin osiin. Mikäli komponentin nimen jälkeen oli kirjoitettu kolme pistettä, kyseisen komponentin kuvaus jatkui toisessa taulukossa.

XooZoo-projektin komponenttitaulukossa komponentit oli jaoteltu ensimmäisen, toisen ja kolmannen tason komponentteihin. Näiden lisäksi taulukkoon oli lisätty sarake, joka kuvaa komponenttien sijaintia nykyisessä opinto-oppaassa.

Tämän jälkeen oli vielä sarake komponenttien kuvailuun, ja viimeiseksi Laati-
misapu-sarake, joka on tarkoitettu vaikeammin hahmotettavissa oleville kom-
ponenteille.

TAULUKKO 2. Osa Tag2IT-projektin komponenttitaulukkoa esittelylistalle.
(Tag2IT 2004 c, 14)

1. Esittelylista	Huomioitavaa
1.1 Kutsu...	Sisältää lakitekstiä.
1.2 Asialista...	Asialista luettelee kokouksessa käsitel- tävät esitykset. Asialistan pituus on yksi sivu.
1.3 Esityslistat...	Esityslistat ovat esityksiä, joita käsitel- lään kokouksessa. Sisältää lakitekstiä.

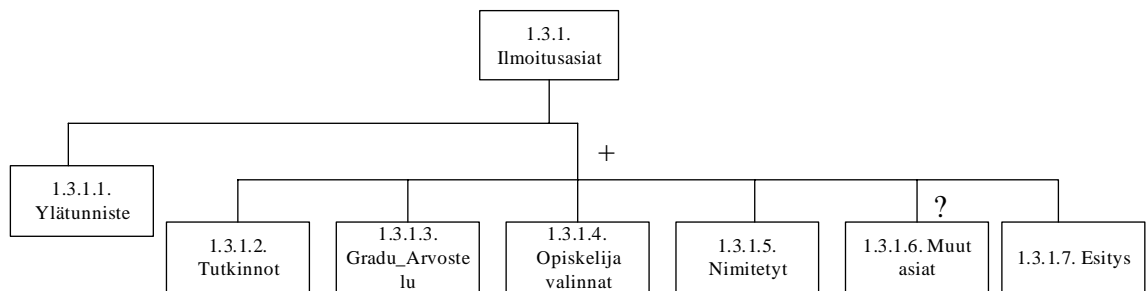
TAULUKKO 3. Osa XooZoo-projektin komponenttitaulukkoa opinto-oppaalle.
(XooZoo 2005d, 26)

1. Tason komponentti	2. Tason komponentti	3. Tason komponentti	Sisällön sijainti nykyisessä oppaassa	Kuvaus	Laatimisapu
1. Opas			"Koko nykyinen opas"	Koko asia- kirjan juu- ri.	
2. Kansi			Oppaan ulko- ja sisäkansi		Tekstiä. Saat- taa tulla myös kuva.
3. Meta					Oppaan meta- tiedoille muo- dostettava oma skeema.

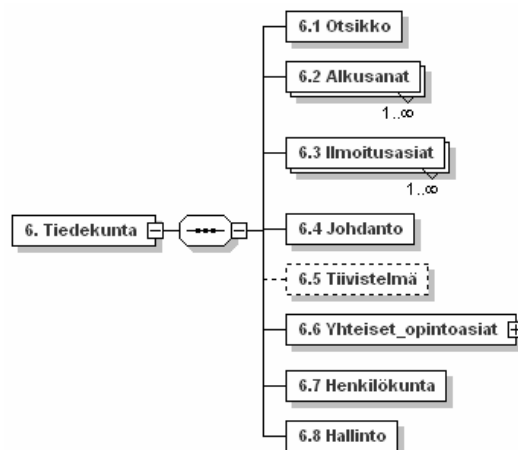
Toisin kuin Tag2IT-projektin tapauksessa, XooZoo-projektin komponenttitau-
lukossa sama komponentti saattoi toistua useammalla rivillä. Esimerkiksi tau-
lukossa esiintyvä 1. tason komponentti "tiedekunta" toistuu taulukon 51 rivillä.

Toisin sanoen tiedekunta-komponentti rakentuu 51 komponentista, joita olivat muun muassa komponentit "otsikko", "tiivistelmä", "johdanto" ja "yhteiset opintoasiat". Nämä 2. asteen komponentit taas rakentuvat joukosta 3. asteen komponentteja.

Se, kuinka komponenttikaavioita sovellettiin, eroaa myös projekteittain. Tag2IT-projekti sovelsi Elm-mallin notaatiota (KUVIO 10) mahdollisimman tarkasti. Sen sijaan XooZoo-projekti hyödynsi Altova XmlSpy -ohjelmalla tuotettavia kaavioita (KUVIO 11).



KUVIO 12. Tag2IT-projektin laatima komponenttikaavio (Tag2IT 2005d, 24)



KUVIO 13. XooZoo-projektin laatima komponenttikaavio (XooZoo 2005h, 47)

Muut kaaviot mallinnettiin projekteissa samoilla työvälineillä; projektien käytävissä oli Microsoft Visio ja Microsoft Word -ohjelmat, joilla asiakirja-analyysin mallit raportteihin laadittiin.

Tag2IT-projekti laati kartoitustyön aikana laaditut mallit raporttiin, josta laadittiin kahdeksan eri versiota ennen dokumentin lopullista hyväksymistä (Tag2IT 2004c). XooZoo-projektissa malleja laadittiin Opinto-oppaan laatimisprosessin kuvaus & sisältökomponentit -raporttiin, josta laadittiin jopa 16 versiota ennen raportin lopullista hyväksymistä (XooZoo 2005h). Jo tästä voidaan huomata, että XooZoo-projektin kartoitustyö RASKE-menetelmän asiakirja-analyysia soveltaen oli työläämpää kuin Tag2IT-projektissa.

5.2 Projektien mallit verrattuna RASKE-menetelmään

Verrattaessa Tag2IT- ja XooZoo-projektien tuottamia asiakirja-analyysin malleja RASKE-menetelmän notaation mukaisiin malleihin, huomataan malleissa selkeitä eroja. Tässä aluvuossa verrataan projektien laatimia malleja seuraavassa järjestyksessä: kontekstikaavio, syöte/tuotos-malli, tilasiirtymäkaavio ja komponenttikaavio. Roolitaulukkoa ja komponenttitaulukoita ei oteta vertailuun mukaan.

Tag2IT-projektin kontekstikaaviota (Liite 2) tarkasteltaessa huomataan, että malli eroaa RASKE-menetelmän notaatiosta. Esimerkiksi sovellusalueelle (ellipsi) osoittavat nuolet on Tag2IT-projektin malleissa kuvattu kiintein nuolin, kun taas RASKE-menetelmässä on nuolissa käytetty katkoviivaa (ks. KUVIO 3, s. 26). Myös XooZoo-projektin kontekstikaavio (Liite3) poikkeaa RASKE-menetelmän notaatiosta. Myöskään tässä kontekstikaaviossa nuolet eivät ole

katkoviivalla merkittyjä. Lisäksi yhdestä kohdealueen toimijasta lähtee useita nuolia, eli yhdellä toimijalla on useita rooleja kohdealueen toiminnon (opinto-oppaan laatiminen) suhteen.

Myös projektien laatimat syöte/tuotos-mallit poikkesivat RASKE-menetelmän notaatiosta. Tag2IT-projektin syöte/tuotos-mallissa (Liite 4) on käytetty kiinteitä nuolia mallintamaan aktiviteettiin (ellipsi) tulevaa syötettä tai siitä lähtevää tuotosta katkoviivojen sijaan, kun taas RASKE-menetelmän notaation mukaan syöte/tuotos-mallissa aktiviteettiin liittyvät syötteet ja tuotokset kuvataan katkoviivoin.

XooZoo-projektin syöte/tuotos-mallissa (Liite5) aktiviteetteihin liittyvät syötteet ja tuotokset on kuvattu RASKE-menetelmän notaation mukaisesti. Syöte/tuotos-malli poikkesi RASKE-menetelmän notaatiosta siten, että mallin aktiviteetteihin oli lisätty numeroita, mikäli aktiviteetti oli nykyisessä prosessissa ongelmallinen. Lisäksi malliin lisättiin aktiviteetteja, jotka eivät ole mukana nykyisessä opinto-oppaan laatimisprosessissa. Näitä aktiviteetteja kuvaavat ellipsit oli erotettu nykyisen prosessin aktiviteeteista värillä.

Molemmat projektit laativat myös tilasiirtymäkaavioita, joissa mallinnettiin dokumenttiolion elinkaarta. Verrattaessa Tag2IT-projektin tilasiirtymäkaaviota (Liite 6) RASKE-menetelmän notaatioon, huomataan, että malli on lähes RASKE-menetelmän notaation mukainen. Kaavion tilat on numeroitu, nimetty ja esitetty suorakaitein. Ainoastaan tilasta toiseen siirtyvää tapahtumaa ei ole selitetty, ts. siirtymää kuvaavan nuolen oheen ei ole selitetty sitä tapahtumaa, joka siirtää dokumenttiolion tilan toiseksi.

XooZoo-projektin tilasiirtymäkaavio (Liite 7) poikkeaa myös RASKE-menetelmän notaatiosta. Projektin laatimassa tilasiirtymäkaaviossa on yhden opinto-oppaan tilaa merkitty värillä. Värillä on tilasiirtymäkaaviossa samanlainen merkitys, kuin projektin laatimassa syöte/tuotos-mallissakin; kyseessä olevaa tilaa ei ole ollut opinto-oppaan elinkaareissa mallin laatimishetkellä mukana vaan se on ehdotus tulevaisuutta ajatellen. Sen sijaan toisin kuin Tag2IT-projekti, XooZoo-projekti on kirjoittanut kaavioonsa siirtymistä kuvaavien nuolien yhteyteen ne tapahtumat, jotka ovat ehtoina opinto-oppaan tilan siirtymisessä toiseksi.

XooZoo-projektin laatimiin tilasiirtymäkaavioihin oli mallinnettu tilojen yhteyteen ehtoja. Kaavion mukaan tilasta voidaan palata edelliseen, mikäli jokin ehto täyttyy. Esimerkkinä ehdosta on ”hylätty tutkintorakenne”, jonka jälkeen kaavioon mallinnetusta tilasta palataan sitä edeltäneeseen tilaan. Ehtojen ja tilasta takaisinpäin siirtyminen on mahdollista myös RASKE-menetelmän notaatiossa, tosin merkintätapa edelliseen tilaan palaamisesta on hieman erilainen. XooZoo-projektin laatimassa tilasiirtymäkaaviossa palaaminen on kuvattu suorakaiteiden oikealta puolelta kulkevalla nuolilla, kun RASKE-menetelmässä palaava nuoli on merkitty samoin kuin tilaa siirtävä nuoli; myös tilasta palaava nuoli on mallinnettu tiloja kuvaavien suorakaiteiden väliin.

Tag2IT-projekti mallinsi komponenttikaaviot Elm-mallin notaatiota noudattamalla, joka on myös RASKE-menetelmän mukainen tapa mallintaa dokumenttien osien suhdetta toisiinsa. XooZoo-projekti päätyi hyödyntämään Altova XmlSpy-ohjelman tuottamia kaavioita. Näin ollen XooZoo-projektin komponenttikaaviot poikkesivat RASKE-menetelmän notaatiosta eniten.

5.3 Yhteenvetoa malleista

Molemmat opiskelijaprojektit sovelsivat kartoitustyössään seuraavia asiakirja-analyysin aikana tuotettavia malleja:

- kontekstikaavio
- roolitaulukko
- syöte/tuotos-malli
- tilasiirtymäkaavio
- komponenttitaulukko ja
- komponenttikaaviot (Tag2IT 2004c : XooZoo 2005h).

Kumpikaan projekteista ei päätynyt käyttämään esimerkiksi roolimallia tai D-R-kaaviota, jotka löytyvät RASKE-menetelmästä. Projektit eivät myöskään tyytynyt pelkään syöte- tai tuotos-malliin, vaan molemmissa projekteissa mallit oli yhdistetty syöte/tuotos-malliksi.

Projekti aiheiden laajuudet on nähtävissä RASKE-menetelmällä tuotetuista malleista. Erityisesti erot korostuivat syöte/tuotos-mallissa, tilasiirtymäkaaviossa sekä komponenttitaulukossa, jotka olivat XooZoo-projektilla huomattavasti Tag2IT-projektin malleja monimutkaisempia. Näitä malleja vertaamalla huomataan, että opinto-oppaan laatiminen on tiedekuntaneuvoston kokousta varten laadittavaa esittelylistaa huomattavasti monimutkaisempi prosessi. Myös opinto-oppaan linkaari on esittelylistan vastaavaa monimutkaisempi; tämä on nähtävissä vertailemalla projektien laatimia tilasiirtymäkaavioita. Lisäksi XooZoo-projektin komponenttitaulukkokin oli Tag2IT-projektin vastaavia huomattavasti pidempi sekä yksityiskohtaisempi. Projekti aiheiden laajuuksien erot on myös

nähtävissä roolitaulukkoista sekä kontekstikaavioista, joihin on kuvattu projektien sovellusalueella toimivia rooleja.

Vaikka RASKE-menetelmä oli projekteissa käytössä, niin kummankaan projektin mallit eivät täsmällisesti noudattaneet RASKE-menetelmän notaatiota vaikka selkeästikin mallit on tunnistettavissa RASKE-menetelmään kuuluviksi (ks. Liitteet 1-7). Esimerkiksi nuolien käyttäminen malleissa poikkesi RASKE-menetelmän notaatiosta. Lisäksi malleissa käytettiin värejä; Tag2IT-projekti erotteli väreillä pöytäkirjaan kuuluvia komponentteja ja XooZoo-projekti merkitsi prosessimalleihin väreillä aktiviteetteja, jotka eivät olleet mukana nykyisessä opinto-oppaan laatimisprosessissa. Tämä värien hyödyntäminen poikkesi eniten RASKE-menetelmän notaatiosta. XooZoo-projekti käytti myös prosessimallissa numerointia, mikäli jokin prosessin aktiviteetti oli laatimisprosessissa ongelmallinen.

Projekteissa päädyttiin muokkaamaan malleja vastaamaan oman kartoitustyön tarpeita. Tag2IT-projekti sovelsi erityisesti Elm-mallin mukaisia komponenttikaavioita siten, että värillä merkityt komponentit olivat osa tiedekunnan toimiston laatimaa pöytäkirjaa. Lisäksi sen komponentin nimi oli lihavoitu, jonka sisältö muuttui jollain tapaa siirryttäessä esittelylistasta pöytäkirjaan. XooZoo-projekti sovelsi eniten syöte/tuotos-malleja sekä tilasiirtymäkaavioita. Tag2IT-projektin tapaan väriä käytettiin erottelemaan sellaisia aktiviteetteja tai tiloja, jotka eivät ole vielä nykyisessä opinto-oppaan laatimisprosessissa mukana. Lisäksi tilasiirtymäkaavion yhteyteen oli lisätty opinto-oppaan laatimisprosessin tavoiteaikataulu.

Molemmat opiskelijaprojektit käyttivät Microsoft Visio -ohjelmaa mallintaessaan RASKE-menetelmän mukaisia malleja. Erityisesti Tag2IT-projektin laatimissa malleissa tämän työkalun vaikutus on nähtävissä; mallit on laadittu ohjelman tarjoamalla valmiilla tyyleillä, kuten nuolilla, jotka poikkeavat RASKE-menetelmän notaatiosta huomattavasti. Roolitaulukot ja komponenttitaulukot määriteltiin molemmissa projekteissa Microsoft Word -ohjelmalla. Lisäksi XooZoo-projekti hyödynsi Altova XmlSpy -ohjelmaa. Kaikki nämä ohjelmat olivat valmiiksi asennettuina projektihuoneiden työasemilla, tosin Altova XmlSpy -ohjelma oli asennettuna vain osalla työasemia.

Kun projektien samoja malleja ja niiden notaatioita verrataan keskenään, on niistä havaittavissa paljon yhteneväisyyksiä. Lähinnä toisiaan ovat projektien laatimat roolitaulukot; Tag2IT-projekti vain lisäsi laatijoiden ja käyttäjien lisäksi yhden ylimääräisen roolin, "Muu informaatiolähde". Muuten roolitaulukot vastasivat toisiaan lähes täysin. Eniten toisistaan poikkeavat komponenttikaaviot; kun Tag2IT-projekti käytti komponenttikaavioissaan Elm-mallin notaatiota, XooZoo-projekti puolestaan käytti apunaan Altova XmlSpy-ohjelmaa ja ohjelman laatimia kaavioita.

Projekteissa luotujen mallien pohjalta on selvästi erotettavissa projektiaiheiden laajuudet ja niiden eroavaisuudet. Poikkeuksetta XooZoo-projektin tuottamat mallit olivat pituudeltaan Tag2IT-projektin vastaavia pidempiä. Esimerkiksi XooZoo-projektin laatima roolitaulukko sisältää huomattavasti enemmän rooleja kuin Tag2IT-projektin vastaava. Myös syöte/tuotos-mallit on XooZoo-projektilla huomattavasti pidemmät. Suurin eroavaisuus lienee kuitenkin projektien komponenttitaulukoilla sekä taulukoiden sisällön että niiden laajuuden suhteen.

6 RASKE-MENETELMÄN KÄYTTÖ JA OMAKSUMINEN OPISKELIJAPROJEKTEISSA

Haastattelu on yksi tiedonhankinnan perusmuoto (Hirsijärvi & Hurme 2000). Tässä luvussa arvioidaan RASKE-menetelmän omaksumista opiskelijoiden keskuudessa Tag2IT- ja XooZoo-projekteihin kuuluneille opiskelijoille suoritet-
tujen haastattelujen pohjalta.

Haastatteluiden tarkoituksena oli kartoittaa opiskelijoiden kokemuksia RASKE-menetelmän soveltamisesta projekteissaan. Kysymysten painopiste oli erityises-
ti RASKE-menetelmän asiakirja-analyysin aikana laadittavien mallien käytöstä ja omaksumisesta oman projektin kohdalla. Haastattelut olivat yksilöhaastatte-
luita, koska tarkoituksena ei ollut selvittää yleistä projektikohtaista kantaa RASKE-menetelmään, vaan tavoitteena oli kartoittaa yksittäisen opiskelijan nä-
kemyksiä ja kokemuksia RASKE-menetelmän mallien soveltamisesta.

Haastattelut olivat puolistrukturoituja haastatteluja, jotka sopivat hyvin kvalita-
tiiviseen tutkimukseen. Puolistrukturoiduissa haastatteluissa kysymysten muo-
to on kaikille haastateltaville sama, mutta kysymysten järjestys voi vaihdella
haastattelukohtaisesti (Hirsijärvi & Hurme 2000). Haastatteluiden tuloksina oli
kuvauksia RASKE-menetelmän soveltamisesta uudellisissa ympäristöissä se-
kä kuvausta siitä, kuinka RASKE-menetelmän soveltaminen opiskelijaprojek-
teissa onnistui opiskelijoiden näkökulmasta. Koska kyseessä on tapaustutki-
mus, mitään yleistyksiä RASKE-menetelmän soveltamisesta ei voida tehdä.

Haastattelut suoritettiin 15.3 – 13.4.2005 välisenä aikana ja haastatteluihin osal-
listui neljä Tag2IT-projektin jäsentä sekä kolme XooZoo-projektin jäsentä. Haas-

tattelupyynnöt esitettiin Tag2IT-projektiryhmään kuuluville opiskelijoille ryhmäpalaverissa, jolloin opiskelijat suostuivat haastateltaviksi (Tag2IT 2005f). XooZoo-projektiryhmälle haastattelupyynnöt lähetettiin sähköpostin kautta ja ne opiskelijat, jotka kokivat olleensa eniten mukana RASKE-menetelmän käyttämisessä, ilmoittautuivat osallistumisesta haastatteluihin (XooZoo 2005j). Haastattelukysymykset keskittyvät pitkälti RASKE-menetelmän asiakirja-analyysin aikana laadittaviin malleihin. Haastattelukysymykset ovat tämän tutkielman liitteenä (Liite 1).

Seuraavaksi käydään läpi haastatteluista saatuja tuloksia. Alaluvussa 6.1 kerrotaan kuinka opiskelijaprojektit päätyivät valitsemaan RASKE-menetelmän työnsä tueksi ja kuinka menetelmään tutustuttiin. Alaluvussa 6.2 esitellään haastatteluihin osallistuneiden opiskelijoiden taustoja sekä heidän mielipiteistään RASKE-menetelmän omaksumisesta yleisellä tasolla. Tämän jälkeen alaluvussa 6.3 kerrotaan mitä opiskelijat ajattelivat laatimistaan RASKE-menetelmän malleista ja viimeiseksi puretaan opiskelijoiden mielipiteet RASKE-menetelmän käytöstä projektityöskentelyssä.

6.1 RASKE-menetelmän valinta ja menetelmään tutustuminen

RASKE-menetelmän valintaan Tag2IT- ja XooZoo-projekteissa vaikutti osaksi se, että jo ennen Projektin johtaminen -kurssin alkamista näiden projektien aiheisiin oli kiinnitetty XML-kieli yhdeksi teknologiaksi. Lisäksi projektiaiheiden kuvauksissa mahdolliseksi menetelmäksi oli ehdotettu RASKE-menetelmää Maler & El Andaloussin menetelmän lisäksi (ks. Liite 8, Liite 9). (Tietojenkäsittelytieteiden laitos 2004a : Tietojenkäsittelytieteiden laitos 2004b)

Tag2IT-projektille RASKE-menetelmän valinta oli jo melko selvää projektin alusta alkaen. RASKE-menetelmän lopulliseen valintaan vaikutti se, että projektiryhmän tekninen asiantuntia tarjosi menetelmästä koulutusta. (Tag2IT 2004e) Lisäksi projektiryhmässä oli kaksi Digitaalisen median suuntautumisvaihtoehdon opiskelijaa, jotka olivat kuulleet RASKE-menetelmästä suuntautumisvaihtoehdon kurssien kautta ennen projektityön alkua. Projektiryhmä harkitsi myös vaihtoehtoisia menetelmiä, mutta päätyi yksimielisesti RASKE-menetelmään tiedekunnan tarjoaman koulutustilaisuuden jälkeen. Koulutuksen järjesti Digitaalisen median lehtori, joka toimi myös projektiryhmän teknisenä asiantuntijana. (Tag2IT 2004b) Koulutustilaisuuden aikana RASKE-menetelmän asiakirja-analyysi vaihe käytiin läpi.

Myös XooZoo-projekti valitsi laatimisprosessin kuvaukseen käytettäväksi menetelmäksi RASKE-menetelmän ryhmän teknisen asiantuntijan ehdotuksen pohjalta. (XooZoo 2005b) Ryhmä tutustui RASKE-menetelmästä saatuun kirjalliseen materiaaliin ennen oman kartoitustyön aloittamista (XooZoo 2004b). Projektiryhmä kävi myös koulutuksessa RASKE-menetelmää koskien (XooZoo 2005b). Koulutustilaisuus oli yhteinen Tag2IT-projektin kanssa (Tag2IT 2004b). Myös XooZoo-projektiryhmässä oli digitaalisen median suuntautumisvaihtoehdon opiskelijoita, jotka olivat kuulleet menetelmästä ennen projektityön alkamista.

Koulutustilaisuuden lisäksi molempien projektien opiskelijoilla oli mahdollisuus tutkia mallidokumentteja aikaisemmista projekteista, joissa oli käytetty RASKE-menetelmää. Myös RASKE-menetelmästä julkaistuja tieteellisiä artikkeleita oli saatavilla, joissa oli nähtävissä RASKE-menetelmän notaation mukaisia malleja. Myös järjestetyssä koulutustilaisuudessa opiskelijoille jaettiin yhteen-

vetomainen ohjeistus RASKE-menetelmän asiakirja-analyysin vaiheista ja vaiheisiin kuuluvista malleista.

6.2 Haastateltavat ja RASKE-menetelmän omaksuminen

Haastatteluihin osallistuneista opiskelijoista suuri osa oli opintojensa loppupuolella. Viisi seitsemästä kertoi, ettei heillä ollut aikaisempaa kokemusta tietojärjestelmätieteiden projektityöskentelystä. Kartoitettaessa haastateltavien tietämystä menetelmistä ja malleista, jokainen haastatteluun osallistunut opiskelija kertoi UML-mallinnuskielen olevan heille tuttu. Sen sijaan Maler & El Andalousin menetelmään oli tutustunut vain kolme opiskelijaa; kaksi XooZoo- ja yksi Tag2IT-projektista. OOA-menetelmä oli tuttu opintojen yhteydestä vain yhdelle opiskelijalle, kun taas ICN-mallit eivät olleet tuttuja yhdellekään vastanneista.

Haastatteluihin vastanneista kukaan ei ollut käyttänyt RASKE-menetelmää ennen projektia. Sen sijaan RASKE-menetelmästä oli kuullut luentojen kautta neljä opiskelijaa, kaksi XooZoo- ja kaksi Tag2IT-projektista. Kysyttäessä RASKE-menetelmän omaksumisesta, vastanneiden enemmistö oli sitä mieltä, ettei menetelmän oppiminen ollut vaikeaa. Yhdelle vastanneista oppiminen oli hieman hankalaa ja yksi kertoi menetelmän oppimisen vaatineen ponnisteluita.

Kysyttäessä mielipidettä siitä, voiko RASKE-menetelmän omaksua vain lukeamalla menetelmästä kirjoitettuja julkaisuja, mielipiteet jakautuivat lähes tasan; neljä vastasi myöntävästi ja kolme kieltävästi. Alla olevaan taulukkoon (TAULUKKO 4, s. 85) on koottu yhteen havaintoja RASKE-menetelmän omaksumisesta opiskelijoittain. Taulukon ensimmäisessä sarakkeessa on lueteltu

haastatteluihin osallistuneet opiskelijat. Kirjain T viittaa opiskelijaan Tag2IT-projektiryhmässä ja kirjain X opiskelijaan XooZoo-projektiryhmässä.

TAULUKKO 4. RASKE-menetelmän omaksuminen opiskelijoittain

Opiskelija	Tutut menetelmät	Kuullut/lukenut RASKE-menetelmästä aikaisemmin	RASKE-menetelmän oppiminen oli vaikeaa	Menetelmän voi oppia lukemalla vain julkaisuja
T1	UML	Kyllä	Ei	Kyllä
T2	UML	Ei	Ei	Kyllä
T3	UML, M & A	Kyllä	Ei	Ei
T4	UML	Ei	Ei	Kyllä
X1	UML	Ei	Jonkin verran	Kyllä
X2	UML, M & A, OOA	Kyllä	Ei	Ei
X3	UML, M & A	Kyllä	Hieman	Ei

Taulukosta voidaan havaita, että ne opiskelijat, joille Maler & El Andaloussin menetelmä oli ennen projektityön alkua tuttu, olivat yhtä mieltä siitä, ettei RASKE-menetelmää voi oppia pelkästään lukemalla aiheesta kirjoitettuja julkaisuja. Loput opiskelijoista, joille Maler & El Andaloussin menetelmä ei ollut tuttu, olivat vastakkaista mieltä; RASKE-menetelmän pystyi oppimaan pelkästään lukemalla julkaisuja. Tämä havainto herättää kysymyksen siitä, kuinka tarkasti muut opiskelijat itse asiassa paneutuivat RASKE-menetelmään.

RASKE-menetelmä soveltaa osia Maler & El Andaloussin menetelmää dokumenttikomponenttien mallintamisessa. Ne opiskelijat, jotka kertoivat Maler & El Andaloussin menetelmän olleen heille tuttu ennen projektityön alkua, antoi-

vat mielenkiintoisia vastauksia RASKE-menetelmän malleja koskien. Vastaukset on koottu alla olevaan taulukkoon (TAULUKKO 5).

TAULUKKO 5. Maler & El Andaloussin menetelmään tutustuneiden opiskelijoiden mielipiteitä RASKE-menetelmän malleista

Opiskelija	Oliko komponenttikaavioista apua DTD määrittämissä laadittavissa?	Helpoin malli hahmottaa	Vaikein malli hahmottaa	Epäselvä malli projektissasi
Tag2IT	Hieman	Tilasiirtymäkaavio	Komponenttikaaviot	Ei ollut
XooZoo	Hieman	Tilasiirtymäkaavio	Komponenttaulukko ja -kaaviot	Komponenttaulukko
XooZoo	Ei ollut	Komponenttikaaviot	Syöte/tuotosmalli	Komponenttaulukko ja -kaaviot

On mielenkiintoista havaita, että Maler & El Andaloussin menetelmästä tutut Elm-mallin mukaiset komponenttikaaviot ovat olleet joko vaikeita malleja tai epäselviä. Ainoastaan yksi opiskelija kertoi komponenttikaavioiden olleen helppo hahmottaa kaikista käyttämistään RASKE-menetelmän malleista. Erityisesti XooZoo-projektiryhmään kuuluneiden vastauksissa tulee ottaa huomioon projektin raportoimat vaikeudet loogista rakennetta mallinnettaessa. Vaikeuksia projektiryhmälle tuotti erityisesti opinto-oppaan sisällön laajuus. Siksi heidän vastauksissaan komponenttikaaviot ja komponenttaulukko esiintyvät joko vaikeimmista tai epäselvistä malleista. On myös mielenkiintoista havaita, ettei yksikään opiskelija, jolle Maler & El Andaloussin menetelmä oli entuudestaan tuttu, kokenut laadittujen komponentti hierarkiamallien antaneen apua DTD-määrittämissä laadittavissa.

Opiskelijaprojekteissa RASKE-menetelmän mukaisia malleja mallinnettiin Microsoft Word, Microsoft Visio sekä Altova XmlSpy -ohjelmilla, joista viimeiseksi mainittua käytettiin vain XooZoo-projektissa. Vastanneista enemmistö oli sitä mieltä, että työkalut olivat riittäviä eikä RASKE-menetelmällä siten tarvitse olla tiettyä mallintamisvälinettä.

6.3 RASKE-menetelmän mallit

Haastatteluissa kysyttiin opiskelijoiden mielipidettä yksittäisistä RASKE-menetelmän malleista. Opiskelijoilta tiedusteltiin mallien soveltamisen helppoutta omassa projektissaan sekä laadittujen mallien selkeyttä. Vastaukset olivat selvästi projektiriippuvaisia. XooZoo-projektillaiset, joiden sovellusalue oli selvästi Tag2IT-projektin kohdealuetta laajempi, olivat huomattavasti kriittisempiä vastauksissaan.

Molemmat projektit aloittivat kartoituksen kohdealueen mallintamisella kontekstikaavioon. Viisi seitsemästä opiskelijasta kertoi kontekstikaavion olevan selkeä. Tag2IT-projektillaisten mielestä malli oli helppo, yhden vastanneen mielestä jopa liian pelkistetty. XooZoo-projektillaisten vastauksista kävi ilmi, että kohdealueen ollessa laaja ja sen hahmottamisen ollessa hankalaa, on kontekstikaavion laatiminen haastavaa. Yksi XooZoo-projektiryhmään kuulunut opiskelija sanoikin, että *"mallin käyttö ei ollut vaikeaa vaan oman kohdealueen hahmotus oli. Asiakkaalla oli eri näkemyksiä sekä eri termejä samoille asioille."*

Kummankin projektin tapauksessa kohdealueen roolit mallinnettiin roolitaulukoon. Taulukko koettiin projektista riippumatta selkeäksi sekä helposti laa-

dittavaksi, tosin kritiikkiä saivat osakseen rooleista käytettävät termit. Yhdelle vastanneista oli alkuun hankalaa hahmottaa sitä, kuinka tarkalla tasolla taulukoon tulisi loppujen lopuksi roolit kuvata. Kommentteja roolitaulukosta olivat muun muassa *”On helppo tehdä ja taulukko on erittäin selkeä”* sekä *”Taulukko voisi olla parempi. Termit voisivat olla paremmat. Mallista saa kuitenkin selvän”*.

Syöte/tuotos-mallin koki selkeäksi vain kaksi haastatteluun vastanneista opiskelijoista. Oli mielenkiintoista havaita, että mallia koskien vastauksista löytyi kaksi täysin päinvastaista mielipidettä. Kun yksi opiskelija kertoi mallin omaksu-
misen olleen hänelle hankalaa sekä luulevansa mallin jäävän kaikkein epäselvimmäksi ulkopuolisille tahoille, toinen taas sanoi, että malli on selkeä sekä helposti omaksuttavissa. Tähän opiskelija vielä lisäsi, että *”luulisi ulkopuolisenkin ymmärtävän mallia katsoessaan mistä on kyse.”*

Syöte/tuotos-mallin koki vaikeimmaksi malliksi kolme opiskelijaa seitsemästä, joka oli vastausten enemmistö. Helpoimmaksi mallin taas koki kaksi opiskelijaa seitsemästä. Opiskelija, joka koki syöte/tuotos-mallin helpoimmaksi, kommentoi mallia seuraavasti: *”Kun idea selvisi, malli oli helppo tehdä”*. Sama opiskelija totesi myös sen, että hänen mielestään syöte/tuotos-mallissa oli kaikkein tärkein tietosisältö. Syöte/tuotos-malli ei ollut yhdenkään haastatteluun vastanneiden mukaan projektissa tarpeeton, eikä sen tarkoitus tai hyödyllisyys jäänyt kenellekään epäselväksi. Sen sijaan sen notaatioon yksi opiskelija kommentoi seuraavaa: *”Mallissa häiritsee liialliset nuolet. Tuotos-nuolen voisi laittaa tehtävien väliin. Mallissa on liikaa toistoa, saman nuolet ja tekstit toistuvat.”*

Myös dokumenttien elinkaarta kuvaava tilasiirtymäkaavio sai osakseen sekä myönteisiä että kielteisiä mielipiteitä. Projektiryhmien väliltä löytyi eroja, kun

Tag2IT-projektiin kuulunut kertoi tilasiirtymäkaavion olevan ”*vielä selkeämpi kuin syöte/tuotos-malli*”, kun taas XooZoo-projektiryhmään kuulunut opiskelija mainitsi tilasiirtymäkaavion olevan vaikeampi tehdä ja sisäistää kuin syöte/tuotos-malli. Yksi vastanneista kaipasi tilasiirtymäkaavion notaatioon muutosta; kaavion tulisi mahdollistaa myös taaksepäin menevä nuoli. Kommentti on mielenkiintoinen, sillä RASKE-menetelmän notaatio kyllä mahdollistaa tilasta palaavan nuolen, kuten jo luvussa 5.2 kerrottiin. Kysyttäessä vaikeinta hahmotettavissa olevaa mallia, tilasiirtymäkaavio ei saanut kannatusta lainkaan. Sen sijaan helpoimmaksi malliksi sen mainitsi kaksi opiskelijaa, joista toinen kertoi käyttäneensä tilasiirtymäkaaviota ennen projektia.

Kysyttäessä oliko joku malleista erityisen käyttökelpoinen projektityössä, syöte/tuotos-malli sekä tilasiirtymäkaavio olivat selkeästi yleisimmät vastaukset. Pelkän syöte/tuotos-mallin mainitsi kaksi opiskelijaa. Mallia kommentoitiin seuraavasti: ”*Mallissa prosessi tuli ilmi ja prosessien mallintaminen oli helppoa ja selkeää*”. Toinen kommentoi mallia siten, että syöte/tuotos mallin avulla saatiin selkeästi kuvattua laatimisprosessia, mikä oli projektin tehtävä. Yksi opiskelijoista vastasi sekä syöte/tuotos-mallin että tilasiirtymäkaavion olleen erityisen käyttökelpoisia malleja. Yksi opiskelijoista taas kertoi vaihtoehtoisesti joko syöte/tuotos-mallin tai tilasiirtymäkaavion olleen erityisen käyttökelpoinen. Syöte/tuotos-malli koettiin käyttökelpoiseksi molemmissa projektiryhmissä, kun taas tilasiirtymäkaavion käyttökelpoisuus sai kannatusta vain XooZoo-projektiin osallistuneilta. Kaksi vastanneista totesi, ettei mikään malleista ollut selkeästi käyttökelpoisempi kuin muut.

Projektit mallinsivat dokumenttikomponentit sekä komponenttitaulukoihin että komponenttikaavioihin. Kun Tag2IT-projekti mallinsi komponenttikaaviot Elm-

mallin notaation mukaan, XooZoo-projekti päätyi hyödyntämään Altova XmlSpy -ohjelmalla tuotettuja kaavioita. XooZoo-projektiryhmään kuuluneiden opiskelijoiden yleinen mielipide komponenttitaulukkoa ja -kaavioita kohtaan oli kriittinen. Malleja ei koettu helpoiksi, selkeiksi taikka edes hyödyllisiksi.

Tag2IT-projektiryhmään kuuluneiden opiskelijoiden mielipiteet komponenttitaulukkoa ja komponenttikaavioita kohtaan poikkesivat selvästi XooZoo-projektiin kuuluneiden opiskelijoiden vastauksista. Taulukkoa pidettiin jopa yksikertaisena sekä Elm-mallin notaation mukaisia komponenttikaavioita selkeämmäksi. Vastauksien selkeää eroavaisuutta projektiryhmien kesken voidaan selittää sillä, että XooZoo-projektissa loogisen rakenteen kuvaus oli selkeästi laajempi kokonaisuus kuin Tag2IT-projektissa sekä itse kohdealue selvästi vaikeammin hahmotettavissa.

Haastatteluihin vastanneilta kysyttiin yleisesti sitä, mikä malleista oli helpoin ja mikä vaikein hahmottaa. Vastaukset hajaantuivat molemmissa tapauksissa melko tasaisesti eri mallien kesken (TAULUKKO 6).

TAULUKKO 6. Mielipiteet malleista opiskelijoittain

Opiskelija	Helpoin malli hahmottaa	Vaikein malli hahmottaa
T1	Komponenttikaaviot	Syöte/tuotos-malli
T2	Roolitaulukko	Syöte/tuotos-malli
T3	Kontekstikaavio	Komponenttikaaviot
T4	Tilasiirtymäkaavio	Komponenttikaaviot
X1	Komponenttikaaviot	Syöte/tuotos-malli
X2	Tilasiirtymäkaavio	Komponenttitaulukko ja komponenttikaaviot
X3	Syöte/tuotos-malli	Komponenttitaulukko ja komponenttikaaviot

Taulukosta huomataan, ettei kukaan opiskelijoista ei kertonut tilasiirtymäkaavion tai roolitaulukon olleen vaikeimpia malleja hahmottaa. Sen sijaan helpoimmin hahmotettavista malleissa ne saivat kannatusta. XooZoo-projektiin osallistuneet opiskelijat kertoivat, että loogisen rakenteen kartoitus soveltaen komponenttitaulukkoa ja komponenttikaavioita, oli vaikeaa. Tag2IT-projektillaisten keskuudesta ei puolestaan löytynyt yksittäistä mallia, jonka soveltamisessa olisi ollut vaikeuksia. Vain yksi vastanneista totesi, ettei minkään mallin soveltamisessa ollut vaikeuksia.

Kysyttäessä mitä malleja projektiryhmät päätyivät muuttamaan, enemmistö kertoi muuttaneensa syöte/tuotos-mallia ja tilasiirtymäkaaviota. Erityisesti XooZoo-projektiin osallistuneet kertoivat muokanneensa syöte/tuotos-mallia. Mallissa yhdistettiin nyky- ja tavoitetila siten, että tulevaisuudessa tapahtuvaa aktiviteettia merkattiin värillä. XooZoo-projekti myös numeroi ne syöte/tuotosmallin aktiviteetit, jotka havaittiin ongelmallisiksi opinto-oppaan laatimisessa. Syöte/tuotos-malli, joka kuvaa opinto-oppaan laatimista tiedekunnan näkökulmasta, sisältää numeroituja aktiviteetteja viisi kappaletta (Liite 5).

Tag2IT-projekti muokkasi komponenttitaulukkoa ja komponenttikaavioita siten, että pöytäkirjoihin kuuluvia komponentteja eroteltiin muista komponenteista värittämällä komponentti. Periaate oli siis sama kuin XooZoo-projektin syöte/tuotosmallin soveltamisessa.

Opiskelijaprojektit eivät käyttäneet kaikkia RASKE-menetelmästä löytyviä malleja. Tarpeettomaksi malliksi koettiin D-R -kaavio ja roolitaulukko. Yksi vastaajista ilmoitti komponenttitaulukon olleen turha. Kysyttäessä siitä, oliko RASKE-

menetelmässä jokin malli, jonka tarkoitus tai hyödyllisyys ei selvinnyt, suurinta kannatusta saivat komponenttikaaviot ja komponenttitaulukko. Enemmistö näistä vastauksista tuli XooZoo-projektiin kuuluneilta opiskelijoilta. Sen sijaan kaikki Tag2IT-projektiin kuuluneet opiskelijoiden enemmistö vastasi, ettei minäkään mallin hyödyllisyys tai tarkoitus jäänyt epäselväksi.

6.4 RASKE-menetelmä projektityöskentelyssä

Kun mielipiteitä RASKE-menetelmän mukaisiin malleihin oli kartoitettu, kysyttiin opiskelijoilta vielä mielipiteitä RASKE-menetelmän sopivuudesta omaan projektiin. Jopa viisi vastaajaa seitsemästä koki RASKE-menetelmän sopineen omaan projektiin hyvin. Kaksi vastaajaa koki menetelmän sopineen tyydyttävästi; näistä molemmat olivat XooZoo-projektiryhmästä. Kysyttäessä sitä, kuinka opiskelijat tulkitsivat asiakkaansa ajattelevan RASKE-menetelmällä tehdyistä tuloksista, yleinen mielipide puolsi sitä, että asiakastahot olivat tuloksiin tyytyväisiä.

Molemmissa opiskelijaprojekteissa käytettiin RASKE-menetelmää ensimmäistä kertaa. Kun opiskelijoilta kysyttiin, mitä he nyt tekisivät toisin, jos he aloittaisivat asiakirja-analyysin uudestaan, vastaukset olivat lähes samankaltaisia projektista riippumatta. Vastaajat kertoivat, että he pitäisivät enemmän kontakteja asiakkaan suuntaan; sekä kokouksia että haastatteluita tulisi järjestää enemmän. Yleisesti oltiin myös sitä mieltä, että tiiviimpi yhteistyö projektiryhmän sisällä olisi paikallaan. Projekteissa kartoitustyöt oli jaettu RASKE-menetelmän mallien mukaisesti siten, että osa ryhmää laati tietyt mallit, joten tämä voi osaltaan selittää sitä, miksi haastatteluiden perusteella ei löytynyt esimerkiksi helpointa

tai vaikeinta mallia. Yksi vastanneista kertoi, että hän tutustuisi itse RASKE-menetelmään paremmin.

Projektin päätyttyä enemmistö haastatteluun vastanneista kertoi kiinnostuneensa rakenteisista dokumenteista enemmän. Yksi vastanneista kertoi kiinnostuksensa vähentyneen. Viisi opiskelijaa seitsemästä lähtisi myös vastaisuudessa mukaan samankaltaisiin projekteihin. Näistä viidestä neljä käyttäisi tällaisessa tapauksessa RASKE-menetelmää ja yksi aikoi tutkia muita vaihtoehtoisia menetelmiä.

Kaiken kaikkiaan haastatteluista kävi ilmi, että molemmissa projekteissa RASKE-menetelmän käyttö koettiin tarpeelliseksi ja hyödylliseksi. Kartoitettaessa yksittäisiä asiakirja-analyysin aikana laadittavia malleja, mielipiteiden hajonta oli voimakasta. Vastauksista käy siten selvästi ilmi, että työtä oli jaettu mallikohtaisesti molemmissa projekteissa; se malli, jota kukin opiskelija henkilökohtaisesti oli mallintamassa, nousi selkeydessä ja helppoudessa ylitse muiden. Näin ei ollut kuitenkaan komponenttikaavioiden ja komponenttitaulukoiden kohdalla, jotka nousivat haastatteluissa selkeästi poikkeuksellisiksi malleiksi. XooZoo-projektiryhmä koki kohdealueensa laajuuden takia opintooppaan loogisen rakenteen kartoituksen hyvin hankalaksi ja aikaa vieneeksi tehtäväksi, jonka lopullinen hyödyllisyys jäi projektiryhmän sisällä hieman epäselväksi. Siten XooZoo-projektiin osallistuneiden opiskelijoiden vastauksista on havaittavissa yhtenäistä kritiikkiä juuri komponenttitaulukkoa ja komponenttikaavioita kohtaan. Tämä yhtenäisyys nosti luonnollisesti näiden mallien esiintymiskertoja haastattelun tuloksia analysoidessa, vaikka Tag2IT-projektiryhmään osallistuneiden opiskelijoiden haastatteluissa nämä mallit eivät erityisesti korostuneet.

RASKE-menetelmän soveltaminen koettiin projekteissa myönteisesti. Molempien projektien tapauksessa menetelmän käyttö edesauttoi rakenteistamistyön etenemistä. Haastatteluihin osallistuneet opiskelijat kertoivat yksimielisesti asiakkaidensa pitäneen RASKE-menetelmällä tehdyistä tuloksista.

7 JOHTOPÄÄTÖKSIÄ

RASKE-menetelmä on kehitetty dokumenttien hallinnan kehittämisprojektien tueksi. Lukukautena 2004 - 2005 Jyväskylän yliopiston informaatioteknologian tiedekunnassa kaksi opiskelijaprojektia sai aiheekseen tiedekunnan dokumenttien hallinnan kehittämistä koskevan projektiaiheen. Molemmat projektit päätyivät soveltamaan työssään RASKE-menetelmää. Tilanne oli ainutkertainen, sillä tietävästi aikaisemmin RASKE-menetelmää eivät olleet käyttäneet kuin vain menetelmän kehitystyössä mukana olleet tahot.

Tag2IT- ja XooZoo-projekteihin osallistuneet opiskelijat saivat koulutusta RASKE-menetelmästä ennen projektityön alkamista. Koulutusta tarjosi tietojenkäsittelytieteen laitoksen lehtori, joka piti projektiryhmille yhteisen koulutustilaisuuden. Koulutuksen jälkeen projektiryhmät kokivat RASKE-menetelmän sopivan projektityönsä tueksi, joten menetelmä päätettiin ottaa käyttöön.

Tämän tutkielman empiirisessä osuudessa on selvitetty, kuinka opiskelijaprojektit sovelsivat RASKE-menetelmää. Erityisesti tutkimuksen kohteena ovat olleet RASKE-menetelmän asiakirja-analyysin aikana laadittavat mallit, joilla kuvataan sovellusalueen rooleja, prosesseja ja dokumentteja. RASKE-menetelmän mukaiset mallit pohjautuvat ICN-malleihin sekä OOA- ja Maler & El Andaloussin menetelmään. Mallit kehitettiin mahdollisimman selkeiksi ja niiden tarkoituksena on kuvata kohdealueella tapahtuvaa oikeaa toimintaa; sitä, miten työ todellisuudessa tehdään.

Tutkielmassa esitettyjen opiskelijaprojektien tapauksessa ICN-mallit eivät olleet tuttuja yhdellekään opiskelijalle. OOA-menetelmän koki tutuksi vain yksi opiskelija seitsemästä ja Maler & El Andaloussin menetelmään oli tutustunut kolme opiskelijaa seitsemästä. Tämä ei kuitenkaan tuntunut olevan este RASKE-menetelmän mallien laatimiselle, koska haastatteluiden mukaan RASKE-menetelmä omaksuttiin projekteissa melko hyvin. Tätä väitettä tukevat myös projektien laatimat raportit, joissa RASKE-menetelmän asiakirja-analyysin aikaiset tulokset on nähtävissä. RASKE-menetelmää ei seurattu projekteissa sellaisenaan, vaan menetelmää sovellettiin projektikohtaisesti.

Tag2IT-projekti mallinsi tiedekunnan toimiston laatimat esittelylistan ja pöytäkirjan erikseen. Koska pöytäkirja laaditaan esittelylistasta, sitä koskevat mallit ovat suppeampia kuin esittelylistasta laaditut samat mallit. Lisäksi pöytäkirjalle laaditut komponenttikaaviot olivat lähes identtisiä esittelylistan vastaaviin verrattuna. Kaavioihin lisättiin pöytäkirjalle ominaiset komponentit ja niitä merkittiin väreillä (sininen). Lisäksi ne komponentit, joiden sisältö muuttui siirryttäessä esittelylistasta pöytäkirjaan, nimettiin lihavoiduin tekstein pöytäkirjalle laadituissa komponenttikaavioissa. Herääkin siis kysymys siitä, olisiko Tag2IT-projekti voinut mallintaa vain pöytäkirjan malleihinsa sen sijaan, että jokaisesta mallista ja kaaviosta laadittiin kaksi toisiaan lähellä olevaa versiota.

Myös XooZoo-projektin tapauksessa RASKE-menetelmän malleja sovellettiin sovellusalueen tarpeita vastaaviksi. Erityisesti prosessimalleihin päädyttiin mallintamaan sellaisia aktiviteetteja ja tiloja, jotka eivät ole nykyisessä oppaan laatimisprosessissa mukana. Näissä malleissa siis otetaan kantaa jo siihen, millainen sovellusalueelle kehitettävän tavoitetilan tulisi olla. Tämä vaikutti mallien notaatioihin; malleja sovellettiin siten, että ne ottivat huomioon myös tavoit-

tetilan. Siten RASKE-menetelmän notaatiota ei malleissa täysin seurattu. RASKE-menetelmän notaatio tarjosi kuitenkin projekteille pohjan, jota vasten mallintaa sovellusalueiden nykytilaa. Kuitenkin projektit päätyivät muuttamaan jollain tapaa lähes jokaista mallia, jotta ne sopisivat projektien kartoitus-työn tarpeisiin. Erityisesti syöte/tuotos-malleja, tilasiirtymäkaaviota sekä komponenttikaavioita muutettiin projektien tarpeiden mukaan.

Verrattaessa projektien laatimia malleja toisiinsa huomataan, että projektien sovellusalueet poikkeavat toisistaan erityisesti laajuuden suhteen. Eroavaisuuden korostuvat erityisesti projektien laatimissa prosessimalleissa eli syöte/tuotos-malleissa ja tilasiirtymäkaavioissa. Tag2IT-projektin prosessimallit ovat XooZoo-projektin laatimiin prosessimalleihin verrattuna huomattavasti suoralinjaisempia ja yksinkertaisempia. XooZoo-projektin tapauksessa opinto-oppaan laatimiseen osallistuu tiedekunnan sisällä useita eri tahoja ja itse laatimisprosessi käy läpi useita eri vaiheita, ennen kuin opinto-opas on valmis. Tag2IT-projektin sovellusalueen tapauksessa prosesseihin osallistuu keskimäärin noin viisi henkilöä.

Kaikkia RASKE-menetelmään kuuluvia malleja ei koettu projekteissa hyödyllisiksi. Esimerkiksi dokumentteja ei mallinnettu D-R -kaavioon (ks. KUVIO 7, s. 29) kummankaan projektin tapauksessa. Tag2IT-projektin tapauksessa malli olisi ollut sinänsä turha, koska projekti mallinsi vain esittelylistan ja pöytäkirjan suhdetta toisiinsa. Tosin tiedekunnan toimisto saa esittelylistaan sisältöä eri lomakkeilta ja asiakirjoilta, mutta nämä tapaukset oli rajattu projektin työstä pois kokonaan. Myöskään XooZoo-projekti ei käyttänyt D-R -kaaviota lainkaan. Toinen malli, jota projektit eivät huomioineet asiakirja-analyysin aikana, on roolimalli. Molempien projektien tapauksessa roolit mallinnettiin roolitaulukoihin.

Tähän on voinut vaikuttaa se, että projektien saamissa mallidokumenteissa ei ollut selkeää esimerkkiä roolimallista.

Tehtyjen haastattelujen pohjalta on vedettävissä johtopäätös, jonka mukaan RASKE-menetelmä on menetelmänä selkeä ja siten sen soveltaminen onnistuu, vaikkei menetelmää olisi ennen käyttänytkään. Tässä tutkielmassa esiteltyjen opiskelijaprojektien tapauksissa RASKE-menetelmä koettiin myös hyödylliseksi. Myös projektien laatimat raportit asiakirja-analyysin aikana laadittuine malleineen tukevat tätä johtopäätöstä.

Haastatteluissa opiskelijoilta tiedusteltiin erityisesti asiakirja-analyysin aikana tuotettavista malleista ja niiden laatimisen helppoudesta. Haastatteluiden perusteella haluttiin myös tietää ovatko RASKE-menetelmän mukaiset mallit selkeitä. Vastauksista löytyi paljon hajontaa; usein opiskelija koki sen mallin selkeäksi ja helpoksi, jota oli ollut itse laatimassa.

Eniten mielipiteitä opiskelijoiden keskuudessa herätti syöte/tuotos-malli, joka koettiin joko erittäin vaikeaksi laatia ja ymmärtää tai vastaavasti kaikkein selkeimmäksi malliksi. XooZoo-projektin tapauksessa komponenttitaulukko koettiin liian hankalaksi projektin sovellusalueen laajuuden ja epäselvyyden takia. Myös komponenttikaaviot saivat samanlaista kritiikkiä osakseen. Kaavioiden hyödyllisyyttä ei ymmärretty eivätkä XooZoo-projektiin kuuluneet opiskelijat kokeneet niistä olleen apua DTD-määrittämissä suunniteltaessa ja laadittaessa.

Myös tilasiirtymäkaaviota kritisoitiin. Erityisesti kaavion kaivattiin mahdollis-tavan ehdollisen siirtymisen tilasta toiseen. XooZoo-projektin tapauksessa ehdollista siirtymistä kuvattiin nuolella, joka palasi samaan, edellisiin tai tuleviin

tiloihin (ks. Liite 7). Myös palaavaa nuolta edelliseen tilaan kaivattiin. RASKE-menetelmän notaation mukaan ehdollisuus tilasiirtymäkaaviossa on mahdollista. Herääkin kysymys siitä, kuinka hyvin menetelmään todellisuudessa perehdyttiin.

Kun opiskelijoilta kysyttiin mikä malleista oli vaikein hahmottaa, neljä opiskelijaa seitsemästä vastasi komponenttitaulukko ja/tai komponenttikaaviot. Kaksi näistä kuului Tag2IT-projektiryhmään ja kaksi XooZoo-projektiryhmään. Yhdelle näistä neljästä opiskelijasta Maler & El Andaloussin menetelmä oli entuudestaan tuttu. Vastaavasti, kun kysyttiin mikä malleista oli helpoin hahmottaa, mikään malleista ei noussut ylitse muiden, mutta yksi opiskelija, jolle Maler & El Andaloussin menetelmä oli entuudestaan tuttu, vastasi komponenttikaavioiden olevan helpoimmin hahmotettavissa. Tästä voidaan vetää johtopäätös, ettei tässä tapaustutkimuksessa Maler & El Andaloussin menetelmän tuntemisella sinänsä ollut vaikutusta siihen, kuinka RASKE-menetelmän soveltaminen onnistui ja millaiseksi menetelmäksi opiskelijat sen kokivat.

Maler & El Andaloussin menetelmän tuntemisella oli kuitenkin vaikutusta siihen, kuinka opiskelijat kokivat RASKE-menetelmän olevan opittavissa. Kun opiskelijoilta kysyttiin, voiko RASKE-menetelmän omaksua pelkästään lukemalla aiheesta kirjoitettuja julkaisuja, kaikki kolme opiskelijaa seitsemästä, jolle Maler & El Andaloussin menetelmä oli entuudestaan tuttu, kertoivat olevansa sitä mieltä, ettei RASKE-menetelmää heidän mielestään voi omaksua pelkästään lukemalla menetelmästä. Sen sijaan loput neljä opiskelijaa, joille Maler & El Andaloussin menetelmä ei ollut entuudestaan tuttu, tunsivat, että menetelmän voi oppia pelkästään lukemalla siitä. Tämä havainto on mielenkiintoinen ja se herättää kysymyksen siitä, ovatko ne opiskelijat, joille Maler & El Andaloussin

menetelmä on ollut entuudestaan tuttu, olleet projektiryhmässä sellaisessa asemassa, että RASKE-menetelmän soveltaminen ja mallien laatiminen on ollut enemmän heidän vastuullaan. Kuitenkin molemmissa opiskelijaprojekteissa RASKE-menetelmän malleja jaettiin ryhmän kesken laadittaviksi ja siten useampi opiskelija osallistui sovellusalueen mallintamiseen. Siten ei voida päätellä johtopäätöstä siitä, että Maler & El Andaloussin menetelmään tutustuneet opiskelijat olisivat olleet enemmän vastuussa asiakirja-analyysin etenemisessä kuin muut projektiryhmien opiskelijat.

RASKE-menetelmää jouduttiin siis soveltamaan, jotta se saatiin vastaamaan projektien sovellusalueen vaatimuksia. Tämä ei ole yllättävä tulos, sillä esimerkiksi Tolvanen (1998) on väitöskirjassaan todennut, että menetelmiä joudutaan aina soveltamaan, jotta ne vastaisivat organisaation tai projektiryhmän tarpeita. Tolvasen mukaan menetelmien soveltaminen voi johtaa usein paikallisten menetelmien kehittämiseen, jolloin organisaatio ottaa olemassa olevista menetelmistä osia, soveltaa niitä, ja kehittää niistä oman menetelmän.

Vaikka tässä tutkielmassa esitetyt opiskelijaprojektit sovelsivat RASKE-menetelmää tarpeidensa mukaan, ei uutta menetelmää kuitenkaan kehitetty. Tag2IT- ja XooZoo-projektit sovelsivat projektityössään RASKE-menetelmää, mutta ei siinä määrin, että projektien aikana olisi luotu uusi (paikallinen) menetelmä. Sen sijaan kun RASKE-menetelmää alettiin vuonna 1994 kehittää, voidaan huomata, että juuri tuolloin oli kyse paikallisen menetelmän kehitystyöstä. Tuolloin dokumenttien hallinnan kehittämisen tueksi laaditut menetelmät eivät vastanneet niitä tarpeita, joita RASKE-projektissa koettiin tarpeellisiksi erityisesti asiakirja-analyysin kohdalla, joten projektin aikana kehitettiin uusi menetelmä, RASKE-menetelmä.

RASKE-menetelmä oli menetelmänä opiskelijaprojekteille sopiva. Menetelmän avulla projektit saivat suoritettua sovellusalueidensa kartoittamisen. Koska kummassakaan projektissa ei ollut mukana sellaisia henkilöitä, jotka olisivat olleet mukana RASKE-menetelmän kehittämistyössä, voidaan vetää johtopäätös, jonka mukaan RASKE-menetelmä on menetelmänä selkeä, että sen omaksuminen ja käyttöönotto onnistuu sitä ennen tuntemattomilta. Opiskelijaprojektit sovelsivat RASKE-menetelmästä vain sen ensimmäistä vaihetta, asiakirja-analyysia, jonka avulla opiskelijat saivat sovellusalueistaan selkeämmän kuvan. Asiakirja-analyysi oli siis vaiheena erittäin tärkeä.

8 YHTEENVETO

Dokumenteilla on merkittävä rooli organisaatioiden toiminnoissa, koska dokumentteihin on tallennettu organisaatioiden toiminnan kannalta keskeiset tiedot (Salminen 1996). Nykyään organisaatioiden dokumentit ovat suuressa määrin digitaalisessa muodossa. Dokumenttien tiedostomuodot ja erilaiset, usein muuttuvat järjestelmät aiheuttavat kuitenkin ongelmia mm. tiedonhakuun ja tiedon saatavuuteen. Näiden ongelmien johdosta useat organisaatiot ovat aloittaneet laajat projektit dokumenttien hallinnan kehittämiseksi. (Salminen, Lyytikäinen & Tiitinen 1999) Yksi ratkaisu edellä mainittujen ongelmien ratkaisemiseksi on siirtyä ns. rakenteisten dokumenttien käyttöön, jotka ovat usein joko SGML- tai XML-dokumentteja. Rakenteisissa dokumenteissa on sekä tietokoneen että ihmisen tulkittavissa oleva rakennemäärittely. (Tiitinen, Päivärinta, Salminen & Lyytikäinen 1997: Salminen, Tiitinen & Lyytikäinen 1999)

Dokumenttien hallintaa kehittävä projekti on usein aikaa vievä ja vaativa tehtävä, joka vaatii onnistuakseen syvällisen asiakirja-analyysin. Asiakirja-analyysin kautta saadaan kohdealueen ongelmakohdat esiin, joiden pohjalta on hyvä etsiä uusia ja parempia ratkaisuja dokumenttien hallinnan kehittämiseksi. RASKE-menetelmä on Jyväskylän yliopiston, eduskunnan sekä valtioneuvoston yhteistyöhankkeessa, RASKE-projektissa, kehitetty menetelmä rakenteisten asiakirjastandardien kehitystyöhön. (Salminen, Lyytikäinen & Tiitinen 1999)

Tässä tutkielmassa oli tarkoituksena selvittää mitä tietojärjestelmien menetelmillä tarkoitetaan sekä miten näitä menetelmiä voidaan arvioida. Arvioinnin tueksi esiteltiin simpukankuorimalli (Tolvanen 1998), jonka mukaan menetelmätietämyksellä on kuusi tasoa. Mikäli menetelmä on eheä, siitä on löydettävissä

sä simpukankuorimallin kaikki tasot. Lisäksi tämän tutkielman tarkoituksena oli kertoa millainen on asiakirja-analyysiin kehitetty RASKE-menetelmä. RASKE-menetelmää myös arvioitiin simpukankuorimallin mukaan. Tutkielman tavoitteena oli vastata kysymyksiin, millainen on RASKE-menetelmä, miten RASKE-menetelmän soveltaminen onnistui Jyväskylän yliopiston opiskelijaprojekteissa sekä oliko RASKE-menetelmän asiakirja-analyysin aikana laadittavat mallit helposti omaksuttavissa. Tämän lisäksi tutkielmassa pyrittiin kartoittamaan niitä kokemuksia, joita opiskelijat kokivat RASKE-menetelmän soveltamisesta omassa projektissaan.

Lukukautena 2004 - 2005 kaksi Tietojenkäsittelytieteiden laitoksen Projektin johtaminen -kurssille osallistunutta opiskelijaprojektia sovelsi työssään RASKE-menetelmää. Toinen projekteista oli Tag2IT-projekti, jonka tehtävänä oli rakenteistaa Jyväskylän yliopiston informaatioteknologian tiedekunnan toimiston laatimat kokousasiakirjat. Toinen projekteista oli XooZoo-projekti, jonka tavoitteena oli rakenteistaa informaatioteknologian tiedekunnan vuosittain julkaisemaan opinto-opasta. Molemmissa projekteissa käytännön toteutus tuli tehdä XML-kieltä toteuttaen.

Tutkimuksen tarkoituksena oli selvittää kuinka nämä opiskelijaprojektit sovelsivat RASKE-menetelmää ja kuinka helppoa menetelmän omaksuminen opiskelijoille oli. Projekteihin osallistuneita opiskelijoita haastateltiin lähinnä RASKE-menetelmän asiakirja-analyysin aikana tuotettavien mallien soveltamisesta.

RASKE-menetelmä sisältää joukon erilaisia malleja kohdealueen yleiskuvauksen, roolien, prosessien ja dokumenttien esittämiseen. Molempien opiskelijaprojektien tapauksissa RASKE-menetelmästä käytettiin samoja malleja projektien sovellusalueiden kuvaamiseen, tosin malleja sovellettiin projektikohtaisesti.

Eniten malleja sovellettiin käyttämällä malleissa värejä. Tag2IT-projekteissa väreillä merkittiin kahden erilaisen dokumenttirakenteen eroja. XooZoo-projektissa väreillä merkittiin eroja nykyisen ja tulevan opinto-oppaan laatimisprosesseissa. Menetelmän soveltaminen ei ole kuitenkaan ainutlaatuista sillä usein menetelmät eivät vastaa organisaation tai työyhteisön tarpeita. Siksi menetelmistä voidaan kehittää ns. paikallisia menetelmiä vastaamaan oman työn tarpeita. (Tolvanen 1998)

Tutkimuksen tuloksina oli kuvauksia RASKE-menetelmän soveltamisesta uudellisissa ympäristöissä sekä kuvausta siitä, kuinka RASKE-menetelmän soveltaminen opiskelijaprojekteissa onnistui. Haastatteluista saatujen tietojen perusteella on vedettävissä johtopäätös, jonka mukaan RASKE-menetelmä on menetelmänä selkeä ja siten sen soveltaminen onnistuu, vaikkei menetelmää olisi ennen käyttänytkään. RASKE-menetelmä koettiin myös hyödylliseksi projektityön tukena. Koska kyseessä on tapaustutkimus, yleistyksiä RASKE-menetelmän soveltamisesta ei voida tehdä.

Yleisesti RASKE-menetelmän soveltaminen koettiin projekteissa myönteisesti. Molempien projektien tapauksessa menetelmän käyttö edesauttoi rakenteistamistyön etenemistä ja enemmistö opiskelijoista olisi valmis käyttämään menetelmää uudestaan samankaltaisissa projekteissa. Tässä tutkielmassa haastatteluihin osallistuneiden opiskelijoiden määrä oli kuitenkin pieni. Lisäksi XooZoo-projektista kaikki opiskelijat eivät osallistuneet haastatteluihin. Mikäli nämä opiskelijat olisi saatu mukaan, olisi haastatteluista voinut nousta vielä joitain mielenkiintoisia näkökulmia. Toisaalta ne opiskelijat, jotka haastatteluihin osallistuivat, olivat keskeisessä osassa mallien laatimisessa ja heidän tietotaitonsa saatiin haastatteluissa tallennettua.

Jatkossa olisi mielenkiintoista tutkia RASKE-menetelmän soveltamista muissa dokumenttien rakenteistamisprojekteissa, joissa työtä tekisivät tässä tutkielmassa esitettyjen opiskelijaprojektien tavoin menetelmää ennen tuntemattomat tahot. Mielenkiintoista olisi myös kartoittaa sitä, miten hyödyllisiksi asiakkaat eli sovellusalueiden käyttäjät RASKE-menetelmän kokevat.

LÄHDELUETTELO

- Avison D.E & Fitzgerald G. 2003. Where now for development methodologies? Communications of the ACM 46(1), 78-82.
- Bray T., Paoli J., Sperberg-McQueen C.M & Maler E. 2000. Extensible markup language (XML) 1.0 (Third Edition) [online], W3C Recommendation [Viitattu 11.02.2005]. Saatavilla [www-muodossa <http://www.w3.org/TR/2004/REC-xml-20040204/>](http://www.w3.org/TR/2004/REC-xml-20040204/)
- Ellis C.A. 1979. Information control nets: a mathematical model of office information flow. Teoksessa Proceedings of the conference on simulation, measurement and modeling of computer systems, ACM SIGMETRICS Performance Evaluation Review, vol 8(3), 225-238.
- Erikkson H-E & Penker M. 2000. UML. Gummerus Kirjapaino Oy, Jyväskylä.
- Goldfarb C.F. 1990. The SGML handbook. Oxford, UK. Oxford University Press.
- Hirsijärvi S. & Hurme H. 2000. Tutkimushaastattelu. Teemahaastattelun teoria ja käytäntö. Yliopistopaino. Helsinki.
- Honkaranta A. 2003. From genres to content analysis. Experiences from four case organizations. Jyväskylän yliopisto, väitöskirja.

- Honkaranta A. & Tyrväinen P. 2005. Content management in organizations. Teoksessa Encyclopedia of Information Science and Technology. M. Khosrowpour (toim.) Hershey, U.S.A., Idea Group Publishing, Inc, 550-555.
- ISO-8613. 1989. Information processing, Text and office systems, Office document architecture (ODA) and Interchange format.
- Järvinen P. & Järvinen A. 2000. Tutkimustyön metodeista. Opinpajan kirja. Tampere.
- Jyväskylän yliopisto. 2004. Informaatioteknologian tiedekunnan opinto-opas 2004-2005.
- Kalermo J. & Rissanen J. 2002. Agile software development in theory and practice. Jyväskylän yliopisto, Tietojärjestelmätieteen pro gradu – tutkielma.
- Lehtinen A. 1998. Tietokoneavusteinen mallinnus rakenteisten asiakirjastandardien kehittämisessä. Jyväskylän yliopisto, Tietojärjestelmätieteen pro gradu –tutkielma.
- Lyytikäinen V., Tiitinen P. & Salminen A. 2000. Graphical information models as interfaces for web document repositories. Teoksessa V. Di Gesù, S. Levialdi, & L. Tarantino (toim.) Proceedings of the working conference of Advanced visual interfaces, Palermo, Italy, May 2000, ACM Press New York, NY, USA, 261-265.

- Lyytikäinen V. 2003. Analysing requirements for content management. Teoksessa O. Camp, J. Filipe, S. Hammoudi & M. Piattini (toim.) Proceedings of the 5th International Conference on Enterprise Information Systems, Angers, France, Huhtikuu 23-26.
- Lyytikäinen V. 2004. Contextual and structural metadata in enterprise document management. Jyväskylän yliopisto, väitöskirja.
- Maler E. & El Andaloussi J. 1996. Developing SGML DTDs. From text to markup. Prentice Hall, New Jersey.
- Nance R.E. & Arthur J.D. 1988. The methodology roles in the realization of a model development environment. (toim.) Proceedings of the 20th conference on Winter simulation, San Diego, California, USA, December 1988, ACM Press New York, NY, USA, 220-225.
- Päivärinta T. 2001. A genre-based approach to developing electronic document management in the organization. Jyväskylän yliopisto, väitöskirja.
- RASKE 2005. RASKE2-projektin kotisivut [online]. Jyväskylä: Jyväskylän yliopisto [viitattu 28.01.2005]. Saatavilla [www-muodossa](http://www.muodossa) <<http://www.it.jyu.fi/raske>>
- Salminen A. 1996. Elektronisten dokumenttien hallinta. *Systeemityö-lehti* (4/96), 18-19. Saatavilla PDF-muodossa <<http://www.pcuf.fi/sytyke/lehti/kirj/st19964/964salm.PDF>>

- Salminen A., Tiitinen P. & Lyytikäinen V. 1999. Usability evaluation of a structured document archive. Teoksessa R.H. Sprague (toim.) Proceedings of the 32nd Annual Hawaii International Conference on System Science, Maui, Hawaii, USA, January 5-8, Los Alamitos, California, USA IEEE Computer Society, 644-655. Saatavilla [www-muodossa <http://csdl.computer.org/comp/proceedings/hicss/1999/0001/02/00012020.PDF>](http://csdl.computer.org/comp/proceedings/hicss/1999/0001/02/00012020.PDF)
- Salminen A., Lyytikäinen V. & Tiitinen P. 2000. Putting documents into their work context in document analysis. *Information Processing and Management*, 36(4), 623-641.
- Salminen A., Lyytikäinen V., Tiitinen P. & Mustajärvi O. 2000. SGML for E-Governance: The case of the Finnish Parliament. Teoksessa A.M. Tjoa, R.R. Wagner & A. Al-Zobaidie (toim.) Proceedings of the 11th International Workshop on Database and Expert Systems Applications, 349 – 353.
- Salminen A., Lyytikäinen V., Tiitinen P. & Mustajärvi O. 2001. Experiences of SGML standardization: The Case of the Finnish Legislative Documents. Teoksessa R.H. Sprague, Jr. (toim.), Proceedings of the 34 the Hawaii International Conference on System Sciences. Los Alamitos CA: IEEE Computer Society.
- Salminen A. 2003. Document analysis methods. Teoksessa Bernie C.L. (toim.) *Encyclopedia of Library and Information Science*, Second Edition, Revised and Expanded. New York: Marcel Dekker, 916-927.

- Salminen A. 2005. Building digital government by XML. Teoksessa R.H. Sprague, Jr. (toim.) Proceedings of the Thirty-Eighth Annual Hawaii International Conference on System Sciences. Los Alamitos, CA: IEEE Computer Society.
- Shlaer S. & Mellor S.J. 1992. Object lifecycles: modeling the world in states. Englewood Cliffs (NJ), Yourdon Press.
- Sutton M.J.D. 1996. Document management for the enterprise: principles, techniques, and applications. New York: Wiley.
- Tag2IT. 2004a. Projektisuunnitelma (versio 1.1). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- Tag2IT. 2004b. Kartoitusvaiheen raportti (versio 1.2). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- Tag2IT. 2004c. Kartoitustyön raportti (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- Tag2IT. 2004d. Viikkoraportti/Viikko 50. Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- Tag2IT. 2004e. Työpajan muistio. Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.

- Tag2IT. 2005a. Toteutusvaiheen raportti (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- Tag2IT. 2005b. Testausvaiheen raportti (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- Tag2IT. 2005c. Päätösvaiheen raportti (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- Tag2IT. 2005d. Tag2IT-projektin raportti. Jyväskylän yliopiston tiedekunnan toimiston kokousasiakirjojen rakenteistaminen (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- Tag2IT. 2005e. Loppuraportti (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- Tag2IT. 2005f. Ryhmäpalaverin muistio vko 9. Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- Tag2IT. 2005g. Katselmoinnin pöytäkirja (26.1.2005). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- Tag2IT. 2005h. Katselmoinnin pöytäkirja (8.2.2005). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.

- Tietojenkäsittelytieteiden laitos. 2004a. Standardikuvauslomake (MemoX), Projektin johtaminen 2004 - 2005. Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- Tietojenkäsittelytieteiden laitos. 2004b. Standardikuvauslomake (Xoo), Projektin johtaminen 2004 - 2005. Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- Tiitinen P., Päivärinta T., Salminen A. & Lyytikäinen V. 1997. Suomalaisten EU-laindäädäntöasiakirjojen rakenteistaminen. RASKE-projektin raportti [online]. [Viitattu 28.01.2005] Saatavilla PDF-muodossa <<http://www.cs.jyu.fi/~airi/raportit/RASKE-EU-raportti.PDF>>
- Tiitinen, P., Salminen, A. & Lyytikäinen, V. 1997. EU-lainsäädäntöasiakirjat Suomessa. RASKE-projektin raportti. Eduskunnan kanslian julkaisu 1/1997. Saatavilla PDF-muodossa: <<http://www.cs.jyu.fi/~airi/raportit/RASKE-EU-asiakirjat.PDF>>
- Tolvanen J-P. 1998. Incremental method engineering with modelling tools: theoretical principles and empirical evidence. Jyväskylän yliopisto, väitöskirja.
- Vartiainen T. 2005. Moral conflicts in a project course in information systems education. Jyväskylän yliopisto, väitöskirja.
- XooZoo. 2004a. Käynnistysvaiheen raportti (versio 1.1) Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.

XooZoo. 2004b. Viikkoraportti 45 (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.

XooZoo. 2004c. Viikkoraportti 46 (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.

XooZoo. 2004d. Viikkoraportti 48 (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.

XooZoo. 2004e. Workshop pöytäkirja (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.

XooZoo. 2005a. Projektisuunnitelma (versio 5.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.

XooZoo. 2005b. Laatimisprosessin kartoitus. Vaiheraportti (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.

XooZoo. 2005c. Loogisenrakenteen kartoitus. Vaiheraportti (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.

XooZoo. 2005d. Sisällön ja käytön kartoitus. Vaiheraportti (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.

- XooZoo. 2005e. Rakenteen määrittäminen ja mallidokumenttien laatiminen XML-kielillä. Vaiheraportti (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- XooZoo. 2005f. Päätämismuutokset. Vaiheraportti (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- XooZoo. 2005g. Loppuraportti (versio 1.1). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- XooZoo. 2005h. Opinto-oppaan laatimisprosessin kuvaus & sisältökomponentit. Raportti (versio 3.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- XooZoo. 2005i. Jatkokehitysehdotuksia (versio 1.0) Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- XooZoo. 2005j. Ryhmäpalaveri – Viikko 10. Muistio (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- XooZoo. 2005k. Jatkokehitys-Workshop muistio (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- Yin R.K. 1989. A case study research – design and methods. Sage Publications, Newbury Park.

LIITE 1. Haastattelukysymykset

1. Tiedot haastateltavasta

1.1 Pääaine sekä taso, jolla olet tällä hetkellä

Pääaine:

- Aineopinnot lähes suoritettu; syventäviä opintoja suoritettu hieman.
- Lähes kaikki syventävät opinnot suoritettu.

1.2 Aikaisempi kokemus tietojenkäsittelytieteisiin liittyvästä projektityöskentelystä

- Ei lainkaan
- Kokemusta 1-2 projektista
- Kokemusta useammasta projektista

1.3 Oliko jokin seuraavista sinulle tuttu ennen projektityön alkua?

- UML (Unified Modeling Language) -mallinnuskieli
- OOA (Object-Oriented Analysis) -menetelmä
- Maler & El Andaloussin menetelmä
- ICN (Information Control Nets) -malli

2. RASKE-menetelmän omaksuminen

2.1 Olitko käyttänyt RASKE-menetelmää ennen projektiasi?

- Kyllä
- Ei

2.2 Olitko kuullut / lukenut RASKE-menetelmästä ennen nykyistä projektiasi?

- Kyllä
- Ei

2.3 Oliko RASKE-menetelmän oppiminen vaikeaa?

- Ei ollut
- Hieman
- Kyllä; menetelmän oppiminen vaati minulta ponnisteluita

2.4 Voiko RASKE-menetelmän mielestäsi omaksua pelkästään lukemalla menetelmästä kirjoitettuja julkaisuja?

- Kyllä
- Ei

3. RASKE-menetelmän mallit

3.1 Mitä työvälineitä käytitte laatiessanne RASKE-menetelmän malleja?

3.2 Pitäisikö RASKE-menetelmällä olla jokin tietty mallintamisväline?

3.3 Oliko kohdealueen yleiskuvauksen mallintaminen kontekstikaavioon helppoa? Onko malli selkeä?

3.4 Oliko roolien mallintaminen helppoa? Onko taulukko/malli selkeä?

3.5 Oliko Syöte/tuotos-malli helposti sovellettavissa? Onko malli selkeä?

- 3.6 Oliko tilasiirtymäkaavio helposti sovellettavissa? Onko malli selkeä?
- 3.7 Oliko dokumenttikomponenttien mallintaminen helppoa?
- 3.8 Oliko laatimastanne komponenttikaavioista apua DTD-tiedostoja laadittaessa?
- Kyllä. Saimme alustavat elementit suoraan kaavioista.
 - Jonkin verran. Komponenttikaaviot antoivat hyvin ideoita elementtien laatimiseen.
 - Hieman. Tarkastelimme komponenttikaaviota harvoin.
 - Ei ollut, koska emme tarkastelleet komponenttikaavioita enää niiden laatimisen jälkeen.
 - En osallistunut DTD:n laatimiseen, joten en osaa sanoa.
- 3.9 Mikä malleista oli helpoin hahmottaa?
- 3.10 Mikä malleista oli vaikeinta hahmottaa?
- 3.11 Oliko jokin malleista erityisen käyttökelpoinen projektissasi?
- 3.12 Oliko RASKE-menetelmässä jokin malli, jonka soveltaminen omassa projektissasi oli vaikeaa?
- 3.13 Oliko menetelmässä malli, jota jouduitte muuttamaan, jotta se olisi sopinut projektiinne?
- 3.14 Oliko jokin malli projektissanne tarpeeton?
- 3.15 Oliko RASKE-menetelmässä malli, jonka tarkoitus/hyödyllisyys ei selvinnyt sinulle?

4. RASKE- menetelmä projektityöskentelyssä

4.1 Sopiko RASKE-menetelmän käyttö omaan projektiisi?

- Hyvin, koska...
- Tyydyttävästi, koska ...
- Huonosti, koska...

4.2 Puuttuiko RASKE-menetelmästä jokin kohde, jota olisi mielestäsi pitänyt mallintaa?

4.3 Mitä mieltä asiakkaasi oli RASKE-menetelmällä tehdyistä tuloksista?

4.4 Millaiseen projektiin RASKE-menetelmä mielestäsi sopisi?

- Pienelle projektille
- Isolle projektille
- En usko, että projektin laajuudella on merkitystä menetelmän sopivuuteen

4.5 Jos aloittaisit alusta asiakirja-analyysin, mitä tekisit toisin? (Esim. ajankäyttö, tiedonkeruu, asiakaskontaktit, tms.)

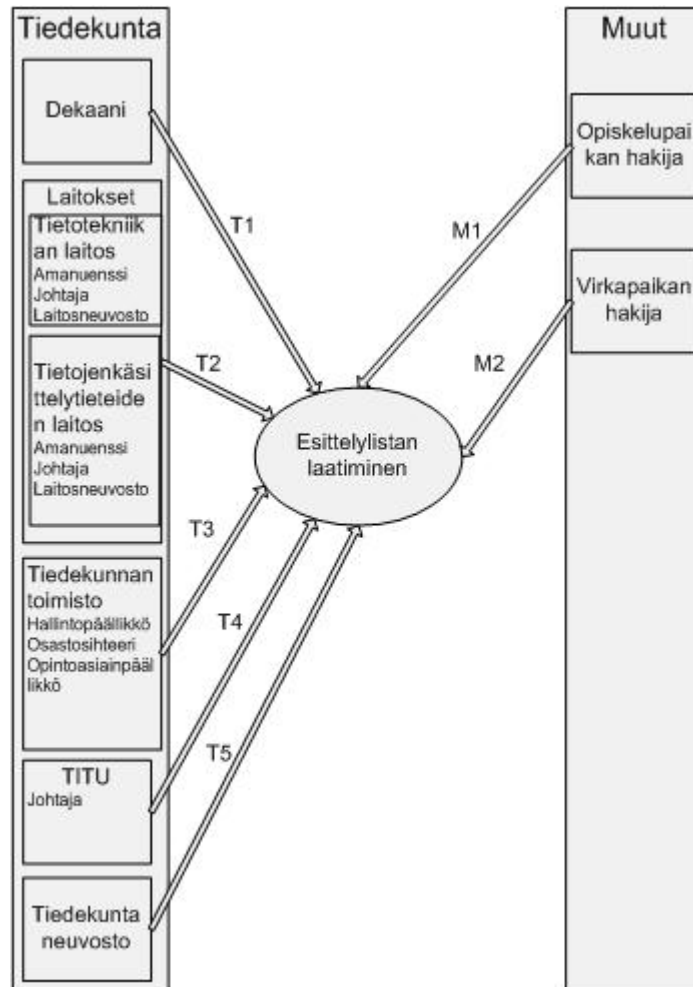
4.6 Kuinka käsityksesi rakenteisiin dokumentteihin tai dokumenttien hallintaan muuttui projektin aikana?

4.7 Muuttuiko oma suhtautumisesi rakenteisiin dokumentteihin tai dokumenttien hallintaan projektin aikana?

- Ei muuttunut
- Muuttui; kiinnostuin rakenteisista dokumenteista ja rakenteistamisesta
- Muuttui; aikaisempi kiinnostus aiheeseen vähenyi jonkin verran.

- 4.8 Lähtisitkö vastaisuudessa mukaan samankaltaisiin projekteihin? Jos kyllä, niin käyttäisitkö RASKE-menetelmää?
- 4.9 Suositteletko RASKE-menetelmän käyttöä muille? Miksi?

LIITE 2. Tag2IT-projektin kontekstikaavio



T1: Dekaanin hyväksyy esittelylistan allekirjoittamalla sen. Dekaanin tekee tarvittaessa esityksiä kokoukseen.

T2: Laitos / laitosneuvosto tekee aloitteen professorinviran täyttämistä sekä hoitaa myös muut laitosta koskevat nimitykset lähettämällä asiaan kuuluvan lomakkeen tiedekunnan toimistoon. Laitos toimittaa opiskelijan tutkintohakulomakkeen toimistoon, kun opiskelijan tutkintoon liittyvät opinnot on suoritettu. Laitos / laitosneuvosto tekee esityksen opetushenkilökunnan työsuunnitelmista (esittelylistan liitteitä). Laitos ilmoittaa asiaankuuluvalla lomakkeella myös tulevista lisenssiaati- ja väitöstutkimuksista, joiden arvostelu käsitellään kokouksessa. Laitos toimittaa oman laitosneuvostossa hyväksytyt talous- ja toimintasuunnitelmansa toimistoon.

T3: TITU lähettää nimityksistä tiedon asiaankuuluvan lomakkeen kautta tiedekunnan toimistoon. TITU lähettää oman toiminta- ja taloussuunnitelmansa.

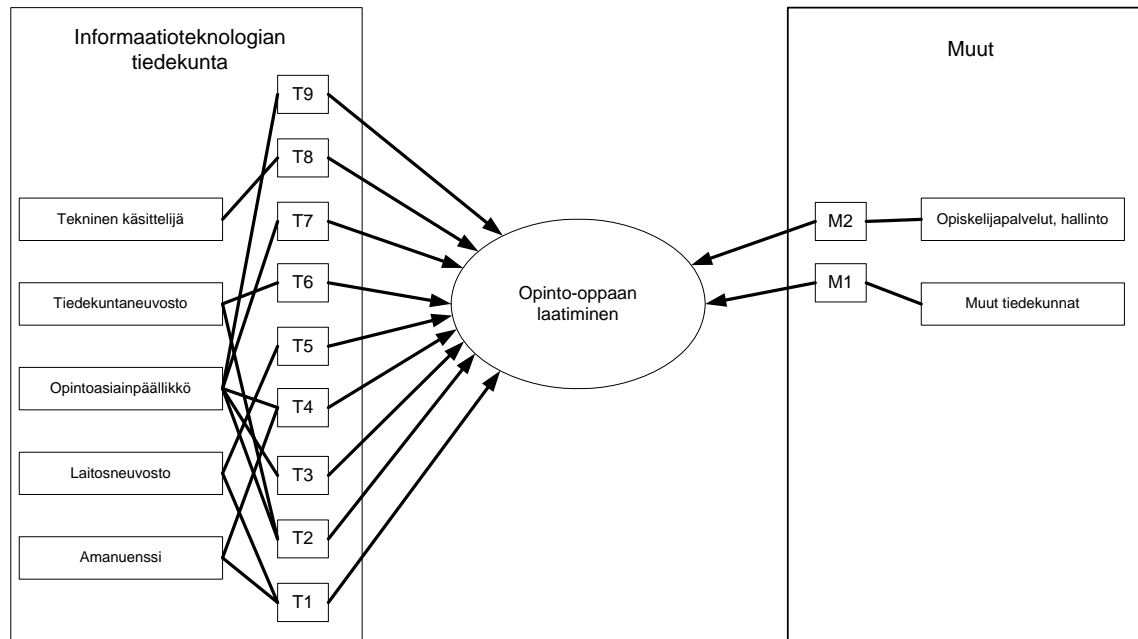
T4: Hallintopäällikkö ja opintoasiainpäällikkö laatii esitykset ja liitteet esittelylistaan. Hallintopäällikkö ja opintoasiainpäällikkö päättää, mitä asioita kokouksessa tullaan käsittelemään ja tekee sen pohjalta asialistan.

T5: Tiedekuntaneuvosto tekee hallintopäällikön esityksen perusteella hallitukselle esityksen, jos asia täytyy päättää hallituksessa.

M1: Opiskelupaikan hakija lähettää kirjallisen valituksen tiedekuntaan. Valituksen tarkastaa esim. tenttikysymyksen laatinut opettaja, joka antaa valituksesta lausunnon. Lausunto käsitellään kokouksessa.

M2: Viran hakija lähettää kirjallisen hakemuksen tiedekuntaan. Hakemusasiakirjojen perusteella laaditaan nimittämismuistio (liitteenä), jossa perustellaan hallintopäällikön tiedekuntaneuvostolle tekemä nimitysesitys. (Tag2IT, 2004c, 4-5.)

LIITE 3. XooZoo-projektin kontekstikaavio



T1: Laitoksen amanuenssi saa tutkintorakenneryhmän esityksen tutkintorakenteesta ja –vaatimuksista. Amanuenssi valmistelee esityksen laitosneuvostolla. Tämän jälkeen laitosneuvosto hyväksyy tutkintorakenteen ja tekee laitoksen esityksen tutkintorakenteesta. Uutta tutkintorakennetta ei esitellä välttämättä joka vuosi.

T2: Opintoasiainpäällikkö koostaa laitosten tutkintorakenteet ja tekee niistä esitykset tiedekuntaneuvostolle. Tiedekuntaneuvosto päättää opintoasiainpäällikön esityksen pohjalta tutkintorakenteet ja –vaatimukset.

T3: Yhteisten opintojen vastuuhenkilö (tämän hetken prosessissa opintoasiainpäällikkö, tulevaisuudessa voisi olla oma yhteisten opintojen vastuuhenkilö) koordinoi tiedekunnan yhteiset opinnot. Lisäksi opintoasiainpäällikkö tai mahdollinen opinto-opasta varten muodostettu opasryhmä tekee ohjeistukset muutoksille uudessa opinto-oppaassa.

T4: Opintoasiainpäällikkö ja laitosten amanuenssit valmistelevat opetusohjelmaa sekä laativat opinto-oppaan sisältöä. Muilta tiedekunnilta (toimija kohdas-

sa M1) saadaan niiden materiaalia opinto-opasta varten, esimerkiksi sivuaineen esittely.

T5: Laitosneuvosto hyväksyy oman laitoksen opetusohjelman ja opinto-oppaan tekstit.

T6: Tiedekuntaneuvosto hyväksyy opintoasiainpäällikön koostaman ehdotuksen opetusohjelmaksi. Tiedekuntaneuvosto päättää tarvittavista korjauksista. Korjauksien koordinointiin valtuutetaan henkilö (esim. opintoasiainpäällikkö).

T7: Opintoasiainpäällikkö yhdessä opiskelijapalveluiden (toimija kohdassa M2) kanssa valmistelee julkaisun (painotyön tarjouspyyntö, layoutin julkaisu).

T8: Tekninen käsittelijä muuntaa tiedoston PDF-muotoon sekä laatii www-julkaisun.

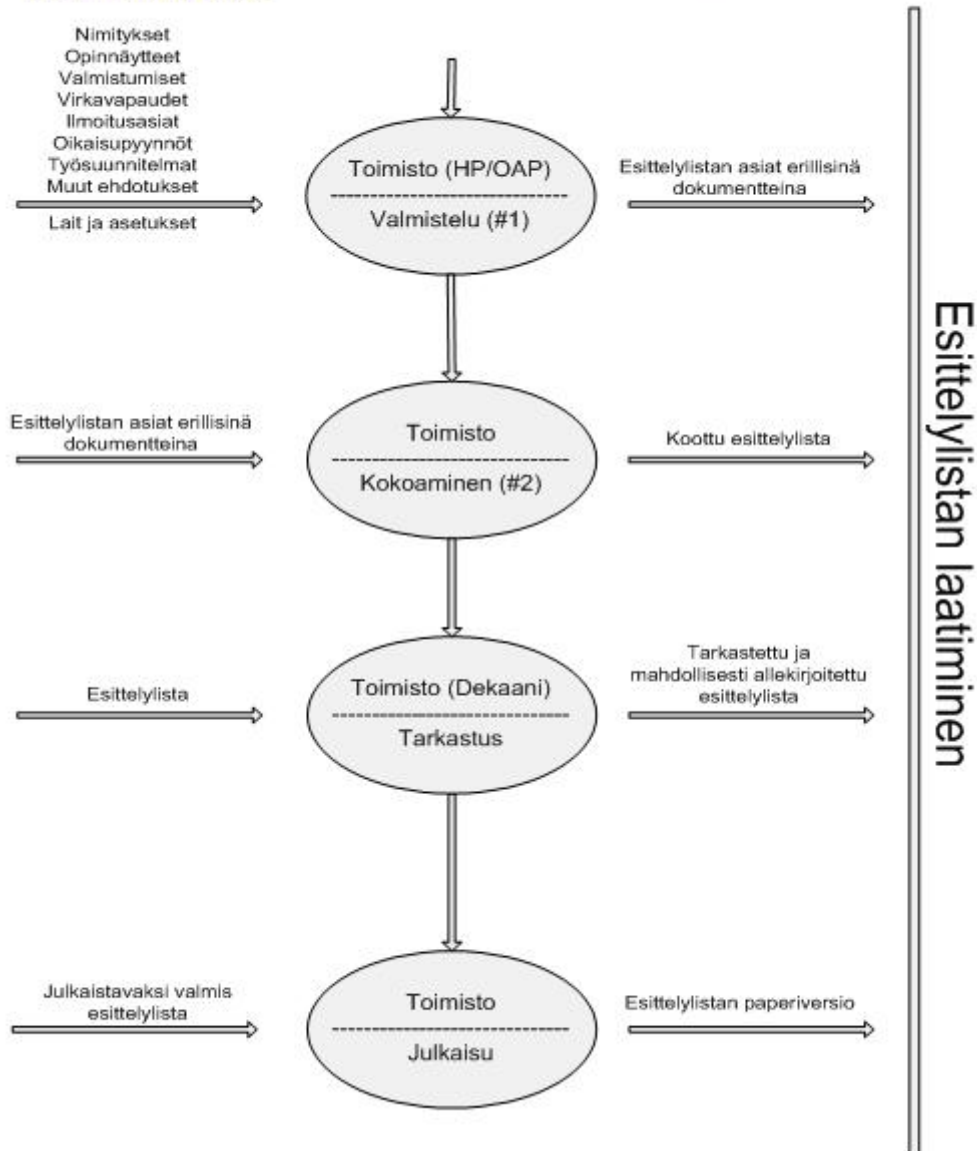
T9: Opintoasiainpäällikkö päättää yksiköiden jakelumäärät, suorittaa jakelun eri jakelijoille sekä hankkii suorajakelutiedot. Lisäksi toimitetaan mallioppaat ja tarvittavat korjausehdotukset painolle.

M1: Ks. kohta T4.

M2: Opiskelijapalvelut ja yliopiston hallinto päättävät painosmäärät, ns. "leikat" (layout) ja laativat tarjouspyynnöt painotyölle. (XooZoo 2005h, 5-6)

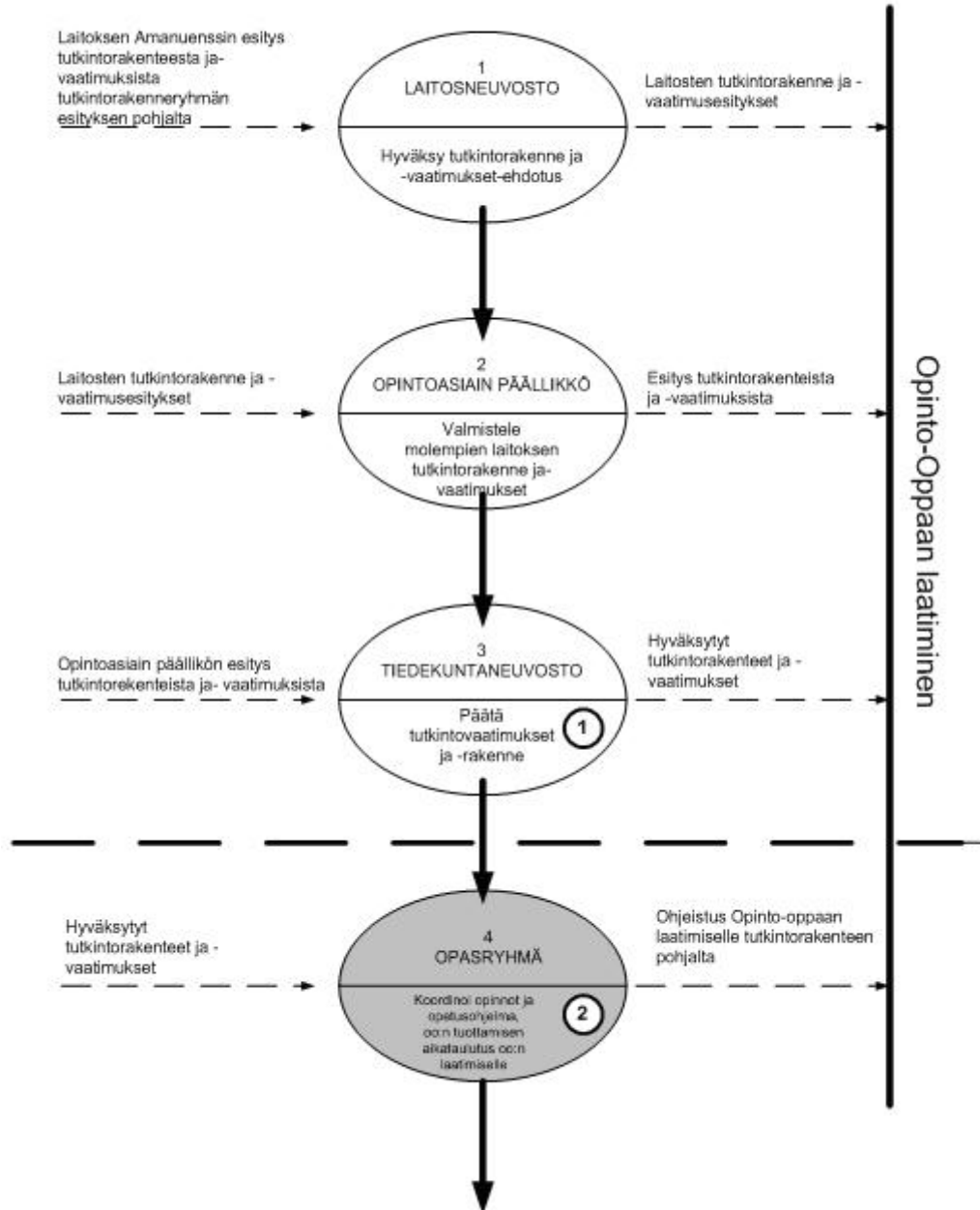
LIITE 4. Tag2IT-projektin syöte/tuotos-malli

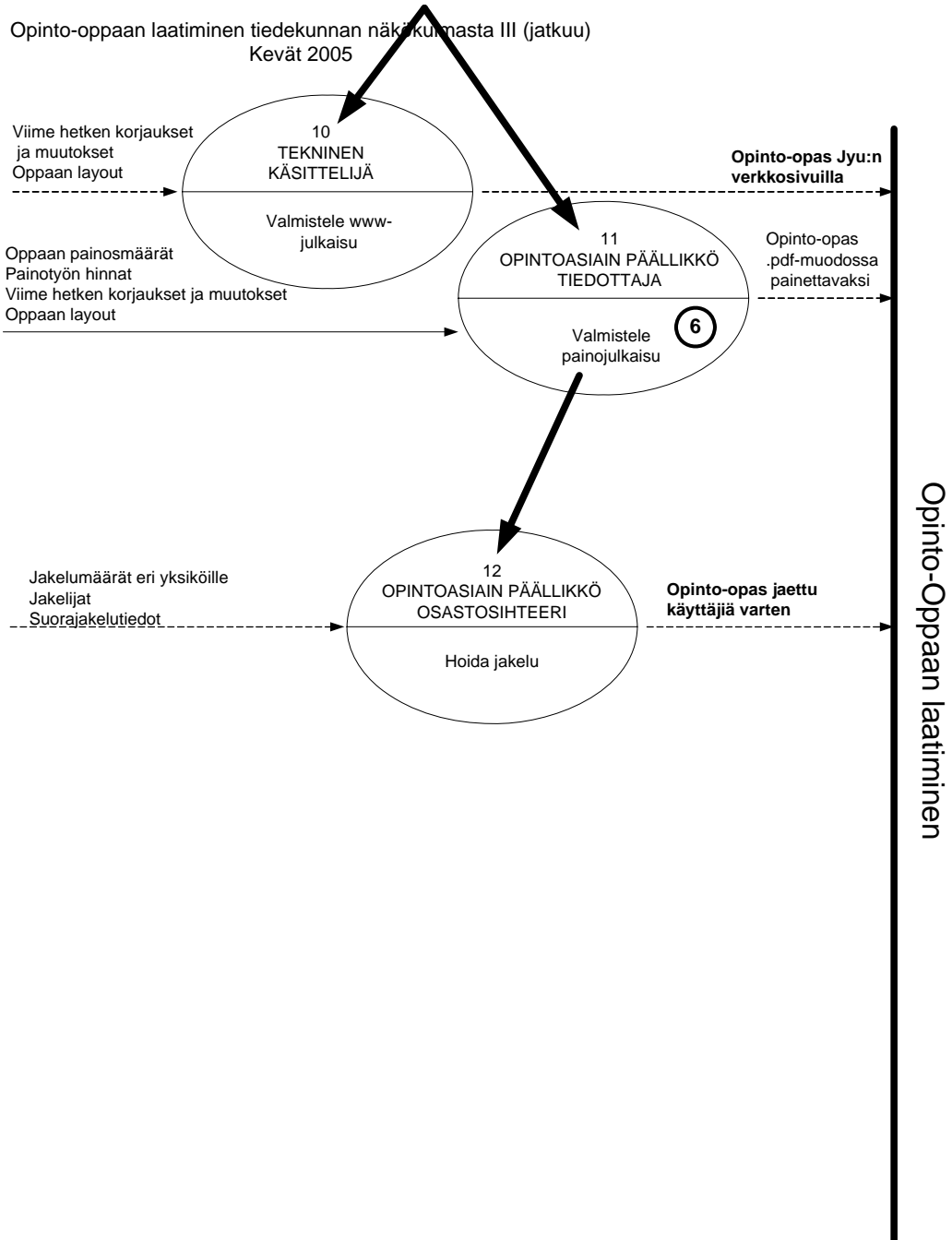
Toiminta on esitetty ympyrällä. Toiminnon tekijä on ympyrässä ylinä. Toimintaan tarvittavat komponentit kerrotaan ympyrään osoittavalla, vasemman puoleisella nuolella. Toiminnan tulos on ympyrän oikeanpuoleinen nuoli. Sulussa oleva # numero on viite raportin loppupuolella olevaan kehittämistarvetaulukkoon.



LIITE 5. XooZoo-projektin syöte/tuotos-malli

Opinto-oppaan laatiminen tiedekunnan näkökulmasta
Kevät 2005

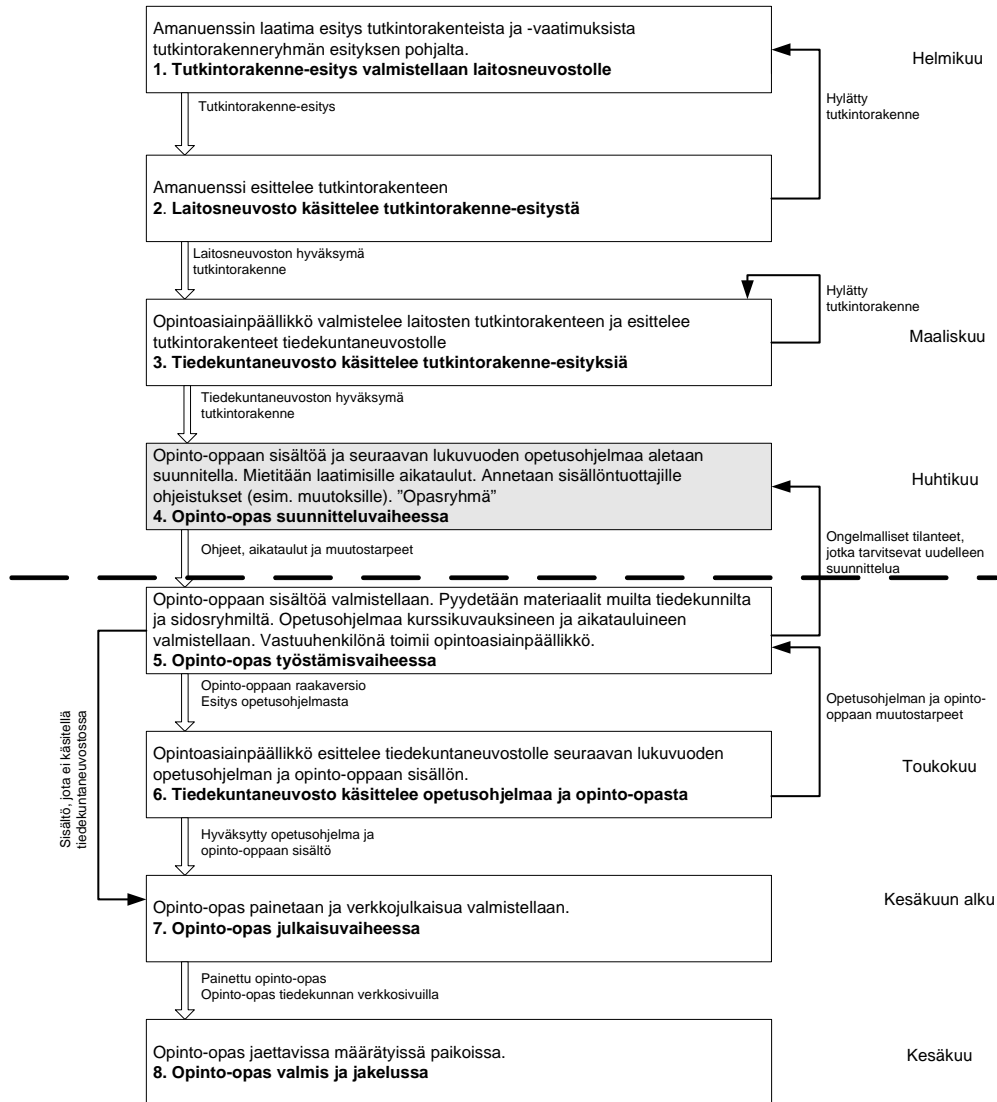




LIITE 6. Tag2IT-projektin tilasiirtymäkaavio



LIITE 7. XooZoo-projektin tilasiirtymäkaavio



LIITE 8. Tag2IT-projektiaiheen kuvauslomake

1 YIRITYKSEN TIEDOT	Yrityksen nimi: Jyväskylän yliopisto Informaatioteknologian tiedekunta	Yrityksen toimiala: Julkinen hallinto Tutkimus ja koulutus
	Postitoimiosoite: Jyväskylän yliopisto. Informaatioteknologian tiedekunta. PL 35 (AG 2.krs), 40014 Jyväskylän yliopisto, Suomi	
	Muut projektin kannalta merkittävät toimipaikat: Agora Center, Virtuaaliyliopisto, Tietojenkäsittelytieteiden laitos, Tietotekniikan laitos.	
	Yrityksen henkilöstömäärä: 7 (toimisto), n. 1900 opiskelijaa, 200 jatko-opiskelijaa.	Projektiin osallistuvien henkilöiden määrä: Tiedekunnan toimisto: 4 TKTL+TTL: 2
2 PROJEKTIAIHEEN ALUSTAVAT TIEDOT	Projektityön nimi: MemoX (XML ja Muistidokumentaation kehittäminen).	
	Lyhyt kuvaus projektista: IT-tiedekunnassa tuotetaan vuosittain suuri määrä kokousten esityslistoja ja pöytäkirjoja. Tällä hetkellä näiden laatiminen tapahtuu useita eri tiedostoja yhdistelemällä. Laatu- ja virhealtuus on työläs ja virhealtuus. Lisäksi dokumenteissa on samaa tietosisältöä, jota tällä hetkellä ei kyetä käyttämään hyväksi suoraan eri dokumentteja tuottaessa. Toistaiseksi tietosisällön tuottaminen on virhealtuus ja lisää työmäärää. Esityslistojen ja pöytäkirjojen jakelua elektronisessa muodossa voisi tehostaa esim. Intranetiä hyödyntämällä, mutta eri muotoisten tallenteiden (.doc, HTML) laatiminen lisää työtä entisestään. MemoX-projektin tehtävänä on kehittää tiedekunnan esityslistojen ja pöytäkirjojen laatimis- ja jakeluprosessia ja siinä käytettäviä työvälineitä mainittujen ongelmien ratkaisemiseksi.	

Luonnehdinta aiheesta (Merkitse rasti siihen kohtaan, joka kuvaa parhaiten projektiaihetta):

|-----X-----|
 Selvitys- ja Ohjelmointi
 Tutkimustyö

Työn sisällön arviointi (Merkitse rasti ruutuun):

Tehtävä/tilanne	Ei (vielä) tiedossa	Ei lainkaan	Hieman	Jonkin verran	Paljon	Erittäin paljon
Määrittely					X	
Suunnittelu					X	
Toteutus (ml. ohjelmointi)			X			
Koulutus			X			
Tutkimus				X		
Jatkokehityksen suunnittelu			X			
Muu						

Kuinka pitkälle projektiaihetta on työstetty tähän mennessä?:

Mahdollisesti käytettävät menetelmät, työvälineet, ja teknologiat on kartoitettu. Näistä osa on kiinnitetty (XML), osa voidaan vaihtaa. Projektiryhmälle voidaan tarjota tukea ja perehdytystä käytettäviin teknologioihin (XML ja liitännäiskielet) sekä organisaation toiminnan kehittämisessä mahdollisesti käytettävien menetelmien käyttöön. Projektiryhmällä on mahdollisuus saada tukea myös Korppi-liitântöjen toteuttamiseen sekä paperijulkaisuprosessiin. Kehittämisessä tarvittava osa tiedekunnan prosessien ja roolien kartoitusta on havaittu tarpeelliseksi myös muun kehittämistyön kannalta.

Miten projektiaihe liittyy organisaatiossanne tällä hetkellä käytössä oleviin järjestelmiin tai käynnissä oleviin projekteihin?:

Projektiryhmän työllä on liittymäkohtia Korppi-järjestelmän ja tiedekunnan Intranetin kehittämistyöhön. IT-tiedekunta on käynnistämässä myös toista kehittämisprojektia (XOO), jonka kanssa projektilaiset voivat tehdä yhteistyötä muun muassa nykyisten ja tulevien toimintatapojen selvitys- ja suunnittelutyössä.

	<p>Aiheen työstämiseen käytettävät mahdolliset toteutusvälineet ja menetelmät:</p> <p>Menetelmät: Maler & El Andaloussi-mentelmä RASKE-menetelmä (OOA, OMT+)</p> <p>Toteutusvälineet: XMLSpy Professional, Authentic, MS Word 2003 (MS InfoPath, Stylevision, MS Access)</p>
3 TOIVOMUKSET RYHMÄN SUHTEEN	<p>Opiskelijoilta toivottava tiedollinen ja taidollinen osaaminen sekä esitys ryhmän ihannekoostumuksesta esimerkiksi osaamisalueiden suhteen:</p> <p>Rakenteisten dokumenttien alueen, ja eritoten XML -kielen ja joidenkin liitännäis-kielten (DTD, XML Schema, XSLT/DOM), sekä näihin liittyvien menetelmien (ks. menetelmät) tuntemus tarjoaa hyvän pohjan projektityöskentelylle; osalla ryhmäläisistä olisi hyvä olla tämän alueen tuntemusta. Muita projektityössä tarvittavia taitoja tai kiinnostuksen alueita ovat: dokumenttien ja sisältöjen hallinta, organisaatioiden toiminnan kehittäminen, vaatimusmäärittely, mallintaminen ja suunnittelu, käyttäjäystävällisten toimisto- ja julkaisu-järjestelmien suunnittelu. Optimaalisessa ryhmässä voisi olla sekä teknisesti että ”käyttäjäläheisesti” orientoituneita henkilöitä.</p>
	<p>Projektin menestykselliseen hoitamiseen edellytettävä kielitaito:</p> <p>Suomen kieli riittää. Englannin kielen osaamisesta on etua tarvittavien teknisten määritysten ja ohjelma-oppaiden tulkinnessa.</p>
	<p>Projektiin osallistumisen estävät rajoitukset (Esimerkiksi opiskelijan työsuhte kilpailevaan yritykseen):</p>
4 MUUTA HUOMIOITAVAA	<p>Projektiryhmä saa apua ja tukea työskentelyynsä rakenteisten dokumenttien menetelmien ja työvälineiden osalta TKTL-laitokselta (Anne Honkaranta). Tiedekunnan Intranet-ratkaisujen kehittäminen tulee ottaa työssä huomioon. Tiedekunnan toimisto huolehtii perehdytyksestä aihepiiriin, ja on käytettävissä niin nykytilan ja ongelmakohtien kartoitukseen, kuin tulevien ratkaisujen ideointiinkin. Nykykäytännöissä ja työvälineissä on ongelmia, jotka aiheuttavat monia ongelmia tiedekunnan toimiston työhön. Projektin tulos edistää tiedekunnan toimintaa, ja hyödyttää siten sekä tiedekunnan hallintoa että kaikkia tiedekunnassa opiskelevia.</p>

LIITE 9. XooZoo-projektiaiheen kuvauslomake

1 YIRITYKSEN TIEDOT	Yrityksen nimi: Jyväskylän yliopisto Informaatioteknologian tiedekunta	Yrityksen toimiala: Julkinen hallinto Tutkimus ja koulutus
	Postitoimiosoite: Jyväskylän yliopisto. Informaatioteknologian tiedekunta. PL 35 (AG 2.krs), 40014 Jyväskylän yliopisto, Suomi	
	Muut projektin kannalta merkittävät toimipaikat: Agora Center, Virtuaaliyliopisto, Tietojenkäsittelytieteiden laitos, Tietotekniikan laitos.	
	Yrityksen henkilöstömäärä: 7 (toimisto), n. 1900 opiskelijaa, 200 jatko-opiskelijaa.	Projektiin osallistuvien henkilöiden määrä: Tiedekunnan toimisto: 4 TKTL+TTL: 2
2 PROJEKTIAIHEEN ALUSTAVAT TIEDOT	Projektityön nimi: XOO (XML ja Opinto-Oppaan hallinnan kehittäminen).	
	Lyhyt kuvaus projektista: <p>IT-tiedekunnassa tuotetaan vuosittain Opinto-opas. Oppaan laatimisprosessi on monimutkainen. Oppaaseen tietoa tulee useista eri lähteistä: Kurssikuvaukset Korpista, esittelytekstejä tiedekunnasta, sekä tietoa opinnoista ja opintorakenteesta kahdelta laitokselta, ja laitoksen useilta henkilöiltä. Opas tuotetaan sekä WWW-muotoon (XHTML-kieli), että paperimuotoon (Opinto-opaskirja). XOO-projektin tavoitteena on kehittää Opinto-oppaan laatimista ja julkaisemista. Laadinnassa hyödynnettäisiin jatkossa XML-kieltä ja sen liitännäiskieliä. Opinto-oppaan laatimista ja koostamista tulisi automatisoida mahdollisimman pitkälle. XOO-projektissa kehitettäisiin sekä oppaan laatimisprosessia, että oppaan laatimisessa käytettäviä työvälineitä.</p>	

Luonnehdinta aiheesta (Merkitse rasti siihen kohtaan, joka kuvaa parhaiten projektiaihetta):

|-----X-----|
 Selvitys- ja Ohjelmointi
 Tutkimustyö

Työn sisällön arviointi (Merkitse rasti ruutuun):

Tehtävä/tilanne	Ei (vielä) tiedossa	Ei lainkaan	Hieman	Jonkin verran	Paljon	Erittäin paljon
Määrittely					X	
Suunnittelu					X	
Toteutus (ml. ohjelmointi)				X		
Koulutus			X			
Tutkimus				X		
Jatkokehityksen suunnittelu			X			
Muu						

Kuinka pitkälle projektiaihetta on työstetty tähän mennessä?:

Mahdollisesti käytettävät menetelmät, työvälineet, ja teknologiat on kartoitettu. Näistä osa on kiinnitetty (XML), osa voidaan vaihtaa. Projektiryhmälle voidaan tarjota tukea ja perehdytystä käytettäviin teknologioihin (XML ja liitännäiskielet) sekä organisaation toiminnan kehittämisessä mahdollisesti käytettävien menetelmien käyttöön. Projektiryhmällä on mahdollisuus saada tukea myös Korppi-liitântöjen toteuttamiseen sekä paperijulkaisuprosessiin. Kehittämisessä tarvittava osa tiedekunnan prosessien ja roolien kartoitusta on havaittu tarpeelliseksi myös muun kehittämistyön kannalta.

Miten projektiaihe liittyy organisaatiossanne tällä hetkellä käytössä oleviin järjestelmiin tai käynnissä oleviin projekteihin?:

Projektiryhmän työllä on liittymäkohtia sekä Korppi-järjestelmän kehittämistyöhön, että Suomen virtuaaliyliopiston eOpinto-opas-hankkeeseen. IT-tiedekunta on käynnistämässä myös toista kehittämisprojektia (MemoX), jonka kanssa projektilaiset voivat tehdä yhteistyötä muun muassa nykyisten ja tulevien toimintatapojen selvitys- ja suunnittelutyössä.

	<p>Aiheen työstämiseen käytettävät mahdolliset toteutusvälineet ja menetelmät:</p> <p>Menetelmät: Maler & El Andaloussi-mentelmä RASKE-menetelmä (OOA, OMT+)</p> <p>Toteutusvälineet: XMLSpy Professional, Authentic, MS Word 2003 (MS InfoPath, Stylevision, MS Access)</p>
3 TOIVOMUKSET RYHMÄN SUHTEEN	<p>Opiskelijoilta toivottava tiedollinen ja taidollinen osaaminen sekä esitys ryhmän ihannekoostumuksesta esimerkiksi osaamisalueiden suhteen:</p> <p>Rakenteisten dokumenttien alueen, ja eritoten XML -kielen ja joidenkin liitännäis-kielten (DTD, XML Schema, XSLT/DOM), sekä näihin liittyvien menetelmien (ks. menetelmät) tuntemus tarjoaa hyvän pohjan projektityöskentelylle; osalla ryhmäläisistä olisi hyvä olla tämän alueen tuntemusta. Muita projektityössä tarvittavia taitoja tai kiinnostuksen alueita ovat: dokumenttien ja sisältöjen hallinta, organisaatioiden toiminnan kehittäminen, vaatimusmäärittely, mallintaminen ja suunnittelu, käyttäjäystävällisten toimisto- ja julkaisu-järjestelmien suunnittelu. Projektityön sisältö on monipuolinen; käyttäjähaastatteluista ja nykytila-analyysistä aina uusien ratkaisujen ohjelmointiin ja dokumenttipohjien suunnitteluun saakka. Optimaalisessa ryhmässä voisi olla sekä teknisesti että ”käyttäjäläheisesti” orientoituneita henkilöitä.</p> <p>Projektin menestykselliseen hoitamiseen edellytettävä kielitaito:</p> <p>Suomen kieli riittää. Englannin kielen osaamisesta on etua tarvittavien teknisten määritysten ja ohjelma-oppaiden tulkinnessa.</p> <p>Projektiin osallistumisen estävät rajoitukset (Esimerkiksi opiskelijan työsuhte kilpailevaan yritykseen):</p>
4 MUUTA HUOMIOITAVAA	<p>Projektiryhmä saa apua ja tukea työskentelyynsä sekä Korppi-henkilöstöltä (Vesa Lappalainen) että rakenteisten dokumenttien menetelmien ja työvälineiden osalta TKTL-laitokselta (Anne Honkaranta). Tiedekunnan toimisto huolehtii perehdytyksestä aihepiiriin, ja on käytettävissä niin nykytilan ja ongelmakohtien kartoitukseen, kuin tulevien ratkaisujen ideointiinkin. Projektin tulos edistää tiedekunnan toimintaa, ja hyödyttää siten sekä tiedekunnan hallintoa että kaikkia tiedekunnassa opiskelevia. Hanke on herättänyt mielenkiintoa myös eräissä toisessa tiedekunnassa. On mahdollista, että projektiryhmäläisille tarjoutuu mahdollisuus jatkaa kesä- tai gradutöiden merkeissä jommassa kummassa yksikössä.</p>