

Pekka Rinne

**XML-dokumenttien koostaminen**  
**Havaintoja opinto-opas -projektista**

Tietojärjestelmätieteen  
pro gradu -tutkielma  
19.12.2005

Jyväskylän yliopisto  
Tietojenkäsittelytieteiden laitos  
Jyväskylä

## TIIVISTELMÄ

Rinne, Pekka Juhani

XML-dokumenttien koostaminen

Havaintoja opinto-opas -projektista / Pekka Rinne

Jyväskylä: Jyväskylän yliopisto, 2005.

118 s.

Tietojärjestelmätieteen pro gradu -tutkielma

Tässä tutkielmassa tarkastellaan XML-dokumenttien koostamista ja tutustutaan aihepiirin käytännön havaintoihin tapaustutkimuksen avulla. Dokumenttien koostamisella tarkoitetaan uusien dokumenttien koostamista olemassa olevista tai uusista lähdedokumenteista tai niiden osista esimerkiksi erilaisia julkaisuja varten. Tutkielman teoriaosuus perustuu aikaisempaan tutkimukseen dokumenttien koostamisesta. Tutkielman laadullinen tapaustutkimus koskee Jyväskylän yliopiston informaatioteknologian tiedekunnan opinto-oppaan kehittämishanketta. Hankkeen aikana tiedekunnan lukuvuoden 2005-2006 opinto-opas koostettiin XML-muotoon muunnetuista lähdedokumenteista.

Tutkielman tavoitteena on verrata opinto-oppaan koostamismallia aiemmassa tutkimuksessa esiteltyyn dokumenttien koostamismalliin.

Tutkielmassa havaittiin aiemmassa tutkimuksessa esitetyn koostamismallin olevan pidemmälle automatisoidumpi ja täten käyttäjäystävällisempi kuin opinto-oppaan koostamismalli. Tulosten perusteella voidaan myös todeta, ettei koostamisessa käytettävien lähdedokumenttien raekoko ole absoluuttinen, vaan se tulisi suunnitella tapauskohtaisesti. Lisäksi tapaustutkimuksen aikana toteutetun koostamista helpottavan kahden tietojärjestelmän välisen integraation havaittiin sujuneen ongelmitta juuri XML:n käytön ansiosta.

AVAINSANAT: Dokumenttien koostaminen, XML, DTD, sisältöjen uudelleenkäyttö, opinto-opas, projekti

## **ABSTRACT**

Rinne, Pekka Juhani

XML Document Assembly. Findings from the Case Study on the IT Faculty  
Study Guide in the University of Jyväskylä / Pekka Rinne

Jyväskylä: University of Jyväskylä, 2005.

118 p.

Master's Thesis

This study is focused on XML document assembly. Document assembly is an activity in which new and existing documents or document fragments are acquired and combined producing multiple outputs for differing users or media. This study reviews the document assembly model developed by Heikkinen. The study also reports findings of a qualitative case study that took place at the Faculty of Information Technology at the University of Jyväskylä between September 2004 and June 2005. On the case study, the assembly of Faculty's student guide from XML-documents edited at place, acquired from a course system interface and from several internal and external content producers was developed and taken into use. In addition of describing the case study and findings from it the thesis compares the XML document assembly model by Heikkinen and the one developed during the case study.

It was perceived that the assembly model developed by Heikkinen was more automatic and thus probably more user-friendly than the assembly model of the student guide. The findings also indicate that the grain size of source documents for assembly should be designed on a case-by-case basis. Moreover, it was discovered that integration between two information systems facilitating the assembly process of the student guide was easy to implement due to the use of XML-technology.

**KEYWORDS:** Document assembly, XML, DTD, content reuse, student guide, project

# SISÄLLYSLUETTELO

1 JOHDANTO.....	6
1.1 Tutkimuksen tausta.....	6
1.2 Tutkimuksen tavoitteet ja tutkimusmenetelmä .....	10
2 DOKUMENTTIEN HALLINTA JA XML-KIELI .....	13
2.1 Dokumenttien hallinta .....	13
2.2 Rakenteinen dokumentti .....	16
2.3 XML-kieli .....	17
2.4 Rakenteiset dokumentit ja dokumenttien hallinta .....	20
2.5 XML-dokumenttien muunnokset.....	24
3 DOKUMENTTIEN KOOSTAMINEN.....	28
3.1 Taustaa .....	28
3.2 Dokumenttien koostamismalli .....	32
3.2.1 Alustusvaihe .....	34
3.2.2 Dokumenttikatkelmien etsintä.....	36
3.2.3 Löydettyjen dokumenttikatkelmien valinta.....	36
3.2.4 Tulodokumentin koostaminen ja erilaiset julkaisut.....	37
3.3 Mallin arviointi .....	38
3.4 Muita koostamismalleja.....	40
4 TAPAUSTUTKIMUKSEN KOHDEALUE .....	43
4.1 Kohdeorganisaatio.....	43
4.2 Projektityöskentelyn opetus tietojenkäsittelytieteiden laitoksella .....	44
4.3 Opinto-opas -dokumentti.....	45
4.4 Ongelmat opinto-oppaan laadinnassa ja koostamisessa .....	47
4.4.1 Teknologiset ongelmat .....	48
4.4.2 Teknologiasta riippumattomat ongelmat.....	49
4.5 Opinto-oppaan kehittämishanke.....	50
4.5.1 XooZoo-projekti.....	52
4.5.2 Xoo-jatko -projekti.....	55
5 TAPAUSTUTKIMUS .....	58
5.1 Viitekehys ja tiedonkeruumenetelmät.....	58
5.2 Opinto-oppaan laatiminen ennen kehittämishanketta .....	60
5.2.1 Toimijat .....	61
5.2.2 Prosessit .....	61
5.2.3 Sisältöyksiköt .....	63
5.2.4 Järjestelmät .....	63
5.3 Opinto-oppaan laatiminen kehittämishankkeen jälkeen.....	64
5.3.1 Toimijat .....	64
5.3.2 Prosessit .....	65

5.3.3 Sisältöyksiköt .....	67
5.3.4 Järjestelmät .....	68
6 OPINTO-OPPAAN KOOSTAMINEN .....	76
6.1 Arvioinnin kohteet .....	76
6.2 Opinto-oppaan koostamismalli .....	77
6.2.1 Lähdesisältöjen hankinta.....	79
6.2.2 Lähdesisältöjen muuntaminen ja esikoostaminen .....	80
6.2.3 Lähdedokumenttien tallennus WWW-palvelimelle .....	80
6.2.4 Lähdedokumenttien valinta ja opaskoosteen määrittely .....	81
6.2.5 Opasversion koostaminen .....	82
6.2.6 Opasversion julkaisu .....	83
6.3 Arviointi.....	84
6.3.1 Koostamismallien vertailu .....	84
6.3.2 Koostamisprosessin automatisointi.....	86
6.3.3 Lähdedokumentit ja niiden raekoko .....	88
6.4 Käytännön kokemuksia opinto-oppaan kehittämishankkeesta .....	91
7 YHTEENVETO .....	93
LÄHDELUETTELO .....	97
LIITE 1: XOOZOO-PROJEKTIN TOIMEKSIANTO.....	105
LIITE 2: OPINTO-OPPAAN LUKU.DTD.....	108
LIITE 3: ENSIMMÄISEN VUODEN OPISKELIJOIDEN OPASVERSION KOOSTEMÄÄRITYS .....	113
LIITE 4: VANHEMPIEN OPISKELIJOIDEN OPASVERSION KOOSTEMÄÄRITYS .....	116

# 1 JOHDANTO

Tässä luvussa esitellään tämän tutkielman peruslähtökohdat, tutkimusongelma, tutkimusmenetelmät ja tutkittavan aihepiirin rajausta.

## 1.1 Tutkimuksen tausta

*Sisällönhallinta* voidaan määritellä työkaluiksi ja menetelmiksi, joiden avulla hallitaan monia erityyppisiä sisältöjä kuten web-sisältöjä tai elektronisia dokumentteja (Honkaranta & Tyrväinen 2005). Boiko (2002) määrittelee sisällönhallinnan informaation ja toiminnallisuuden luomisen ja jakelun hallinnaksi. Sisällönhallinnassa täytyy tietää millaista informaatiota halutaan tarjota, kenelle ja miten. Honkaranta & Tyrväinen (2005) jakavat sisällönhallinnan kolmeen osaluueeseen:

1. Web-sisältöjen hallinta
2. Dokumenttien hallinta ja
3. Rakenteisten dokumenttien hyödyntäminen

Rakenteisten dokumenttien eräs tutkimusalue on dokumenttien koostaminen ja rakenteisten dokumenttien hyödyntäminen koostamisessa (Honkaranta & Tyrväinen 2005). Tässä tutkielmassa keskitytään XML-dokumenttien koostamiseen, joten sisällönhallinnan alueelta tutkielma koskee pääasiassa rakenteisten dokumenttien hyödyntämistä.

Tutkielman aihepiiri liittyy vahvasti myös dokumenttien hallintaan. Dokumenttien hallinnassa on pääpiirteittäin kyse elektronisten dokumenttiversioiden luonnista, muokkaamisesta ja tallentamisesta johonkin dokumenttikokoelmaan, niiden katselemisesta ja hyväksikäyttämisestä sekä lopuksi poistamisesta, kun

niitä ei enää tarvita. (Anttila 2001) Myös tietyn informaation löytyminen tietystä yksittäisestä dokumentista on olennaista (Heikkinen 2000).

Nyky aikaisten tietoverkkojen avulla päästään käsiksi yhä helpommin erilaisiin alati kasvaviin digitaalisiin dokumenttikokoelmiin ja tietovarastoihin. Digitaalisten ja rakenteeltaan heterogeenisten dokumenttien määrän kasvaessa syntyy tarve apuvälineille oikean tiedon löytämiseen, yhdistelyyn ja uudelleenkäyttöön suurista dokumenttimassoista. (Heikkinen 2000; Honkaranta & Tyrväinen 2001) Apuväline edelläkuvattuun ongelmaan on dokumenttien koostaminen. *Dokumenttien koostamisella* (document assembly) tarkoitetaan sisältöjen muodostamista joko aiemmin laadituista kokonaisista dokumenteista tai niiden sisältöosista eli katkelmista (Heikkinen 2000). Aiemmin laadittu sisältö voi olla rakenteisessa muodossa (Salminen, Tiitinen & Lyytikäinen 1999) tai vaikkapa saatavilla tietokannasta. Dokumenttien koostaminen tarkoittaa käytännössä jotakin tiettyä tarkoitusta varten valikoitujen sisältöosien uudelleenkäyttöä halutun koostekokonaisuuden aikaansaamiseksi.

Dokumenttien koostamisella voi olla useita tavoitteita, kuten esimerkiksi

1. uusien, ihmisten tulkittaviksi tarkoitettujen dokumenttien koostaminen (Lehtonen, Petit, Heinonen & Lindén 2002),
2. sisältöjen uudelleenkäyttö (document content reuse; syndicated content) (Boiko 2002; Rockley 2002),
3. ohjelmien välisten dokumenttien tai tiedon siirto EAI:n (Enterprise Application Integration) yhteydessä (Antila & Pohjonen 2001) tai
4. monikanavajulkaisun tehostaminen, mahdollistaen erimuotoiset julkaisut samoista lähdedokumenteista (Lehtonen 2001).

Dokumenttien koostamisesta on siis tavoitteista riippuen olemassa erilaisia näkemyksiä ja erinimisiä käsitteitä, joista etenkin sisältöjen uudellenkäyttö Rockleyn (2002) mukaan on perusajatukseltaan samankaltainen käsite.

Jotta tulodokumentti voidaan koostaa erikseen tallennetuista digitaalisista dokumenteista tai dokumentin osista, on dokumenteissa oltava sen loogista rakennetta ja sisältöä kuvaavaa metatietoa. Tämä on edellytys myös sisältöjen tehokkaalle uudelleenkäytölle. (Honkaranta & Tyrväinen 2001) Rakenteiset dokumentit tarjoavat mahdollisuuden dokumenttien sisällön, esitysmuodon ja loogisen rakenteen erottamiselle käyttäen sovittua merkkäuskieltä. *Rakenteiset dokumentit* ovat ihmisen ja tietokoneen tulkittavissa olevia dokumentteja, joille on laadittu rakennemäärittely. (Honkaranta & Tyrväinen 2005) Rakenteisissa dokumenteissa rakenneosat ovat nimettävissä ja tunnistettavissa siten, että ohjelmat pystyvät käsittelemään niitä. Rakenteisten dokumenttien osat on merkatu sovitulla tavalla, ja merkatut osat muodostavat dokumentin loogisen rakenteen. (Salminen 2004b) Erottamalla dokumentin sisältö esitysmuodosta ja ulkoasumäärittelyksistä, voidaan samasta sisällöstä luoda erilaisia esitysmuotoja erilaisia laitteita varten. Sisältö kirjoitetaan kerran, ja se voidaan esittää monin erilaisin tavoin riippuen tiedon vastaanottajasta ja laitteistosta. Tätä kutsutaan monikanavajulkaisuksi. (Dunn 2003) Uutta rakenteisten dokumenttien standardia edustaa XML-kieli (Extensible Markup Language).

XML (Bray, Paoli, Sperberg-McQueen ym. 2004) on W3-järjestön (W3C 2005) 1990-luvun lopulla määrittelemä suositus merkkäuskieleksi, jolla digitaalisen dokumentin rakenneosat voidaan erottaa toisistaan sovitulla tavalla dokumenttityypissä määriteltyjen elementtitunnisteiden ja attribuuttien avulla. Tämän ansiosta XML-muotoisten dokumenttien looginen rakenne ja sisältö kyetään erottamaan toisistaan ohjelmallisesti (Salminen, Tiitinen & Lyytikäinen 1999; Bray ym. 2004). XML-dokumenttien määrän ennustettiin vuosituhannen vaihteessa lisääntyvän merkittävästi, sillä XML oli jo tuolloin - vain muutama vuosi



syntymänsä jälkeen - saavuttanut erittäin nopeasti valtavan suosion (Heikkinen 2000).

Nykyään XML:n hyödyt tiedonsiirtostandardina on havaittu laajalti, ja sen käyttö on yleistynyt monissa organisaatioissa eri sovellusalueilla digitaalisten dokumenttien hallinnasta elektroniseen liiketoimintaan (Bartocci, Mariani & Merelli 2003; Päivärinta & Munkvold 2005). XML-teknologia on yleistymisensä myötä tullut myös perinteisempien tietokoneiden käyttäjien ulottuville, sillä esimerkiksi eräät nykyaikaiset tekstinkäsittely- ja taulukkolaskentaohjelmistot sisältävät XML-ominaisuuksia. Tekstinkäsittelyohjelmissa XML-ominaisuus tarkoittaa käytännössä perinteisen tekstinkäsittelyohjelman, kuten Microsoft Wordin, käyttämistä XML-tiedostojen muokkaamiseen. (Alcorn, Kaplan & Tucker 2004) Voidaankin todeta, että XML on yleistymässä myös tavallisten tietotekniikan käyttäjien keskuudessa, eikä se täten ole enää pelkästään asiantuntijoiden soveltama teknologia.

Edellä mainittujen ominaisuuksien ansiosta XML sopii myös dokumenttien koostamiseen ja ratkaisemaan ongelmia sisältöjen uudelleenkäytön alueella. Lähdedokumentit voidaan merkata käyttäen sisältöorientoituneita elementtitunnistimenimiä, jonka johdosta mikä tahansa elementtitunnistein merkattu dokumentin osa voidaan tunnistaa ja poimia koostedokumenttiin.

Tutkielmassa esitetty näkemys dokumenttien koostamisesta perustuu Heikkisen (2000) esittelemään koostamismalliin, joka on kuvattu tarkemmin luvussa 3. Mallin periaatteena on poimia erilaisten vaiheiden seurauksena elektronisesta dokumenttikokoelmasta sopivia dokumenttikatkelmia ja muodostaa niistä yleistä rakennemäärittystä noudattava tulodokumentti. Tulodokumentista voidaan tämän jälkeen julkaista erilaisia tulosteita esimerkiksi paperi- tai verkkojulkaisuja varten.

## 1.2 Tutkimuksen tavoitteet ja tutkimusmenetelmä

Tämän tutkielman tavoitteena on selvittää aiemman tutkimuksen perusteella mitä tarkoittaa dokumenttien koostaminen sekä sisältöjen uudelleenkäyttö ja esitellä tapaustutkimuksen avulla yksityiskohtaisesti erään organisaation sovellettu XML-dokumenttien koostamistapa. Tapaustutkimus koskee Jyväskylän yliopiston informaatioteknologian tiedekunnan opinto-oppaan kehittämishanketta, jonka tuloksena tiedekunnan opinto-opas tuotettiin XML-muodossa keuhällä 2005. Tapaustutkimuksen tuloksia peilataan tässä tutkielmassa esiteltyyn dokumenttien koostamisen aikaisempaan tutkimukseen ja tarkastellaan, miltä osin opinto-oppaan koostaminen on samankaltaista tai erilaista kuin aiemmassa tutkimuksessa esitetty koostaminen.

Hirsjärven, Remeksen & Sajavaaran (2000, 123) mukaan tapaustutkimus on yksi traditionaalisimmista tutkimusstrategioista. Heidän mukaansa *tapaustutkimus* on "yksityiskohtaista, intensiivistä tietoa yksittäisestä tapauksesta tai pienestä joukosta toisiinsa suhteessa olevia tapauksia". Tässä tutkielmassa tutkittavassa tapauksessa useista ja erilaisista lähteistä peräisin olevista XML-muotoon muunnetuista lähdedokumenteista koostetaan erilaisten vaiheiden kautta tulodokumentti monikanavajulkaisua varten.

Hirsjärven, Remeksen & Sajavaaran (2000) mukaan tapaustutkimuksen tavoitteena on tyypillisesti ilmiöiden kuvailu. Yinin (1994) mukaan tapaustutkimuksen kohdealueen aiempaan tutkimukseen ja teoriaan tulee tutustua huolella, jotta tapausta voitaisiin arvioida erilaisista näkökulmista ja löytää tarkempia kysymyksiä ja ongelmia. Tässä tutkielmassa tapaustutkimuksen havaintoja verrataan aiemmassa tutkimuksessa kehitettyyn koostamismalliin (Heikkinen 2000), ja tarkastellaan miltä osin tapaustutkimus poikkeaa aiemmassa tutkimuksessa havaituista ilmiöistä ja miksi. Lisäksi tutkielmassa raportoidaan dokumenttien koostamiseen liittyviä havaintoja sekä käytännön kokemuksia.

Yinin (1994) mukaan tapaustutkimuksessa on erityisen tärkeää päättää miten kerättävä tieto yhdistetään asetettuun alkuteoreemaan. Tässä tutkielmassa dokumenttien koostamiseen liittyviä käytännön ilmiöitä ja ongelmia verrataan Heikkisen (2000) esittämään dokumenttien koostamismalliin. Tutkielman tapaustutkimus antaa oivan tilaisuuden tutkia käytännön tapauksen kautta dokumenttien koostamiseen liittyviä ongelmia ja ilmiöitä. Teoria-osuus puolestaan antaa hyvän pohjan verrata käytännön havaintoja kattavaan aiemmassa tutkimuksessa esitettyyn malliin.

Tapaustutkimuksen kohteen valintaan vaikutti kirjoittajan osallistuminen opinto-oppaan uuden XML-pohjaisen laadintaprosessin kehitykseen ja käyttöönottoon. Lisäksi laadintaprosessin ja oppaan uuden julkaisujärjestelmän kokeilu sekä XML-kielen hyödyntämisen tutkiminen kohdealueella innoittivat kirjoittajaa raportoimaan opinto-opas -projektin tuloksista tämän tutkielman muodossa. Tapaustutkimuksen taustalla on Jyväskylän yliopiston informaatioteknologian tiedekunnan opinto-oppaan kehittämishanke, joka jakautui kahteen eri projektiin, XooZoo-projektiin sekä Xoo-jatko -projektiin. XooZoo-projekti (XooZoo 2005c) oli Jyväskylän yliopiston informaatioteknologian tiedekunnan tietojenkäsittelytieteiden laitoksella lukuvuonna 2004–2005 järjestetyn Projektin johtaminen -opintojakson aikana toteutettu opiskelijaprojekti. XooZoo-projektin pääasiallisina tavoitteena oli määrittää opinto-oppaan laadintaprosessi, kartoittaa oppaan käytettävyyttä ja luoda alustavat rakennemääritykset uudelle XML-muotoiselle opinto-oppaalle. XooZoo-projektin jälkeen käynnistettiin tiedekunnan oma jatkokehittämisprojekti, Xoo-jatko -projekti. Xoo-jatko -projektin aikana XooZoo-projektin tuotoksia jatkokehitettiin ja Jyväskylän yliopiston IT-tiedekunnan lukuvuoden 2005-2006 opinto-oppaat tuotettiin koostamalla ne XML-muodossa olevasta lähdemateriaalista. Tutkielman aineistona on käytetty molempien projektien tuotoksia, mutta pääosin tutkimuksen aineisto perustuu Xoo-jatko -projektin tuotoksiin sekä projektin aikana havaittuihin käytännön havaintoihin.

Tapaustutkimuksen aineistoa kerätään yleensä esimerkiksi havainnoimalla tutkittavaa kohdealuetta, haastattelemalla kohdealueen kanssa tekemisissä olevia ihmisiä ja tutkimalla tapaukseen liittyviä dokumentteja (Hirsjärvi, Remes & Sajavaaran 2000). Tämän tutkielman tutkimusmateriaali perustuu pitkälti tutkijan tutustumiseen aiempaan tutkimukseen dokumenttien koostamisesta ja opinto-oppaan kehittämishankkeen aikana syntyneiden tuotosten sekä dokumenttien tutkimiseen. On tärkeää huomata, että tutkija oli itse mukana opinto-opas -projektissa alusta lähtien. Siten suuri osa tutkimuksen materiaalista perustuu myös tutkijan omaan havainnointiin tapauksen käytännön ratkaisuista koko kehittämishankkeen ajalta.

Tapaustutkimuksen arviointi-osa jakautuu lukuihin 5 ja 6. Arviointiosuudessa kuvaillaan ensin yleisemmin opinto-oppaan kehittämishanketta, jonka jälkeen tarkastellaan XML-muotoisen opinto-oppaan koostamismallia sekä oppaiden julkaisuun käytettyä tekniikkaa. Arvioinnissa keskitytään tarkimmin oppaiden koostamiseen ja siihen liittyneisiin ilmiöihin verrattuna aikaisempaan tutkimukseen.

Tapaustutkimuksen arvioinnin ensimmäinen osa, luku 5, noudattaa alun perin dokumenttien hallinnan alueelle kehitettyä (Salminen 2003b) jaottelua toimijoihin (toimijarooleihin), sisältöyksiköihin, teknologioihin, ja prosesseihin. Arvioinnin toinen osa, luku 6, sisältää opinto-oppaan koostamismallin yksityiskohteisemmän kuvauksen ja vertailun tutkielman luvussa 3 kuvattuun Heikkisen (2000) esittelemään dokumenttien koostamismalliin. Yhteenvedossa esitellään tutkielmassa esille tulleita keskeisiä huomioita ja johtopäätöksiä.

## 2 DOKUMENTTIEN HALLINTA JA XML-KIELI

Luvussa esitellään dokumentin, rakenteisten dokumenttien ja dokumenttien hallinnan käsitteet sekä käydään läpi XML-kielen perusasioita. Lisäksi tarkastellaan rakenteisten dokumenttien ja etenkin XML:n suhdetta dokumenttien hallintaan. Luvussa esitellään myös tyylikieli XSL, jonka avulla XML-dokumentteja voidaan muun muassa muuntaa toisiksi XML-dokumenteiksi. Lisäksi luvussa käydään läpi rakenteisten dokumenttien vahvuuksia ja heikkouksia dokumenttien hallinnassa sekä arvioidaan miten tyylikieli XSL:ää voidaan hyödyntää koostettaessa uusia XML-dokumentteja.

### 2.1 Dokumenttien hallinta

*Dokumentti* voidaan määritellä yksittäiseksi tunnistettavissa olevaksi fyysiseksi sisältökokonaisuudeksi, jonka on laatinut yksi tai useampi kirjoittaja (Schamber 1996). Sutton (1996) puolestaan määrittelee dokumentin ihmisen tulkittavaksi tarkoitetuksi tietoyksiköksi. Anttila (2001, 1) taas määrittelee dokumentin olevan ”ihmisen käsiteltäväksi tarkoitettu tietojoukko”. Salmisen (2003) mukaan dokumentilla on seuraavat ominaisuudet:

- Dokumentti on tarkoitettu ihmisen havainnoitavaksi ja ymmärrettäväksi ja se sisältää tietoa jostakin tietystä aiheesta.
- Sillä on sisältö sekä yksi tai useampi esitysmuoto.
- Sen sisältö koostuu osista, osat koostuvat symboleista ja osien rakenne tukee dokumentin ymmärtämistä
- Se tallennetaan jollekin medialle.
- Se voidaan yksilöidä ja käsitellä yksikkönä

Salmisen (2003) mukaan digitaalisella dokumentilla on edellisten lisäksi myös seuraavat ominaisuudet:

- Dokumentin sisältö on tallennettu digitaaliselle medialle.
- Dokumentti on tekemisissä laitteistojen ja ohjelmistojen kanssa, jotka tunnistavat dokumentin osien symbolit ja rakenteen digitaalisesta sisällöstä ja tuottavat niistä ihmisen ymmärtämän esitysmuodon.

Koulopouloksen & Frappaolon (1995) mukaan dokumentilla on myös elinkaari, jonka aikana dokumentti muuttuu sisällöltään tai ulkonäöltään. Anttilan (2001, 5) mukaan dokumentin elinkaari on seuraavanlainen:

1. Dokumentin luonti
2. Dokumentin muokkaus
3. Dokumentin tarkastus / hyväksyntä (voidaan siirtyä kohtaan 2)
4. Dokumentin julkaisu
5. Dokumentin haku / katselu (voidaan siirtyä kohtaan 2)
6. Dokumentin arkistointi
7. Dokumentin poisto

Dokumentteja on perinteisesti suosittu organisaatioiden tietojen säilytyksessä (Honkaranta & Tyrväinen 2005). Suuri osa organisaatioiden dokumenteista on tänä päivänä elektronisessa muodossa, sillä elektronisten dokumenttien hallinta tietokoneen avulla on huomattavasti helpompaa paperidokumentteihin verrattuna (Ashman 2000). Elektronisten dokumenttien ominaisuuksiin kuuluu esimerkiksi joustavampi muokkaus, tallennus elektronisiin dokumenttikokoelmiin ja linkitys muihin dokumentteihin hyperlinkkien avulla (Schamber 1996).

Nykyisin erilaisia dokumentteja tuotetaan valtavia määriä päivittäin. Dokumenttien digitaalisuus mahdollistaa niiden käsittelyn sekä säilyttämisen tietokoneen avulla. Erilaisia dokumenttimuotoja voivat olla esimerkiksi sopimukset, muistiot, laskentataulukot, piirustukset ja sähköpostiviestit. (Anttila, 2001)

Tietomäärän kasvaessa tietoa on pystyttävä hallitsemaan tehokkaasti, sillä samalla myös epäolennaisen tiedon määrä kasvaa. Ongelmaksi muodostuu löytää olennainen ja ajantasainen tieto epäolennaisen tiedon joukosta. Organisaatioissa tulee olla selkeät säännöt ja välineet elektronisten dokumenttien tuottamiseen, tallentamiseen, muokkaamiseen ja tarkasteluun koko dokumentin elinkaaren ajan. Dokumenttien hallinta tukee organisaation toimintaa, prosesseja ja tuottavuutta hallitsemalla dokumenttitietoa.

Dokumenttien hallinnasta voi vastata dokumenttienhallintajärjestelmä (Anttila 2001). Dokumenttienhallintajärjestelmän tärkeitä ominaisuuksia ovat Honkarrannan & Tyrväisen (2005) ja Anttilan (2001) mukaan esimerkiksi:

- dokumenttien versionhallinta,
- käyttäjäroolien ja dokumenttien käyttöoikeuksien hallinta,
- tuki dokumenttien elinkaarelle ja muille työkuluille,
- dokumenttien metatietojen/ominaisuustietojen hallinta,
- monikanavajulkaisu useille eri laitteille ja erilaisille medioille,
- dokumenttien sisään/uloskuittaus (järjestelmä lukitsee/vapauttaa muokattavan dokumentin päällekirjoitusten varalta) sekä
- dokumenttien luokittelu sisällön mukaan ja haku.

Paikannettaessa tallennettuja dokumentteja järjestelmästä on tiedettävä mitä etsiä. Tallennetut dokumentit tulee voida paikantaa helposti tarkastelua ja muokkaamista varten ja myös halutun tiedon löytäminen yksittäisestä dokumentista tulee olla mahdollista (Heikkinen 2000). Anttilan (2001) mukaan dokumenttienhallinta onkin enemmän toimintamalli kuin tekninen kysymys. Tosin teknologia ja sen oikeaoppinen käyttö ovat kriittisiä asioita dokumenttien hallinnan käytännön toteutuksessa. Honkarannan (2003) mukaan organisaatioiden dokumenttienhallinta on työkalu, joka tukee organisaation liiketoimintaprosesseja sekä tehtäviä, joissa ihmiset ovat pääosassa erilaisissa rooleissa ja ryhmissä.

## 2.2 Rakenteinen dokumentti

Elektroninen dokumentti voi olla rakenteisessa muodossa, jolloin sitä kutsutaan rakenteiseksi dokumentiksi. *Rakenteisissa dokumenteissa* rakenneosat ovat nimettävissä ja tunnistettavissa siten, että ohjelmat pystyvät käsittelemään niitä. Rakenteisten dokumenttien osat merkataan jollakin sovitulla tavalla, jolloin merkatuista osista muodostuu dokumentin looginen rakenne. Rakenteisia dokumentteja voidaan hyödyntää sekä pysyvänä tallennusmuotona että tiedonsiirrossa (Korhonen 2003; Salminen 2004b).

1980-luvulla standardoitu merkkauškieli SGML (Standard Generalized Markup Language) perustuu ajatukseen tiedon esittämisestä rakenteisina dokumentteina. SGML:n perusajatuksena on dokumentin rakenteen, sisällön sekä ulkoisen esitysmuodon erottaminen toisistaan. Tällä pyritään joustavaan dokumenttien ja dokumentteihin sisältyvän tiedon hallintaan. XML-kieli perustuu samaan ajatukseen. (Salminen 2004b) XML-kieli on yksinkertaistettu versio SGML-kielestä ja se on kehitetty informaation vaihtoon Internetissä. XML on SGML:n yksinkertaistettu osajoukko. (Salminen 2004a; Salminen, Tiitinen & Lyytikäinen 1999)



SGML- ja XML-dokumenttien dokumenttityyppien määrittelemiseen on käytävissä *dokumenttityypin määrittely* eli *DTD* (Document Type Definition), joka sisältää säännöt siitä, minkä nimisiä loogisen rakenteen osia dokumentissa saa olla, ja missä järjestyksessä (Salminen 2004a).

### 2.3 XML-kieli

*XML* (*Extensible Markup Language*), on W3-järjestön kehittämä rakenteisten dokumenttien merkkauškieli, joka on johdettu SGML-kielestä. XML on avoin standardi, ja se kehitettiin Internetin tiedon hallintaa silmälläpitäen. XML sisältää joukon sääntöjä, joiden avulla informaatio voidaan määritellä ja esittää rakenteisina XML-muotoisina dokumentteina. XML-dokumenteissa informaatio-rakenteet voidaan nimetä ja merkitä yksikäsitteisen merkkauksen avulla, jolloin sovellukset pystyvät käsittelemään niitä. XML on metakieli, jonka avulla XML-dokumenttien sisällön kuvaaminen onnistuu yhtenäisellä tavalla. (Salminen 2004a; Salminen 2005) Näiden ominaisuuksien ansiosta XML-dokumenttien looginen rakenne ja itse sisältö voidaan erottaa toisistaan.

XML-dokumenteilla on fyysinen ja looginen rakenne. *Fyysinen rakenne* koostuu entiteeteiksi kutsutuista yksiköistä. *Entiteetit* ovat XML-dokumenttiin kuuluvia nimettyjä ja itsenäisiä osia, jotka voivat viitata toisiin entiteetteihin. Entiteetti voi sisältää tekstiä tai binääridataa tai molempia. Dokumentti muodostuu entiteettien joukosta viittauksien välityksellä. Koko XML-dokumentin käsittävä entiteetti on nimeltään dokumenttientiteetti (tai juurientiteetti).

*Looginen rakenne* koostuu nimetyistä elementtitunnisteista. *Elementtitunnisteet* voivat sisältää toisia elementtitunnisteita eli lapsielementtitunnisteita, tai dokumentin varsinaista tekstisisältöä. Elementtitunnisteiden sisältöä voidaan tarkentaa *attribuuttien* avulla. XML-dokumentti sisältää aina yhden *juurielementtitunnisteen*, jonka sisällä kaikki muut elementtitunnisteet ovat. Elementtitunnis-

teet muodostavat täten hierarkkisen puumaisen rakenteen. XML on laajennettava merkkiauskieli, jossa käyttäjä voi itse määrittellä käyttämänsä elementtitunnisteet ja attribuutit DTD:ssä. (Salminen 2004a) Dokumenttien koostamisen kannalta DTD on erittäin tärkeä. Lähdedokumentin looginen rakenne on tunnistettavissa DTD:stä, jolloin lähdedokumentin ohjelmallinen jatkokäsittely koostamisprosessissa onnistuu.

Kuvio 1 havainnollistaa XML-dokumentin loogista rakennetta.

Esittely	<?xml version="1.0" encoding="ISO-8859-1"?>
Elementtitunnisteet ja attribuutit sekä sisältödata	<pre> &lt;autonohjekirja&gt;   &lt;huolto&gt;     &lt;moottori&gt;       &lt;osa id="öljytikku"&gt;         &lt;ohje&gt;Vedä tikku, tarkasta öljy&lt;/ohje&gt;       &lt;/osa&gt;     &lt;/moottori&gt;   &lt;/huolto&gt;   &lt;ajaminen&gt;     &lt;kesa&gt;Käytä kesällä ilmastointia.&lt;/kesa&gt;     &lt;talvi&gt;Varo mustaa jäätä talvella.&lt;/talvi&gt;   &lt;/ajaminen&gt; &lt;/autonohjekirja&gt; </pre>
Kommentti	<!-- Tekijä: Pekka Rinne -->

Kuvio 1. XML-dokumentin osat

Kuvion 1 XML-dokumentin alussa sijaitsevassa esittelyosassa on ohjeita XML-dokumentteja käsittelevälle XML-jäsentimelle (XML parser). *XML-jäsentin* on ohjelma, joka tarkistaa onko sille annettu syöte XML-kielen ilmaisu eli XML-dokumentti. (Salminen 2004c). Esittelyosio kertoo XML-jäsentimelle, että dokumentti on XML-kielen dokumentti ja mitä merkistöä se käyttää. Itse dokumentin sisältö ja merkkiaus on XML-dokumentin juurielementtitunnisteen (Kuviossa 1 autonohjekirja) sisällä. Juurielementtitunniste sisältää lapsielementti-

tunnisteita sekä sisältödataa. Esimerkin elementtitunniste <osa> sisältää myös attribuutin nimeltä id, joka tarkoittaa tietoa kertoen moottorinosan olevan öljytikku. Kommentit eivät ole XML-dokumentin pakollinen osa, mutta niihin voi lisätä vaikkapa XML-dokumenttiin liittyviä metatietoja, kuten esimerkissä on lisätty dokumentin tekijätiedot.

XML-kielessä dokumenttien rakenneosien nimet ja niiden väliset suhteet määritellään DTD:ssä. Kuka tahansa voi määritellä DTD:n, joka kuvaa käytettävän dokumenttityypin rakenteen. Tämä mahdollistaa XML-dokumenttien laatimisen ja räätälöimisen tapauskohtaisesti toisin kuin esimerkiksi HTML-kielessä, jossa rakenne-elementtitunnisteet noudattavat ennalta määrättyä DTD:tä ja kuvaavat lähinnä ulkoasuun liittyviä ominaisuuksia (Bray ym. 2004; Salminen, Tiitinen & Lyytikäinen 1999). Kuviossa 2 esitellään kuviossa 1 esitellyn XML-dokumentin yksinkertaistettu DTD.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!ELEMENT autonohjekirja (huolto, ajaminen)>
<!ELEMENT huolto (moottori)>
<!ELEMENT moottori (osa+)>
<!ELEMENT osa (ohje)>
<!ATTLIST osa
id ID #REQUIRED>
<!ELEMENT ohje (#PCDATA)>
<!ELEMENT ajaminen (kesa, talvi)>
<!ELEMENT kesa (#PCDATA)>
<!ELEMENT talvi (#PCDATA)>
```

Kuvio 2. esimerkki DTD-määrittämisestä

Kuvio 2 on esimerkki DTD-määrittämisestä, jossa on määritelty elementtitunnisteiden nimet ja mahdolliset attribuutit. Elementtitunnisteet voivat sisältää toisia elementtitunnisteita eli lapsielementtitunnisteita tai sisältöä. Esimerkissä autonohjekirja-juurielementtitunnisteen lapsielementtitunnisteita ovat huolto ja

ajaminen. Mikäli elementtitunniste sisältää tekstipohjaista sisältöä kuten kuvion 2 ohje, se määritellään kirjoittamalla elementtitunnisteen sisältötyypiksi #PCDATA. Symboli + osa-elementtitunnisteen jälkeen tarkoittaa, että moottori-elementtitunnisteella on vähintään yksi lapsielementtitunniste nimeltä osa.

XML-dokumentti voi olla hyvin muodostettu (well formed) tai validi (valid). XML-dokumentti on *hyvin muodostettu*, jos

- sillä on yksi tai useampia elementtitunnisteita,
- sillä on vain yksi juurielementtitunniste,
- elementtitunniste alkaa ja loppuu saman elementtitunnisteen sisällä,
- entiteetit ovat hyvin muodostettuja eli elementtitunniste ei voi loppua eri entiteetissä kuin se alkaa ja
- elementtitunnisteen sisältö täyttää standardin määrittelemät hyvin muodostetun XML-dokumentin vaatimukset.

XML-dokumentti on *validi*, jos se on hyvin muodostettu ja noudattaa siihen liittyvän DTD:n määrittämää kielioppia. Lisäksi validissa XML-dokumentissa rakennemäärittelyyn täytyy olla viittaus ennen ensimmäistä elementtitunnistetta. XML-jäsentimen tehtävänä on tarkastaa, onko XML-dokumentti hyvin muodostettu tai validi. (Antila & Pohjonen 2001; Bray ym. 2004; Salminen 2004a)

## 2.4 Rakenteiset dokumentit ja dokumenttien hallinta

Yksi dokumenttien hallinnan tutkimusalueista keskittyy rakenteisiin dokumentteihin, ja niissä olevien loogisten sisältöyksiköiden hyödyntämiseen dokumenttien hallinnassa. Rakenteisten dokumenttien avulla voidaan saavuttaa hyötyjä dokumenttien hallinnassa perinteiseen dokumenttitietoon verrattuna.

SGML- ja XML-kielissä looginen sisältöyksikkö on erotettavissa merkkauksen avulla (Honkaranta 2003). Täten rakenteisten dokumenttien avulla erikokoisten ja erimuotoisten tekstidokumenttien looginen rakenne ja sisältö voidaan haluttaessa merkata tarkasti, jolloin tiettyyn informaatioon päästään helposti ja nopeasti käsiksi ohjelmallisesti. Perinteisen dokumenttitiedon tapauksessa joudutaan tyytymään perinteiseen vapaaseen tekstihakuun, sillä sisältöä saatikka rakennetta kuvaavaa metatietoa ei perinteisissä dokumenttimuodoissa ole tarjolla.

Anttilan (2001) mukaan erityisesti XML-kielen olennaisia asioita dokumenttien hallinnan kannalta ovat seuraavat:

- dokumenttia kuvaavat ominaisuustiedot saadaan siirtymään dokumentin mukana,
- dokumentin rakenne kulkee dokumentin mukana eikä sitä tarvitse kuvata erikseen,
- dokumentit ovat laitteistoriippumattomia,
- dokumenteilla on hyvä säilyvyys johtuen avoimesta rakenteesta,
- useiden päätelaitteiden tuki on helppo toteuttaa ja
- XML tukee eri kielisten ja eri merkistöjä käyttävien dokumenttien tekoa.

Anttila kuitenkin huomauttaa, että ”suurimmat hyödyt saavutetaan silloin, kun toimitaan hajautetussa organisaatiossa ja kaikilla on valmiudet XML:n käyttöön”.

Suuri osa organisaatioiden informaatiosta on säilöttynä elektronisissa dokumenteissa, jotka taas ovat usein tallennettuna jossakin dokumenttitietokannassa. (Ashman 2000). Kun tietokannasta haetaan tietoa, palautetaan käyttäjälle

tyypillisesti kokonainen dokumentti, vaikka haluttu tieto saattaa olla vain tietty osa dokumenttia. (Wilkinson 1994). Perinteiseen dokumenttitietoon verrattuna rakenteisten dokumenttien, kuten XML-dokumenttien, osat voidaan merkitä sisältöä kuvaavilla tunnisteilla, jolloin on mahdollista käsitellä kokonaista dokumenttia pienempiä osia. Tämän ansiosta taas itse dokumentin sisältöön on mahdollista tehdä esimerkiksi perinteistä vapaata tekstihakua tarkempia hakuja. Sisältöä kuvaavien elementt tunnisteiden avulla myös rakenteisten dokumenttien osien ohjelmallinen uudelleenkäyttö on mahdollista.

Rakennemäärittelyn huolellinen suunnittelu avustaa organisaation dokumenttien laatimista ja tallennettavan tiedon oikeellisuutta: kun organisaation informaation tuottamista varten suunnitellaan tarpeita vastaavat rakennemäärittelyt, voidaan varmistua, että kaikki dokumentit tulee merkattua suunnitellulla tavalla. Tämä edellyttää kuitenkin sitä, että dokumentit tallennetaan rakennemäärittelyksiä noudattaen, josta esimerkiksi editori tai viime kädessä dokumenttienhallintajärjestelmä huolehtii. (Dunn 2003)

XML-määrittelyn avoimuuden ansiosta XML:n käyttö ei ole riippuvaista mistään tietystä ohjelmistotoimittajasta. XML-kielen hyödyntäminen edellyttää ainoastaan ohjelmistoja, jotka tukevat XML-spesifikaatiota. (Brown 2003)

Mikäli DTD suunnitellaan pelkästään oman organisaation tarpeet huomioiden, on vaarana, että dokumenttien rakennemäärittelyä voidaan hyödyntää tehokkaasti vain oman organisaation sisällä. Monille teollisuuden aloille kuten esimerkiksi elektronisen liiketoiminnan alueelle on kehitetty omia yleisiä määrittelyjä, joita yritykset voisivat käyttää yhteisten elektronisten dokumenttien rakenteiden tuomien etujen saavuttamiseksi. (Li 2000) Yhteisten määrittelyjen käyttöönotto säästää organisaation vaivalta aloittaa oman rakennemäärittelyn suunnittelu tyhjästä. Lisäksi se takaa dokumenttien yhteensopivuuden samaa määrittelyä käyttävien organisaatioiden kanssa esimerkiksi elektronisessa kaupankäynnissä tai organisaatioiden välisessä tiedonsiirrossa (Dunn 2003). Yhtei-

sissä määrityksissä on kuitenkin se ongelma, etteivät ne välttämättä ota huomioon kaikkien yritysten mahdollisia erityistarpeita. Mikäli yleinen määrittely otetaan käyttöön, tulisi luonnollisesti valita sellainen määrittely, josta on mahdollisimman suuri hyöty oman organisaation sisällä sekä yhteistyökumppaneiden kanssa.

XML:n käyttöönotto organisaatioissa vie usein paljon aikaa ja aiheuttaa kustannuksia. Organisaation tuottamat dokumentit on analysoitava sekä muunnettava rakenteiseen muotoon ja henkilökunta täytyy kouluttaa uusiin työtapoihin sekä rakenteisten dokumenttien ideologiaan. XML:n käyttöönotto vaatii usein myös uusien työkalujen ja järjestelmien hankinnan, käyttöönoton, koulutuksen sekä ylläpidon. (Brown 2003)

Käyttöönottokustannukset kasvavat suhteessa organisaation kokoon. Janssenin ja Sandahlin (1996) havaintojen mukaan erään suuren organisaation yhden prosessin dokumenttienhallinnan toteuttaminen SGML-kielen avulla vei kehittäjiltä paljon resursseja etenkin projektin alkupuolella, koska työtavat, teknologia ja työkalut olivat koulutettavalle henkilökunnalle uusia. Erityisen paljon aikaa vei DTD-suunnittelu; sen kanssa jouduttiin tekemään paljon kompromisseja lukuisien sisällöntuottajien erilaisten tarpeiden vuoksi. Myös työkalujen, kuten SGML-editoreiden kanssa työskentely tuntui sisällöntuottajista vaivalloiselta. Editorien käytettävyys ja kehittyneisyys ei siis ollut korkeatasoista, vaikka SGML-standardi oli jo tuolloin kymmenisen vuotta vanha.

Rakenteiset dokumentit ovat myös kooltaan raskaita, sillä merkkkaus sisältää paljon informaatiota itse dokumentin sisällöstä. Tällaista tietoa kutsutaan metatiedoksi tai tässä tapauksessa paremminkin rakenteiseksi metatiedoksi. Metatiedon sanotaan olevan tietoa tiedosta (Lyytikäinen 2004). Metatiedon, eli tässä tapauksessa merkkauksen kuljettaminen dokumentin mukana elementtitunnisteissa ja attribuuteissa on raskasta, varsinkin kun tietoliikenne on runsasta.

Suurten XML-dokumenttien tallentaminen ja jäsentäminen vaatii täten myös paljon resursseja organisaation laitteistolta. (Vaughan-Nichols 2003)

XML-kielen laajennettavuuden ansiosta jokainen voi määrittellä itse tarvitsemansa dokumenttityypin. Tämä voi johtaa kuitenkin useiden erilaisten rakennemäärittelyiden variantteihin, jotka saattavat olla käytössä samankaltaisissa asiayhteyksissä. Tällaiset heterogeeniset rakennemäärittelyt tuottavat ongelmia etenkin silloin, kun rakenteisia dokumentteja ja niiden osia halutaan uudelleenkäyttää ja koostaa ohjelmallisesti uusiksi dokumenteiksi. (Heikkinen 2000)

XML-dokumenttien käyttöön tiedonsiirrossa saattaa liittyä myös tietoturvaohjeita. XML-dokumentin sisältö on yleisimmin suojaamatonta tekstidataa, joka voi helposti joutua väärin käsiin siirrettäessä dataa suojaamattoman verkon yli. Ongelma on suuri varsinkin silloin, kun dokumentti sisältää salaista tietoa. XML-dokumentteja ei ole useinkaan suojattu esimerkiksi kryptaamalla, sillä se lisäisi jo valmiiksi suuren XML-dokumentin tiedostokokoa. (Vaughan-Nichols 2003) Jonkinlainen digitaalinen käyttöoikeuksien hallinta tai suojattu yhteys olisi aina paikallaan siirrettäessä XML-dokumentteja verkon yli.

## 2.5 XML-dokumenttien muunnokset

W3C-järjestön määrittämä tyylikieli XSL (Extensible Stylesheet Language) (Adler, Berglund, Caruso ym. 2001) on yksi XML:n tärkeistä liitännäisstandardeista. XSL jakautuu kahteen eri kieleen: muunnoskieleen XSLT (XSL Transformations) (Clark 1999) ja muotoilukieleen XSL, josta sekaannuksien välttämiseksi käytetään myös lyhennettä XSL-FO (XSL-Formatting objects).

XSLT-kielen avulla XML-dokumentti tai sen osa voidaan muuntaa toisen DTD:n mukaiseksi XML-dokumentiksi tai jonkin muun merkkaukielen mukaiseksi dokumentiksi. (Korhonen 2003) Tämä on erityisen olennaista koostettaes-



sa uusia dokumentteja. Mikäli esimerkiksi XML-dokumenttikokoelmasta poimitaan koosteeseen erilaisten rakennemääritysten mukaisten dokumenttien osia, on ne muunnettava uuden rakennemäärittelyn mukaiseksi, jotta dokumentti olisi validi. (Heikkinen 2001)

Yksinkertaisesti selitettynä XSLT:n avulla lähdedokumentista voidaan poimia jonkin tietyn elementtitunnisteen tai attribuutin sisältö ja tulostaa se tulosdokumenttiin jonkin tietyn elementtitunnisteen tai attribuutin sisällöksi. Tätä toimintaperiaatetta voidaan hyödyntää koostettaessa uutta dokumenttia useista lähdedokumenteista tai niiden osista.

XSLT on XML-kielen syntaksin mukainen muunnoskieli, joka sisältää ohjeet toimenpiteistä, jotka lähdedokumentille halutaan suorittaa. XSLT-kieltä voidaan hyödyntää esimerkiksi siten, että koostamisen tuloksena syntynyt XML-dokumentti muunnetaan toiseksi XML-dokumentiksi tai vaikkapa XHTML-määrittelyn (Pemberton, Austin, Axelsson ym. 2000) mukaiseksi dokumentiksi verkkojulkaisua varten. (Clark 1999; Korhonen 2003)

Käytännön tasolla muunnoksesta vastaa XSLT-prosessori. Lähdedokumenttina on hyvin muodostettu XML-dokumentti, joka halutaan muuntaa jonkin toisen merkkauškielen mukaiseksi tulospuuksi ja lopulta tulosdokumentiksi XSLT-muunnostiedoston avulla. XSLT-muunnostiedostossa on määritelty mitkä osat, elementtitunnisteen tai attribuutit, XML-dokumentista otetaan mukaan muunnokseen, ja minkä merkkauškielen mukaiseksi ne muunnetaan. Tulosdokumentti voi olla myös tekstidokumentti. (Korhonen 2003)

XSLT-muunnostiedosto voi sisältää myös viittauksia ulkoisiin entiteetteihin, kuten toisiin XML-dokumentteihin. Tämä mahdollistaa tulosdokumentin koostamisen useista eri lähdedokumenteista tai niiden osista XSLT-muunnostiedoston, XML-jäsentimen ja XSLT-prosessorin avulla.

Mikäli tulosedokumentille halutaan määritellä esitystapa, täytyy XSLT-muunnostiedoston sisältää myös esitystapaa ja sisällön asemointia koskevat säännöt. XSL-FO on lyhyesti kuvattuna muotoilukieli, jonka avulla XML-dokumentin sisältö voidaan muuntaa erilaisiin esitysmuotoihin, joita erilaiset sovellukset osaavat esittää. XSL-FO sisältää muotoiluohjeita, joilla XML-dokumentin sisältö voidaan tarkkaan asemoida esityksessä. Tässä tapauksessa XSLT-prosessorin tekemän muunnoksen jälkeen tulospuu ohjautuu muotoiluvaiheeseen, jossa XSL-muotoilija muuntaa XML-dokumentin haluttuun esitysmuotoon, kuten esimerkiksi PDF tai RTF. (Korhonen 2003)

Vaikka XSLT tarjoaakin teknologisen ratkaisun XML-dokumenttien muuntamiseen toisenlaisten rakennemääritysten mukaisiksi XML-dokumenteiksi, se ei yksin ratkaise ongelmaa, joka koskee samansisältöisiä, mutta eri tavalla merkattuja dokumenttisisältöjä. XSLT-muunnostiedosto sisältää säännöt siitä, mitkä elementtitunniste- tai attribuuttisisällöt poimitaan lähdedokumenteista mukaan tulosedokumenttiin ja haluttaessa myös esitystavan tulosedokumentille.

Ongelmaksi muodostuu kuitenkin se, mistä tiedetään minkä nimisistä elementtitunnisteista tai attribuuteista tieto poimitaan, kun elementtitunnisteiden nimet ovat XML:ssä vapaasti määriteltävissä. Ongelma korostuu varsinkin silloin, kun dokumenttikokoelma on suuri ja samansisältöisiä mutta eri tavalla merkattuja relevantteja lähdedokumentteja on paljon. Ongelma toistuu myös rakennemäärittelyn muuttuessa: XSLT-muunnostiedoston toiminta on riippuvainen DTD:ssä määritellyistä elementtitunniste- ja attribuuttinimistä, joten DTD:n muuttuessa myös XSLT-muunnostiedostoa on muutettava. Esimerkiksi Jenssen & Sandahl (1996) ovat havainneet DTD:n olevan erittäin muutosaltis rakenteisten dokumenttien kehittämishankkeissa työskennellessä. Tästä voidaan päätellä, että myös XSLT-muunnostiedostot vaativat jatkuvaa päivitystä DTD:iden ohella. Tämä puolestaan lisää ylläpidon työtaakkaa.

Heikkinen (2000) on esitellyt väitöskirjassaan dokumenttien koostamismallin, joka ottaa edellä mainitun rakenteisten dokumenttien heterogeenisiin rakennemäärittelyihin liittyvän ongelman huomioon. Mallin toiminta esitellään seuraavassa luvussa, ja sitä käytetään tämän tutkielman perusnäköyksenä dokumenttien koostamisesta.

### 3 DOKUMENTTIEN KOOSTAMINEN

Luvussa selvitetään mitä dokumenttien koostaminen ja sisältöjen uudelleenkäyttö tarkoittaa ja esitellään dokumenttien koostamismalli Heikkisen (2000) mukaan. Malli toimii tutkielman perusnäkökulmana dokumenttien koostamisesta johon tapaustutkimuksen havainnot myöhemmin verrataan. Lopuksi mallia arvioidaan sisältöjen uudelleenkäytön kannalta.

#### 3.1 Taustaa

Sisältöjen uudelleenkäyttö on ollut dokumenttienhallinnan tavoitteena jo pitkään. Itse asiassa niin pitkään, kuin dokumentteja on ollut. Dokumentit päivittyvät usein inkrementaalisesti, joka tarkoittaa sitä, että monet dokumentin osat säilyvät sellaisenaan kun taas toiset kaipaavat lisäyksiä. Myös uusia dokumentin osia saattaa syntyä. Uusien dokumenttien luonnissa käytetään usein hyväksi aiemmin laadittuja sisältöjä tai niiden palasia. Sisältöjen uudelleenkäyttö edellyttää relevantin materiaalin löytämistä, materiaalin muokkaamista ja sisältöpalasten yhdistämistä oikealla tavalla. (Levy 1993) Rockley (2002) puolestaan määrittelee sisältöjen uudelleenkäytön toiminnaksi, jossa olemassa olevia sisältökomponentteja käytetään hyväksi uusien dokumenttien luomisessa. Tällaisia sisältökomponentteja ovat esimerkiksi tekstimuotoiset kappaleet, lauseet tai vaikkapa yksittäiset sanat.

Miksi sitten sisältöjä tulisi käyttää uudelleen? Monissa tilanteissa valmiita sisältöjä täytyy käyttää uudelleen erilaisissa tilanteissa esimerkiksi eri asiakkaiden kanssa. Olemassa olevia sisältöjä sopivasti muokkaamalla samankaltaisia sisältöjä ei tarvitse aina tuottaa uudelleen. Uudelleenkäytössä voidaan huomioida koko XML-dokumentti tai vain osa siitä. Sisältöjä voi olla myös tarpeen käyttää uudelleen esimerkiksi sellaisissa tilanteissa, joissa usean eri tuotteen jotkin valmistusvaiheet tai vaikkapa tekniset kuvaukset ovat samoja. Tällöin on järkevää

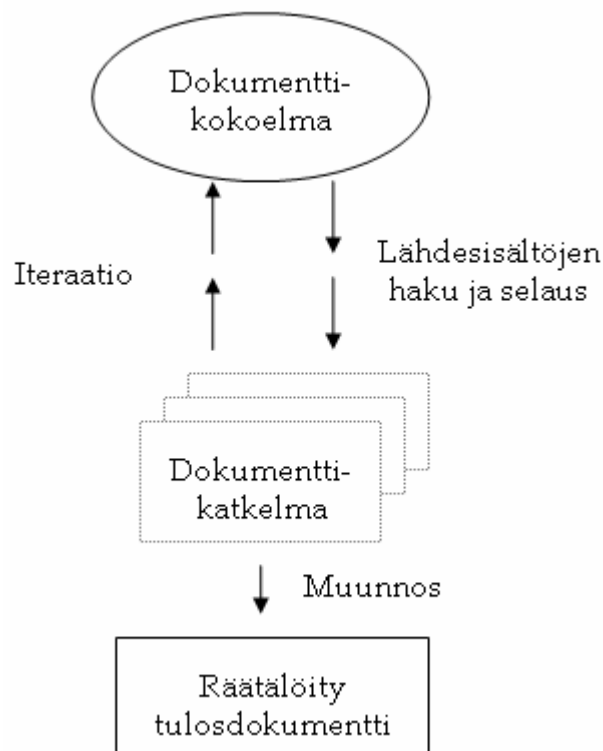
käyttää yhtä sisältöä useassa paikassa, joihin se sopii. Sisältö tarvitsee kirjoittaa ja tallentaa vain yhteen paikkaan, josta sisältö kussakin uudelleenkäyttötapauksessa haetaan. Lisäksi, kun uudelleenkäytettävä sisältö on tallennettuna vain yhdessä paikassa, tarvitsee muutoksetkin tehdä vain yhteen paikkaan sisällön vaatiessa päivittämistä. Mikäli organisaation informaatio on yhdenmukaisesti suunnitellussa rakenteisessa muodossa kuten XML-muodossa, on merkatun sisällön uudelleenkäyttö helppoa. Organisaation tarpeita varten suunniteltu rakennemäärittäminen, kuten DTD (Bray ym. 2004) tai XML Schema (Fallside & Walmsley 2004) pakottaa sisällöntuottajat kirjoittamaan yhdenmukaista tekstiä. Tämä puolestaan varmistaa sisällön kuvaamisen metatiedon, eli elementtitunnisteiden ja attribuuttien avulla. (Dunn 2003) Rockleyn (2002) mukaan sisältöjen uudelleenkäytön etuja ovat esimerkiksi:

- sisältöjen yhdenmukaisuus,
- pienemmät sisällöntuotannon ja sisältöjen ylläpidon kustannukset sekä
- sisältöjen nopea muokattavuus.

Sisältöjen uudelleenkäytöllä voidaan myös saavuttaa kustannussäästöjä organisaatiossa, joissa suuri määrä liiketoiminnallista tietoa on varastoituna elektroniseen muotoon. Mikäli jokin yrityksen osasto esimerkiksi julkaisee jonkin yrityksen tuottaman tuotteen teknisen kuvauksen, olisi järkevää uudelleenkäyttää jo olemassa olevia muiden osastojen tuottamia tuotteeseen liittyviä sisältöjä eikä luoda niitä uudelleen kuvausta varten. Mikäli informaatio on tuotettu yhtenäisellä tavalla XML-muotoon, sisältöjen uudelleenkäyttö on suoraviivaista ja säästää kustannuksia. Lisäksi yrityksen tuottamilla dokumenteilla on yhtenäinen muoto, joka helpottaa sisältöjen uudelleenkäyttöä. (Dunn 2003) Ongelmia kuitenkin syntyy, mikäli dokumenttien merkkkaus ei ole yhdenmukaista. Silloin tarvitaan erilaisia työkaluja relevanttien sisältöjen löytämiseen dokumenttiasasta.

Dokumenttien koostaminen Heikkisen (2000) mukaan tarkoittaa pitkälti samaa asiaa kuin esimerkiksi Rockleyn (2002) tai Boikon (2002) esittämä sisältöjen uudelleenkäyttö. *Dokumenttien koostaminen* on Heikkisen (2000) mukaan tietokoneavusteista uusien dokumenttien muodostamista digitaalisista dokumenttikokoelmista, dokumenteista ja dokumentin osista. Toiminta tarkoittaa halutun tulodokumentin kannalta merkityksellisten kokonaisten dokumenttien tai dokumentin palasten, eli dokumenttikatkelmien, löytämistä, muokkaamista ja yhdistämistä koostedokumentiksi. Dokumenttien koostaminen ja sisältöjen uudelleenkäyttö voidaan nähdä samankaltaisina asioina. Erotuksena se, että Heikkisen (2000) esittelemä XML-dokumenttien koostamisen malli tuo teknologiset ratkaisut sisältöjen uudelleenkäytön problematiikkaan.

Kuviossa 3 on kuvattu Heikkisen (2000) esittelemän XML-dokumenttien koostamismallin periaate karkealla tasolla.



Kuvio 3. Dokumenttien koostamisen periaate

Kuviossa 3 ellipsi kuvaa XML-lähdedokumenteista koostuvaa kokoelmaa ja nuolet dokumenttien koostamisprosessin kulkusuuntaa. Katkoviivalla kuvatut laatikot ovat koostamisen välituloksia ja yhtenäisellä viivalla kuvattu laatikko lopputulos, josta voidaan tehdä erilaisia julkaisuja.

Mallissa perustana on kokoelma XML-dokumentteja. Kokoelmasta haetaan ja valitaan iteratiivisesti sopivia dokumenttikatkelmia, jotka tunnistetaan XML-merkkauksen perusteella joko automaattisesti, tai haluttaessa myös manuaalisesti. Tämän jälkeen osaset yhdistetään ja tarvittaessa muunnetaan eri muotoon. Valituista ja yhdistellyistä dokumenttikatkelmista koostetaan uusi dokumentti, eli räätälöity tulosdokumentti. Tulosdokumentista on mahdollista julkaista erilaisia tulosteita erilaisiin käyttötarkoituksiin, kuten Internetiin, matkapuhelimiin tai vaikkapa paperille.

Jotta sisältöjen uudelleenkäyttöä olisi mahdollista tukea tietokoneavusteisesti, täytyy sisältöjen, eli dokumenttien olla sellaisessa muodossa ja sisältää tarvittavia metatietoja, jotta tietokoneet voisivat suorittaa dokumenteille haluttuja toimenpiteitä (Honkaranta & Tyrväinen 2001). Kun dokumentit on saatettu sopivaan muotoon, sovellukset voivat käsitellä niitä halutulla tavalla metatietojen ja käyttäjän antamien kommentojen avulla. XML tarjoaa tähän tarvittavat ominaisuudet, sillä XML-dokumenttien looginen rakenne ja sisältö on erotettu toisistaan. XML-dokumentin merkkkaus sisältää tarvittavat metatiedot, jotta ohjelmallinen uudelleenkäyttö on mahdollista. Tavoitteena on yksi dokumentti moneen käyttötarkoitukseen, eikä monta dokumenttia kukin omaa käyttötarkoitustansa varten. Tämän ansiosta esimerkiksi dokumentteihin kohdistuvat päivitykset on tarpeellista tehdä vain lähdedokumentteihin (Lehtonen 2001) Dokumenttien koostamisen ehkäpä tärkein seikka on valmiiden sisältökomponenttien uudelleenkäyttö.

### 3.2 Dokumenttien koostamismalli

Tämän tutkielman teoria dokumenttien koostamisesta perustuu Heikkisen (2000) esittelemään XML-dokumenttien koostamismalliin. Kyseessä on puoliau- tomaattinen XML-dokumenttien koostamisjärjestelmä. Puoliautomaattisuus tarkoittaa sitä, että järjestelmän käyttäjä osallistuu käyttöliittymän välityksellä koostamisprosessiin tietyissä vaiheissa antaen järjestelmälle koostamisessa tar- vittavia komentoja ja parametreja. Tietyt vaiheet järjestelmä hoitaa automaatti- sesti. Koostaminen on interaktiivinen prosessi, jossa käyttäjä ohjaa järjestelmää tietyissä vaiheissa koostaakseen haluamansa dokumentin sopivista lähdedo- kumentin palasista, joita kutsutaan katkelmiksi.

Järjestelmän toiminta perustuu eri tavalla merkattujen XML- lähdedokumenttien sisältömallien yhdenmukaistamiseen. Lähdedokumentit ja niiden mahdollisesti saatavilla olevat rakennemäärittelyt analysoidaan ja mieli- valtaisesti määriteltyjen rakennemäärittelysten elementtitunnistetyypit luokitel- laan Heikkisen (2000) esittelemän yleisen rakennemäärittelyksen, yleisen DTD:n (generic DTD) mukaisiin yleisiin luokkiin. *Yleiset luokat* kuvaavat digitaalisten dokumenttien tyypillisiä loogisia rakenteita. Koostamisen tulosedokumentit ovat myös lopulta yleisen DTD:n mukaisia, sillä valitut dokumenttikatkelmat koos- tetaan muuntamalla ne automaattisesti yhdeksi yleisen DTD:n mukaiseksi do- kumentiksi.

Heikkisen (2000) laatima lähdedokumenttien rakenteiden yhdenmukaistaminen yleisen DTD:n mukaisiksi luokiksi ei perustu elementtitunnisteiden nimiin, vaan toimenpidettä varten laadittu algoritmi kokoaa tietämystä elementtitun- nisteiden välisistä suhteista ja elementtitunnisteiden keskimääräisistä tekstipi- tuuksista dokumenteissa. Täten voidaan tunnistaa ja luokitella dokumenttien loogisia kokonaisuuksia. Esimerkiksi lyhyt tekstinpätkä luokitellaan yleiseen luokkaan *String*, kun taas pidempi kappale luokitellaan yleiseen luokkaan *Para*.

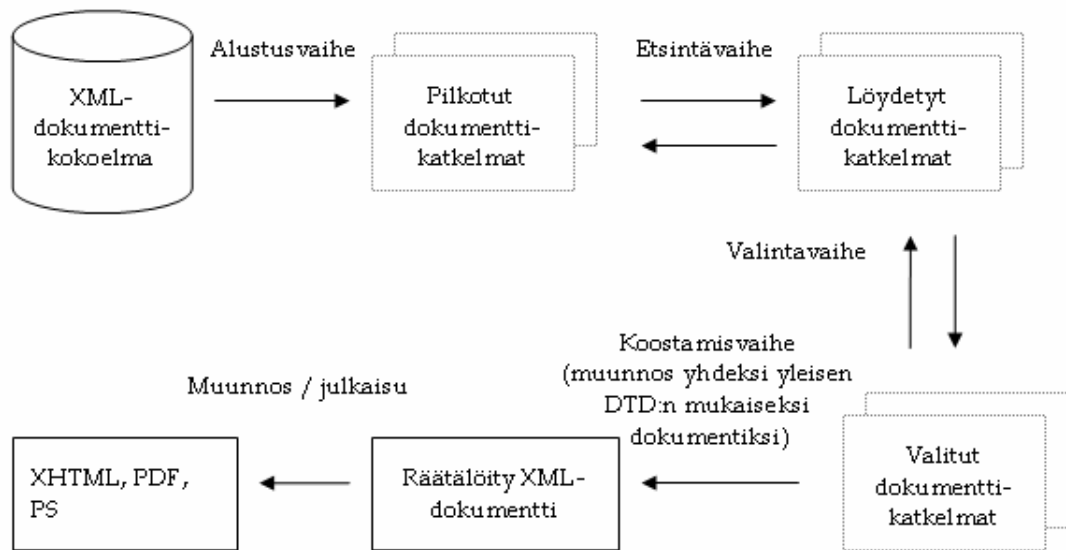


Luokittelu on välttämätöntä, mikäli lähdemateriaali on heterogeenistä. Lähdedokumenttien elementtitunnisterakenteet voivat olla samankaltaisia, mutta eritavalla nimettyjä. Esimerkkinä voidaan käyttää vaikkapa tekstikappaleen määrittävää elementtitunnistetta, joka voi yhdessä dokumentissa olla nimeltään "kappale" ja toisessa "paragraph". Tällaiset elementtitunnisteet kuuluvat tällöin samaan yleiseen luokkaan nimeltä *Para*.

Malli pyrkii olemaan mahdollisimman yleiskäyttöinen yhdenmukaistamalla käytettävän lähdemateriaalin yleisen DTD:n mukaiseksi. Täten mallin voidaan katsoa olevan sopiva esimerkki tähän tutkielmaan.

Kuviossa 4 esitetään mallin viisi vaihetta, jotka ovat:

1. alustusvaihe, jossa analysoidaan ja esikäsitellään dokumenttikokoelmaa,
2. relevanttien dokumenttikatkelmien etsintävaihe,
3. löydettyjen dokumenttikatkelmien valintavaihe,
4. tulodokumentin koostamisvaihe sekä
5. julkaisuvaihe.



Kuvio 4. Dokumenttien koostamismalli (Heikkinen 2000, 48)

Kuviossa 4 on kuvattu pääpiirteittäin Heikkisen (2000, 48) väitöskirjassa esittelemä XML-dokumenttien koostamismalli. Malli lähtee liikkeelle kuviossa 4 olevasta lieriöstä, joka kuvaa XML-dokumenttikokoelmaa. Nuolet kuvaavat dokumenttikatkelmien kulkusuuntaa koostamisprosessissa. Taaksepäin prosessissa osoittavat nuolet kuvaavat dokumenttikatkelmien iteratiivista etsintä- ja valintaprosessia. Katkoviivalla kuvatut laatikot ovat dokumenttikatkelmien tiloja kunkin vaiheen jälkeen. Kokonaisella viivalla kuvatut laatikot ovat koostamisen lopputuloksia tai koosteesta muunnettuja julkaisuja.

Seuraavissa alaluvuissa kutakin Heikkisen (2000) esittelemän mallin vaihetta kuvataan tarkemmin.

### 3.2.1 Alustusvaihe

Ennen varsinaista koostamista on alustusvaihe. Alustusvaiheessa järjestelmä tekee kolme automaattista toimenpidettä. Ensiksi dokumenttikokoelmassa olevien XML-lähdedokumenttien elementtitunnisteet kartoitetaan ja luokitellaan

yleisiin luokkiin, jonka jälkeen sisällöt pilkotaan katkelmiksi eli itsenäisiksi lähdedokumentin osiksi. Tämän jälkeen katkelmat vielä ryhmitellään osajoukkoihin niiden sisältöjen mukaan.

Alustusvaiheessa lähdedokumentteja analysoidaan ja esikäsitellään varsinaista koostamista varten. Mikäli lähdedokumentit ovat rakenteeltaan erilaisia, eli niissä on käytetty erinimisiä elementtitunnisteita merkkamaan samankaltaista sisältöä, ne luokitellaan. Luokittelu tapahtuu elementtitunnistetyyppien mukaan siten, että samantyyppiset ja samankaltaista sisältöä sisältävät elementtitunnisteet kuuluvat samaan yleiseen luokkaan.

Seuraavaksi lähdedokumentit pilkotaan *katkelmiksi*, jotka ovat mahdollisimman yhtenäisiä ja itsenäisiä dokumenttien osia. Katkelma voi koostua esimerkiksi lähdedokumentista kootuista luvuista, aliluvuista tai kappaleista, jotka käsittelevät samaa asiaa. Luvut tai aliluvut ovat kuitenkin usein isoja kokonaisuuksia, jotka saattavat sisältää myös erillisiä asiakokonaisuuksia. Ajatuksena on erotella nämä erilliset asiakokonaisuudet, ja sisällyttää kuhunkin katkelmaan yksi asiakokonaisuus.

Alustusvaiheen viimeisessä osassa dokumenttikatkelmat ryhmitellään kokoelmiksi, joissa kussakin on mahdollisimman samanlaisia ja samansisältöisiä dokumenttikatkelmia. Ajatuksena ryhmittelyssä on se, että yhden kokoelman sisältämät katkelmat ovat keskenään mahdollisimman samanlaisia, mutta mahdollisimman erilaisia muiden kokoelmien sisältämien katkelmien kanssa. Mikäli katkelmat sisältävät paljon samoja sanoja, ne kuuluvat samaan ryhmään. Näin voidaan tunnistaa jotakin tiettyä asiaa käsittelevät katkelmat. Katkelmien ryhmittely kokoelmiin mahdollistaa katkelmakokoelmien selaamisen käyttöliittymän välityksellä kohtuullisenkokoisina paloina, joiden sisällöt ovat mahdollisimman yhtenäisiä.

### 3.2.2 Dokumenttikatkelmien etsintä

Ryhmittelyn jälkeen käyttäjä etsii järjestelmän käyttöliittymän avulla kiinnostavia dokumenttikatkelmia katkelmakokoelmien joukosta, jotka saattaisivat olla relevantteja halutun lopputuloksen kannalta. Hakutapoja on neljä: ensimmäisessä käytetään tekstihakua ja hakuehtoja, jotka kohdistuvat merkattujen dokumenttikatkelmien sisältämiin teksteihin, rakenteisiin ja attribuuttien arvoihin. Toinen tapa perustuu ryhmiteltyjen dokumenttikatkelmien selailuun, jossa dokumenttikokoelman samankaltaiset sisällöt on ryhmitelty selailtaviksi kokonaisuuksiksi. Kolmas tapa etsiä dokumenttikatkelmia on navigoida manuaalisesti hierarkkisten katkelmien rakenteissa. Tätä vaihtoehtoa kannattaa kuitenkin käyttää vasta sitten, kun kaksi ensin mainittua etsintätapaa on jo suoritettu. Neljäs tapa on antaa järjestelmän etsiä jo valittuja katkelmia vastaavia osumia dokumenttikokoelmasta.

### 3.2.3 Löydettyjen dokumenttikatkelmien valinta

Kun käyttäjä on suorittanut edellä mainitun dokumenttikatkelmien etsinnän, hänelle sopivat dokumenttikatkelmat voidaan valita jatkokäsittelyyn eli tulosedokumentin koostamiseen. Kun joukko haluttuja katkelmia on löytynyt, käyttäjä valitsee ne tulosedokumentin koostamista varten käyttöliittymän avulla. Löydetty dokumenttikatkelmat esitetään listana, josta käyttäjä voi valita tai hylätä katkelmia tai täydentää niitä. Katkelmien täydentäminen tarkoittaa esimerkiksi sellaista tilannetta, jolloin johonkin sisältöön halutaan liittää lisää kontekstietoa. Tällöin voidaan tutkia, löytyykö katkelman läheisyydestä - useimmiten katkelmaa edeltävistä rakenteista - esimerkiksi johdanto-kappaletta kuvaava elementtitunniste tai rakennekokonaisuus, josta voitaisiin liittää katkelman yhteyteen kontekstisidonnaista tietoa.

### 3.2.4 Tulodokumentin koostaminen ja erilaiset julkaisut

Edellisissä vaiheissa iteratiivisen dokumenttikatkelmien haku- ja valintaprosessin seurauksena kootut katkelmat muunnetaan järjestelmässä automaattisesti yhdenmukaiseen, yleisen DTD:n mukaiseen rakenteeseen. Tuloksena syntynytä XML-dokumenttia voidaan tämän jälkeen käsitellä ohjelmallisesti. Kyseessä on räätälöity tulodokumentti, joka on Heikkisen (2000) esittelemän yleisen DTD:n mukainen. Tulodokumenttia voidaan tässä vaiheessa vielä muokata, ja sen rakennetta sekä sisältöä voidaan tarvittaessa tarkentaa tehokkaampaa jatkokäsittelyä varten. Dokumentista voidaan tämän jälkeen luoda erilaisia julkaisuja vaikkapa XHTML- tai PDF-muotoon. Julkaisut voidaan toteuttaa esimerkiksi XSL-tyylikielen avulla.

Koostamista voidaan havainnollistaa esimerkillä, jossa koostetaan tulodokumentti samassa dokumenttikokoelmassa olevista XML-dokumenteista, jotka sisältävät tietoa vaikkapa jostakin säveltäjästä. Oletetaan, että samaa säveltäjää koskevaa tietoa löytyy kahdesta erillisestä XML-dokumentista, jotka halutaan yhdistää tulodokumenttiin. Tällaisia tietoja voivat olla esimerkiksi yhdessä XML-dokumentissa sijaitsevat säveltäjän yleiset tiedot, kuten nimi-, historia- ja julkaisutiedot. Toinen XML-dokumentti puolestaan voisi sisältää tarkempia tietoja kustakin julkaistusta sävellyksestä, kuten mahdolliset lyriikat tai julkaisijan. Nämä tiedot halutaan yhdistää tulodokumenttiin säveltäjän yleisten tietojen lisäksi.

Edellä esitellyn koostamismallin mukaisesti heterogeenisesti merkattujen XML-lähdedokumenttien elementtitunnisteet luokitellaan ensin yleisen DTD:n mukaisiksi ja pilkotaan katkelmiksi, jotka edelleen ryhmitellään samansisältöisiksi katkelmakokoelmiksi. Täten käyttäjä voi valita käyttöliittymän välityksellä tietyn säveltäjän tiedoista koostuvia katkelmia tulodokumenttiin. Tämä tapahtuu esimerkiksi siten, että käyttäjä ensin etsii katkelmia, joissa säveltäjän nimi esiintyy ja lisää haluamansa katkelmat tulodokumenttiin.

### 3.3 Mallin arviointi

XML ja sen liitännäisstandardit tarjoavat tehokkaat työkalut dokumenttien säilytysmuodoksi ja monikanavajulkaisun toteuttamiseksi dokumenttisisältöjen koostamisessa, hallinnassa ja uudelleenkäytössä.

Heikkisen (2000) esittelemän mallin eduksi voidaan mainita tapa, jolla lähdedokumentit päivitetään: mikäli koostettuja tulosedokumentteja muokataan, muokkaukset eivät päivity lähdedokumentteihin. Lähdedokumentit on päivitettävä erikseen, jolloin tulevaisuudessa saadaan ajantasaisia koostedokumentteja. Tämän johdosta dokumenttikokoelmaan ei siirrettä yleisen DTD:n mukaisia muunnettuja dokumentteja, ja näin lähdedokumenttien rakenne pysyy johdonmukaisena sekä päivitykset hoidetaan selkeällä tavalla yhdensuuntaisesti.

Monikanavajulkaisun kannalta malli on myös hyvä. Samoja lähdedokumentteja on mahdollista käyttää hyväksi erimuotoisten julkaisujen tuottamisessa, joka on monikanavajulkaisun perusajatus (Lehtonen 2001).

Heikkisen (2000) esittelemää mallia voidaan soveltaa monenlaisissa koostamistapauksissa, joissa lähdemateriaali on XML-muodossa. Mallin toiminta perustuu lähdemateriaalin analysointiin ja dokumenttikatkelmien muuntamiseen yleisen DTD:n mukaisiksi. Täten mallia voidaan soveltaa esimerkiksi yhden organisaation sisällä erilaisissa sisällönhallintatilanteissa tai vaikkapa laajemmin esimerkiksi tietosisältöjen koostamiseen erilaisista maantieteellisesti hajautuneista digitaalisista dokumenttikokoelmista ja tietovarastoista Internetin välityksellä. Tällainen tilanne voisi tulevaisuudessa olla vaikkapa oppimateriaalin koostaminen virtuaalista tutkintoa varten eri oppilaitosten digitaalisista tietovarastoista. Toisaalta voidaan kysyä, onko Heikkisen (2000) esittelemä yleinen DTD aina sopiva kaikenlaisissa koostamistilanteissa? Joissakin tapauksissa yleisen DTD:n määrittämä merkkaustapa ei välttämättä täytä kaikkia tarpeita.

Malli hyödyntää tehokkaasti XML-tekniikan tarjoaman mahdollisuuden jakaa sisältö pieniin merkattuihin osiin eli katkelmiin, joita voidaan uudelleen käyttää ohjelmallisesti. Lisäksi malli sisältää käyttöliittymän, jolla katkelmia voidaan etsiä ja poimia käyttöön. Mallin tavasta pilkkoa lähdemateriaali katkelmiksi herää kuitenkin kysymys: voidaanko olla täysin varmoja siitä, että järjestelmä löytää varmasti juuri oikeat tietosisällöt, eikä jätä jotakin tärkeää tietoa pois, tai ettei katkelmaan pääse vahingossa epärelevanttia tietoa? Tällainen problematiikka korostuu etenkin silloin, kun lähdemateriaalia on paljon ja ehkä jopa eri kielisenä (Lehtonen, Petit, Heinonen, & Lindén 2002). Käyttäjän mukanaolo koostamisprosessissa toki auttaa relevantin tulospöytäkirjan luonnissa, mutta hänkään ei välttämättä huomaa kaikkia puutteita. Lisäksi on vaarana, ettei käyttäjä ehkä osaa antaa oikeita parametreja koostamisesta huolehtivalle järjestelmälle halutun tulospöytäkirjan koostamiseksi. Etenkin keskivertokäyttäjällä on havaittu olevan vaikeuksia parametrien syöttämisessä oikeassa muodossa järjestelmälle (Lehtonen, Petit, Heinonen, & Lindén 2002). Järjestelmän käyttöliittymän käytettävyydellä onkin täten erittäin tärkeä osa koostamisprosessissa.

Myös tietojen tallennus nostaa esiin kysymyksiä. Heikkisen (2000) esittelemä malli ei varsinaisesti ota kantaa katkelmien tallennukseen tai yleensäkkään dokumenttien hallintaan koostamisprosessissa. Mikäli lähdemateriaali päivittyy ja uudet katkelmakokoelmat luodaan, mihin aiemmin muodostetut katkelmakokoelmat tallentuvat, vai tallentuvatko ne ollenkaan? Jonkinlainen versiointi tai vanhan tilanteen palauttaminen olisi epäilemättä tarpeellinen ominaisuus, mikäli vanhoihin katkelmiin haluttaisiin palata myöhemmin. Kaikkien mahdollisten versioiden tallennus puolestaan vaatisi paljon levytilaa, ja lisäksi tarvittaisiin myös mahdollisesti erillinen dokumenttien hallinta -järjestelmä huolehtimaan eri versioita.

### 3.4 Muita koostamismalleja

Heikkisen (2000) käyttämä käsite dokumenttien koostaminen (document assembly) ei suinkaan ole ainoa yleisessä käytössä oleva käsite sisältöjen uudelleenkäytöstä. Sisältöjen uudelleenkäyttö (content reuse) on yleinen termi varsinkin sisällönhallinnan alueella. Esimerkiksi Boiko (2002) on esittänyt sisältöjen yhdistelymallin (content syndication), joka tarkoittaa käytännössä samaa asiaa. Myös Weitzman (2004) puhuu sisältöjen uudelleenkäytöstä eikä dokumenttien koostamisesta, vaikka hänen esittelemänsä tapaus eräästä sisällönhallintajärjestelmästä on samankaltainen prosessiltaan kuin Heikkisen (2000) esittelemä malli. Sekä Heikkisen että Weitzmanin (2004) mallissa on pohjana XML-muotoisten dokumenttikatkelmien muodostaminen ja uudelleenkäyttö erityyppisissä julkaisuissa. Dokumenttien koostaminen ei täten ole yleisesti maailmalla käytössä oleva termi. Käsitteistön yhtenäistäminen olisi hyödyllistä kyseisellä kohdealueella, sillä esimerkiksi haettaessa alan tutkimuskirjallisuutta saattaa monta relevanttia lähdettä jäädä tutkimatta, mikäli dokumenttien koostamiseen liittyvä haku on tehty vain yhdellä monesta mahdollisesta hakusanasta (Honkaranta & Tyrväinen 2005).

Hiukan kauempaa haettu esimerkki tiedon uudelleenkäytöstä löytyy EAI:n puolelta. *EAI:n* (Enterprise Application Integration) eli yrityksen sovellusten integroinnin keskeinen ongelmakenttä on tiedon vaihto eri-ikäisten ja eri teknologioilla toteutettujen sovellusten välillä. Tästä syystä järjestelmien välillä liikkuva data on heterogeenistä ja vaatii usein sovelluskohtaista muuntamistyötä. Tässäkin tapauksessa tietoa eli dataa uudelleenkäytetään, joskin erilaisella periaatteella kuin dokumenttien koostamisessa. XML:n avulla datakonversioiden määrää tiedonsiirrossa sovellusten välillä voidaan vähentää. Kun sovellukset päivitetään ymmärtämään XML-formaattia, voidaan tiedonsiirto sekä tiedon jakaminen järjestelmien ja sovellusten välillä hoitaa yhtenäisellä tavalla. (Antila & Pohjonen 2001)



Käytännössä kyseessä ei aina ole yhtä selkeä dokumenttien koostamistilanne kuin Heikkisen (2000) esittelemä malli. Lähdedokumentit eivät välttämättä ole rakenteisessa muodossa ja saattavat tulla erilaisista järjestelmistä, joiden erilaiset tietomallit saattavat aiheuttaa ongelmia koostamisprosessissa (McIntosh 2000).

Esimerkiksi McIntoshin (2000) mallissa pääperiaatteena on koostaa verkkojulkaisuja erilaisista sisältölähteistä, kuten tietokannoista tai erilaisista heterogeenisistä dokumenttikokoelmista. Tällaisia kokoelmia voivat olla esimerkiksi erilaiset artikkeli-, dokumentti- tai mainosmateriaalikokoelmat erilaisissa organisaatioissa. Useaa erityyppistä sisältökokoelmaa käyttävässä dokumenttien koostamismallissa lähdemateriaali ei aina ole rakenteisessa muodossa, joka luo erityisiä haasteita kyseisten sisältökokoelmien ylläpidolle ja järjestelylle. Tilanne, jossa kaikki julkaistavaksi tarvittava tieto ei ole valmiiksi rakenteisessa muodossa, saattaa olla monien organisaatioiden ongelma.

Heikkisen (2000) esittelemän XML-dokumenttien koostamismallin edut verrattuna McIntoshin (2000) malliin ovat selkeät, sillä lähdedokumentit ovat XML-muodossa, jolloin dokumenttien käsittely helpottuu. Tällainen ideaalitalanne ei kuitenkaan aina ole triviaali asia erilaisissa koostamis-tilanteissa, kuten esimerkiksi McIntoshin (2000) malli osoittaa. Organisaatioiden tulisikin suunnitella sisältöjenhallintansa yhdenmukaiseksi niiltä osin, mitä sisältöjä halutaan uudelleenkäyttää ja yhtenäistää näiden sisältöjen tallennusmuodot. Tällä tavoin sisältöjen uudelleenkäyttö helpottuu ja yhdenmukaistuu, kun käytössä on yksi yhtenäinen lähdeformaatti. XML on ominaisuuksiensa ja yleistyneisyytensä puolesta erittäin varteenotettava teknologiavaihtoehto ongelman ratkaisuksi.

Heikkisen (2000) esittelemä koostamismalli ei suinkaan ole ainoa tapa koostaa XML-dokumentteja. Malli on kuitenkin esitetty hyväksytyssä väitöskirjassa, ja sitä on myös testattu käytännössä prototyyppijärjestelmän avulla lupaavin tuloksin (Heikkinen 2000). Muitakin työkaluja XML-dokumenttien koostamiseen

on olemassa, mutta ne ovat usein vain yhden organisaation tarpeisiin räätälöityjä järjestelmiä (Lehtonen 2001).

## 4 TAPAUSTUTKIMUKSEN KOHDEALUE

Tässä luvussa esitellään tutkielman tapaustutkimuksen tausta, lähtökohdat ja kohdeorganisaatio. Tapaustutkimus käsittelee Jyväskylän yliopiston informaatioteknologian tiedekunnan opinto-oppaan laadintaprosessia ja julkaisujärjestelmää, joita kehitettiin tiedekunnassa kahdessa eri projektissa. Järjestelmän käyttöönottoa testattiin lukuvuoden 2005-2006 opinto-oppaan laadinnassa. Luvussa esitellään opinto-oppaan kehittämishankkeeseen johtaneita seikkoja sekä itse kehittämishanke, joka toteutettiin kahtena osaprojektina vuosien 2004-2005 aikana. Luvussa käydään myös lyhyesti läpi mitä projektityö informaatioteknologian alalla tarkoittaa, ja kuinka projektityöskentelyä opetetaan Jyväskylän yliopiston informaatioteknologian tiedekunnassa tietojenkäsittelytieteiden laitoksella.

### 4.1 Kohdeorganisaatio

Vuonna 1998 perustettu Jyväskylän yliopiston informaatioteknologian tiedekunta (jatkossa IT-tiedekunta) koostuu kahdesta ainelaitoksesta joita ovat tietojenkäsittelytieteiden laitos (TKTL) ja tietotekniikan laitos (TTL). Tiedekunnan tieteenalaja ovat kauppatieteellinen sekä luonnontieteellinen ala. Lisäksi tiedekuntaan kuuluu erillislaitoksena tietotekniikan tutkimusinstituutti (TITU), joka harjoittaa tutkimus- ja kehittämistoimintaa sekä yhteiskunnallista palvelutehtävää. Vuonna 2004 tiedekunnassa opiskeli noin 1800 maisteriopiskelijaa ja 180 jatko-opiskelijaa. Tiedekunnan vuosittainen opiskelijoiden sisäänottomäärä on noin 250 opiskelijaa. Tiedekunnan hallintoa hoitavat tiedekunta- ja laitosneuvostot, sekä niiden alaisuudessa työskentelevä hallintohenkilöstö. Ainelaitosten informaatioteknologian alan (jatkossa IT-alan) opetus on osin yhteistä. Tiedekunnan erityisiä vahvuuksia ovat ohjelmoinnin laadukas perusopetus, laitosten

opiskelijaprojektit, tieteellinen laskenta, aktiivinen maisteriklinikkatoiminta sekä tuloksekas tohtorikoulutus. (Heikkilä, Hiltunen, Hämäläinen ym. 2005)

## 4.2 Projektityöskentelyn opetus tietojenkäsittelytieteiden laitoksella

Projektityöskentelyn katsotaan olevan tärkeä osa IT-alan opetusta. Opiskelijat oppivat soveltamaan teoriaa käytännössä, tuottamaan ja noudattamaan projektisuunnitelmia, viemään projektin läpi aikataulussa ja toimimaan asiakkaiden kanssa. Projektityöskentely kehittää opiskelijoiden ammattitaitoa IT-alan projekteissa työskentelemisessä ja lisäksi opiskelija oppii sosiaalisia sekä ryhmätyöskentelyn taitoja. (Vartiainen 2005)

Jyväskylän yliopiston IT-tiedekunnan molemmat ainelaitokset järjestävät projektiopetusta, johon kuuluu IT-alan käytännön harjoittelua. Tietojenkäsittelytieteiden laitoksella järjestetään vuosittain opintojakso, jolla opinnoissaan hieman pidemmälle ehtineet kyseisen ainelaitoksen opiskelijat harjoittelevat IT-alan projektityöskentelyä oikeiden asiakkaiden ja toimeksiantojen kanssa. Kyseessä on tämän tutkielman kirjoitushetkellä 15 opintopisteen laajuinen syventäviin tietojärjestelmätieteen opintoihin kuuluva Projektin johtaminen -opintojakso. Opintojaksolla on aikaisemmin ollut useampiakin nimiä, kuten työprojekti tai kehittämisprojekti. Opintojaksolla suunnitellaan ja toteutetaan IT-alan kehittämissankkeita opiskelijaryhmissä, sekä keskitytään etenkin projektin johtamisen harjoitteluun. Jokaisella ryhmällä on oma ohjaaja, joka seuraa ja opastaa ryhmän työskentelyä ja oppimista koko projektin ajan. Kukin opintojaksolle osallistuva opiskelija käyttää opintojaksoon noin 400 tuntia, joista 275 tuntia on varattu toimeksiannolle, ja loput 125 tuntia opintojakson opetukseen kuten esimerkiksi luentoihin ja ohjaukseen. (Vartiainen 2005)

Vartiaisen (2005) mukaan *Projektin johtaminen -opintojakson* päätavoitteita ovat:

- teorian ja käytännön yhdistäminen IT-alan kehittämistyössä,
- antaa opiskelijoille kokonaisvaltainen ja realistinen näkemys tietojärjestelmäsuunnittelusta ja
- antaa opiskelijoille henkilökohtaisia kokemuksia projektityöstä.

Opiskelijat refleктоivat projektin aikana jatkuvasti oppimistaan projektityötaidoista. Refleктоitavia asioita ovat ryhmä- ja projektityö, IT-alan menetelmien käyttö, projektin suunnittelu ja seuranta, viestintä ja vuorovaikutus, opiskelijan oma kontribuutio sekä orientoituminen projektiin. (Vartiainen 2005)

### **4.3 Opinto-opas -dokumentti**

IT-tiedekunta julkaisee vuosittain opinto-oppaan, jota käyttävät muun muassa tiedekunnan omat uudet ja vanhat opiskelijat, jatko-opiskelijat, sivuaineopiskelijat, potentiaaliset uudet opiskelijat, tiedekunnan oma opetus- ja hallintohenkilöstö sekä yliopiston muu hallintohenkilöstö. Opinto-oppaan sisällössä keskeisellä sijalla ovat tiedekunnan tutkintovaatimukset ja niistä tiedottaminen. Oppaasta on julkaistu painettu versio ja lisäksi tiedekunnan WWW-sivuilla on ollut oppaan HTML-versio.

Opinto-oppaan suhteellisen laaja sisältö kattaa etenkin opiskelijan näkökulmasta tarpeellista tietoa liittyen tiedekunnassa opiskeluun, tiedekunnan henkilöstöön, tiedekunnan tarjoamiin opintojaksoihin, opinto-ohjaukseen, muualla suoritettujen opintojen korvaavuuksiin ja muihin yliopisto-opiskeluun liittyviin asioihin.

Opinto-oppaan sisällön looginen rakenne koostuu seuraavista osista:

- etu- ja takakansi,

- sisäkansi (nimiösivu),
- oppaan tekijä- ja julkaisutiedot,
- yksi tai useampi numeroimaton esiluku,
- sisällysluettelo,
- numeroidut tekstiluvut alalukuineen (oppaan varsinainen tekstisisältö) sekä
- liitteet.

Opinto-oppaan etu- ja takakannet voivat sisältää esimerkiksi kansitekstejä, kuvia ja nimiösivu, jossa on tyypillisesti ollut yliopiston ja tiedekunnan nimi sekä oppaan painamisvuosi. Tämän jälkeen on esitelty oppaan julkaisutiedot, kuten oppaan työryhmän henkilöiden nimet julkaisunumerot ja mahdollisesti oppaan painopaikka.

Esiluvut ovat numeroimattomia lukuja, jotka ovat olleet johdantoa oppaalle. Tyypillisesti esilukuna on ollut esimerkiksi tiedekunnan dekaanin tervehdys ja uusille opiskelijoille tarkoitettu ensimmäisen opiskeluviikon ohjelma. Esilukujen jälkeen on oppaan käytön helpottamiseksi tulostettu sisällysluettelo. Sisällysluettelon jälkeen ovat seuranneet oppaan varsinaiset tekstiluvut.

Oppaan varsinainen tekstisisältö muodostuu käytännössä muun muassa yleisistä tiedekunnassa opiskeluun liittyvistä asioista, tutkinnoista, sivuainemahdollisuuksista ja ainelaitoksilla opiskeluun liittyvistä yksityiskohdista. Vuoden 2004-2005 opasta laadittaessa toteutettiin sisältöön liittyvä huomattava uudistus, jonka myötä molempien ainelaitosten opiskeluun ja tutkintoihin liittyvät asiat olivat oppaassa samassa luvussa vierekkäin vertailun helpottamiseksi.

Oppaan sisältömuotoilu ja ulkoasu on koostunut eritasoisista numeroiduista otsikoista, tekstikappaleista, erilaisista tekstimuotoiluista kuten lihavointi tai

kursivointi, taulukoista, erityyppisistä listoista, kuvioista sekä kuvateksteistä ja muutamista alaviitteistä. Sisältö on tuotettu käyttäen tavanomaisia tekstinkäsittelyohjelmien, kuten Microsoft Wordin (Microsoft 2005) ominaisuuksia. Tekstiluvut ovat noudattaneet yhtenäistä rakennetta ja muotoilua.

Tekstilukujen jälkeen oppaan lopussa on tyypillisesti ollut yksi tai useampi liite. Esimerkiksi lukuvuonna 2004-2005 liitteenä oli tiedekunnan henkilötietolistaus, joka sisälsi suuren määrän henkilökunnan yhteystietoja sekä heidän roolejaan ja tehtäviään erilaisissa tiedekunnan toiminnoissa. Nämä tiedot päivittyvät jatkuvasti, eikä esimerkiksi painetun oppaan tiedot ole etenkään henkilöiden tehtävien ja roolien osalta ajan tasalla koko lukuvuotta. Oppaan tietojen päivittämisen automatisoinnille oli täten havaittu tarve.

Yksi laaja liiteosio on sisältänyt tiedekunnan opintojaksojen kuvaukset. Lukuvuoden 2004-2005 oppaaseen kuvaukset oli koostettu manuaalisesti Korppi-opintotietojärjestelmästä (Korppi 2005) saadusta näyttötulosteesta, joka oli havaittu erittäin työlääksi tavaksi. Opintojaksot oli kyseisessä oppaassa jaoteltu järjestävän laitoksen, opintojakson tason ja luennointiajankohdan mukaan.

Oppaan tallennusmuoto on ollut Microsoft Word -tiedostomuoto, josta on luotu erilaisia kolmannen osapuolen työkaluja käyttäen painoa varten PDF-versio sekä verkkojulkaisua varten HTML-versio.

#### **4.4 Ongelmat opinto-oppaan laadinnassa ja koostamisessa**

Opinto-oppaan laatimisesta vastuussa olevien henkilöiden mukaan oppaan laadintaprosessi on perinteisesti ollut mutkikas ja tietojärjestelmiin sekä ihmisiin jakautuvia sisältölähteitä sekä sisällöntuottajia on paljon. Opinto-oppaan laadintaan on liittynyt sekä teknologisia että teknologiasta riippumattomia ongelmia, jotka on kuvattu tarkemmin seuraavissa alaluvuissa.

#### 4.4.1 Teknologiset ongelmat

Suuri teknologiaan liittyvä ongelma oppaan laadintaprosessissa on ollut se, että lähestulkoon kaikilla sisällöntuottajilla on ollut käytössään omat Word-tekstinkäsittelyohjelman tyyliohjelmat. Tämä on tuottanut hankaluuksia erityisesti koostettaessa opasta useista lähdedokumenteista. Sisällöntuottajien lukuisat eri tyyliohjelmat ovat johtaneet esimerkiksi siihen, että koostettaessa opasta sisällöntuottajien teksteillä, on Word-tiedostoon tullut mukana uusia ja erinimisiä tyyliohjelmia. Nämä lukuisat erilaiset ja erinimiset tyyliohjelmat ovat pahimmassa tapauksessa kaataneet koko Word-ohjelman tiedoston koon suhteettoman kasvamisen vuoksi. Kyseinen ongelma on aiheuttanut erittäin turhauttavaa työtä oppaan koostamisessa etenkin opasversioita päivitettäessä. Päivitettäessä opasta koostamalla uudet versiot tuotetuista sisällöistä erilaisiin tyyliohjelmiin ja muotoiluun liittyvät ongelmat ovat aina toistuneet. Sisällöntuottajien lukumäärä ja heterogeenisyys sekä kiireiset aikataulut ovat vaikeuttaneet yhteisten teknologisten käytänteiden luomista oppaan laatimisessa.

Oppaan sisällöntuottaminen itsessään on ollut varsin manuaalinen prosessi, vaikka osa oppaan sisällöstä on ajan mittaan tullut saataville tietojärjestelmistä. Tällaisia tietoja ovat esimerkiksi IT-tiedekunnan opintojaksojen kuvaukset, jotka esimerkiksi vuoden 2004-2005 opinto-opasta laadittaessa kopioitiin oppaaseen suoraan näytöltä Korppi-opintotietojärjestelmän tulosteesta. Opintojaksokuvausten manuaalinen kopiointi oppaaseen oli havaittu erityisen työlääksi: mikäli opintojaksokuvauksiin oli tehty olennaisia muutoksia Korppiin, oli uudet kuvaukset jouduttu kopioimaan uudelleen manuaalisesti opinto-opaaseen. Lisäksi joissain tapauksissa opintojaksokuvausten poimintaan ei ollut saatu kaikkia tarvittavia opintojaksoja esimerkiksi siksi, että opintojaksosta vastaava henkilö oli merkannut opintojakson järjestämistiedot vahingossa väärin Korppi-järjestelmään. Tällöin opintojaksokuvausten poiminta ja kopiointi jouduttiin toistamaan tai kopioimaan näyttötulosteesta puuttuvien opintojaksojen tekstit liitetekstien väliin. Koska opintojaksokuvaukset on tallennettu Korppi-



järjestelmään, havaittiin, että Korpin ja oppaan välillä olisi hyvä olla tiedonsiirtoyhteys opintojaksokuvausten kopiointia varten. Integraatio Korpin ja oppaan välillä vähentäisi manuaalisen kopioinnin aiheuttamia ongelmia.

Erillisestä tietojärjestelmästä tulevaa tietoa ovat myös oppaaseen liitettävät tiedekunnan henkilötiedot, jotka ovat peräisin Jyväskylän yliopiston henkilökunnan sähköisestä puhelinluettelosta. Samat tiedot ovat nykyisin osittain myös Korppi-järjestelmässä.

Rajapintoja erillisiin järjestelmiin ei kuitenkaan ole ollut ja kaikki tiedot on koostettu oppaaseen manuaalisesti. Tiedonsiirron automatisointi opasta koostettaessa, niiltä osin kuin se on mahdollista, olisi hyödyllistä manuaalisen työn vähentämiseksi.

#### **4.4.2 Teknologiasta riippumattomat ongelmat**

IT-tiedekunnan ainelaitokset tuottavat oppaaseen omat osuutensa ja osa tiedoista tulee muilta tiedekunnilta sekä laitoksilta. Oppaan sisällöntuotanto on ollut melko sirpaloitunutta, joka tarkoittaa sitä, että oppaan sisällön tuottamisessa ei ole osoitettu selkeitä kokonaisuuksia tietyille vastuuhenkilöille. Yksi henkilö on voinut tuottaa sisältöosuuksia eri puolille opasta. Täysin selkeitä vastuualueita sisällöntuotannossa ei ole ollut lukuun ottamatta ainelaitosten osuuksia.

Opinto-oppaan laatimisprosessin ongelmat ovat johtuneet muun muassa puutteellisista ohjeistuksista sekä epätarkasta sisällöntuotannon aikataulutuksesta. Vastuuhenkilöiden puuttuminen prosessin tietyissä vaiheissa on erityisesti hankaloittanut sisällöntuotantoa. Oppaan laatimisesta vastuussa olevien henkilöiden mukaan vastuualueista on ollut jopa ristiriitaisia näkemyksiä (XooZoo 2005a).

Eräs ongelma on ollut myös tuotettujen sisältöjen eli raakatekstiversioiden hyväksyttäminen oppaan lopulliseksi sisällöksi. Aiemmin on havaittu, että tekstejä on hyväksytetty ehkä hieman turhankin monilla tahoilla, josta on usein seurannut turhia viivästyksiä oppaan valmistumiseen. Aiempina vuosina myös tulevien lukuvuosien opetusohjelmien laatimisen viivästyminen on vaikuttanut oppaan valmistumiseen viivästyttämällä sisällöntuotantoa. Lisäksi sisällön jaottelusta ja sisältö-osien tarkoituksenmukaisuudesta käyttäjän kannalta on ollut epävarmuutta.

Sisältöön liittyviä kysymyksiä ovat olleet esimerkiksi:

- onko oppaan sisältämä informaatio riittävän relevanttia,
- puuttuuko oppaasta jotakin olennaista tietoa,
- onko opintoihin liittyvät asiat esitetty riittävän selkeällä tavalla ja
- tarvitaanko opinto-oppaista erilaisia versioita esimerkiksi uusille ja vanhoille opiskelijoille.

#### 4.5 Opinto-oppaan kehittämishanke

Vuosien varrella opinto-opasta koskevia kehittämiskohteita oli ilmennyt paljon. Kaikki ongelmat eivät olleet korjattavissa pelkästään teknologian avulla. Koska kyseessä on joka vuosi toistuva laajahko prosessi, IT-tiedekunta päätti syventyä kehittämiskohteisiin tarkemmin syksyllä 2004 käynnistämällä kehittämishankkeen parantaakseen opinto-oppaan sisältöä, laadintaprosessia sekä oppaan laadintaan ja julkaisuun käytettäviä teknologioita.

Opinto-oppaan kehittämishanke jakautui kahteen osaan: opiskelijaprojektina toteutettuun XooZoo-projektiin ja XooZoo-projektin tuotoksista jatkaneeseen tiedekunnan omaan Xoo-jatko -projektiin. Kehittämishankkeen sisällön ja ta-



#### 4.5.1 XooZoo-projekti

Lukuvuonna 2004-2005 järjestetyn Projektin johtaminen –opintojakson yhtenä toimeksiannota oli IT-tiedekunnan opinto-oppaan hallinnan, sisällön ja rakenteen kehittäminen. Toimeksiannon otti vastaan opiskelijoista koostuva kuusi-henkinen projektiryhmä, jotka olivat suuntautuneet opiskeluissaan erilaisiin IT-alan osa-alueisiin. Projektin nimeksi asetettiin XooZoo. Projektiryhmän lisäksi projektiin osallistui myös muita henkilöitä eri rooleissa. Asiakkaana oli IT-tiedekunnan opintoasiainpäällikkö, teknisinä asiantuntijoina ryhmää opastivat digitaalisen median lehtori tietojenkäsittelytieteiden laitokselta ja ohjelmistotekniikan lehtori tietotekniikan laitokselta. Lisäksi ryhmällä oli opintojakson puolesta ohjaaja, joka opasti ja seurasi projektiryhmän toimintaa projektin hallinnollisissa asioissa.

XooZoo-projektin tarkoituksena oli kartoittaa IT-tiedekunnan opiskelijoiden mielipiteitä opinto-oppaasta, selvittää oppaan laadintaprosessin ongelmakohdat ja esittää niille parannusehdotuksia. Lisäksi, mikä tämän tutkimuksen kannalta on tärkeintä, projektin tehtävänä oli laatia alustavat XML-määrittelyt opinto-oppaan laadintaprosessin teknologiaperustaksi sekä oppaan monikanavajulkaisun tueksi. Projektin käynnistyskokous pidettiin 12.10.2004 ja viimeinen johtoryhmän kokous pidettiin 15.3.2005, jolloin projekti päätettiin. Projektille asetetut tavoitteet saavutettiin pääpiirteittäin. XooZoo-projektin toimeksianto on tämän tutkielman liitteenä (Liite 1).

XooZoo-projektin pääasiallisia tuloksia olivat:

- IT-tiedekunnan opiskelijoiden mielipiteiden kartoitus opinto-oppaan sisällöstä sähköisen kyselyn avulla,
- oppaan laadintaprosessin ja sen ongelmakohtien kuvaus sekä parannusehdotukset,

- alustavat rakennemääritykset, eli DTD:t uuden XML-muotoisen oppaan tuottamista varten sekä
- lukuvuoden 2004-2005 opinto-oppaan malliksi merkkäminen laadituilla rakennemäärityksillä.

Opinto-oppaan sisältöä ja sen käyttöä koskeva raportti perustui syksyllä 2004 järjestettyyn sähköiseen kyselyyn, joka toteutettiin Korppi-järjestelmän avulla. Raportin tarkoituksena oli kartoittaa opiskelijoiden mielipiteitä opinto-oppaan sisällöstä ja oppaan käytöstä. Kysely suunnattiin informaatioteknologian tiedekunnan tietotekniikan ja tietojenkäsittelytieteiden laitosten perus- ja jatko-opiskelijoille. Kyselyt lähetettiin 1550 opiskelijalle ja niihin vastasi yhteensä 284 opiskelijaa. Kysely sisälsi erilaisia kysymyksiä liittyen esimerkiksi opinto-oppaan sisällön olennaisuuteen, asioiden esitystapaan ja asiakokonaisuuksien sijaintiin oppaassa. Erityisesti oppaan rakenteeseen kohdistui kritiikkiä. Osa opiskelijoista ei esimerkiksi pitänyt siitä, että tiedekunnan molempien laitosten asiat oli sisällytetty samaan lukuun. Myös muun muassa oppaan jakeluun liittyviin asioihin otettiin kantaa. (XooZoo 2005b)

Oppaan laadintaprosessin ongelmakohdat ja toimijat selvitettiin kartoittamalla laadintaprosessin vaiheet. Vaiheet kuvattiin RASKE-menetelmän (Salminen 2003) mukaisilla kaavioilla. Tietoa hankittiin myös haastatteluin ja työpajayhteistyöstä. Laadintaprosessista tehtiin oma raportti (XooZoo 2005a). Raportissa esitettiin myös parannusehdotuksia laadintaprosessin selkeyttämiseksi ja toimijoiden vastuualueiden tarkentamiseksi. Parannusehdotukset liittyivät laadintaprosessin aikataulutuksen kehittämiseen, selkeän opasryhmän ja vastuuhenkilöiden asettamiseen vastuuseen oppaan laadinnasta, selkeiden ohjeistuksien korostamiseen oppaan laadinnassa ja dokumenttienhallinnan kehittämiseen.

XML oli valittu oppaan lähdemateriaalin teknologiaksi jo ennen XooZoo-projektin käynnistämistä. Perusteena tälle oli XML-kielen perusajatus erottaa sisältö- ja ulkoasumääritykset toisistaan, jotta vältyttäisiin lukuisilta erilaisilta tyylimäärityksiltä ja niiden tuomilta ongelmilta. Tällaiset ongelmat olivat korostuneet vanhassa oppaan laadintaprosessissa. XML-standardin avoimuus oli perusteena teknologiavalinnalle, sillä sen ansiosta opinto-oppaan laadinnan kehittämistä ei tarvinnut sijoittaa esimerkiksi minkään kaupallisen työkalun vaatimaan alustaan tai teknologiaan. Lisäksi opinto-oppaan lähdedokumentteja tuli olla mahdollista muokata periaatteessa millä tahansa työkalulla, joka XML:n yhteydessä on mahdollista.

XooZoo-projektin aikana laadittiin yhteensä kuusi alustavaa rakennemääritystä eli DTD:tä uutta XML-muotoista opasta varten. DTD-suunnittelussa perehdyttiin oppaan loogiseen rakenteeseen ja sisällöllisiin vaatimuksiin. Suunnittelutyö oli iteroiva prosessi, jonka aikana havaittiin, että DTD oli lähes mahdoton suunnitella kerralla valmiiksi. Uusia vaatimuksia ilmeni koko ajan testauksien aikana. Projektin aikana luodut rakennemääritykset olivat:

1. DTD oppaan julkaisutietoja ja kantta varten (julkaisutiedot.dtd),
2. DTD sisällöntuottajien tuottamille tekstisisällöille (luku.dtd),
3. DTD tekstiosien laatija- ja valmiustietojen syöttöä varten (meta.dtd),
4. DTD tutkintojen normaalia tekstisisältöä tarkempaa merkkausta varten (tutkinnot.dtd),
5. DTD opintojaksokuvauksia varten (opintojaksot.dtd) ja
6. DTD henkilötietojen merkkausta varten (yhteystiedot.dtd).

XooZoo-projektin tuotoksiin kuului myös lukuvuoden 2004-2005 opinto-oppaan sisällön malliksi merkkautaminen laadittujen rakennemääritysten avulla. Merkkautaminen tapahtui kopioimalla vanhan oppaan elektronisesta versiosta

sisältöjä XMLSpy-ohjelmaan (Altova 2005) ja merkkamalla ne sopivin elementti-attribuuttitunnistein loogiseksi kokonaisuudeksi. Jokaisesta vanhan oppaan luvusta tehtiin oma XML-tiedosto tiedonhallinnan helpottamiseksi.

Oppaan sisällöstä merkattiin ainoastaan varsinaiset sisältöluvut ja merkkaamisessa käytettiin hyväksi pääasiassa luku.dtd:tä. Täten esimerkiksi opintojaksojen kuvaukset ja henkilötietojen merkkkaus jäivät keskeneräisiksi. Tutkinnot.dtd-määrittystäkin testattiin tiedekunnan erilaisten tutkintojen tarkempaan merkkaukseen, mutta se havaittiin erittäin keskeneräiseksi ja sen kehittämistyö päätettiin siirtää jatkokehitykseen. Merkkaustyön aikana havaittiin monia lisävaatimuksia rakennemäärittäyksille, jotka kirjattiin ylös testauskommentteihin. Kommentteja hyödynnettiin myöhemmin DTD-määrittysten viimeistelyssä projektin loppupuolella ja Xoo-jatko -projektissa.

XooZoo-projektin aikana ei ehditty luomaan valmista opinto-oppaan julkaisujärjestelmää. Projektin päättyessä keväällä 2005 IT-tiedekunta oli kuitenkin vakuuttunut siitä, että kehittämistyön jatkaminen olisi hyödyllistä. Täten XooZoo-projektin tuotoksia päätettiin kehittää Xoo-jatko -projektissa.

#### **4.5.2 Xoo-jatko -projekti**

XooZoo-projektin jälkeen kehittämistyötä jatkoi tiedekunnan yhteinen työryhmä. Jatkokehittäminen tarkoitti käytännössä IT-tiedekunnan lukuvuoden 2005-2006 opinto-oppaiden tuottamista XooZoo-projektin tuotosten pohjalta, koostamista ja monikanavajulkaisua XML-kielen ja XooZoo-projektissa luotujen DTD:iden avulla sekä opinto-oppaiden uuden julkaisujärjestelmän rakentamista, pilotointia, testausta ja dokumentointia. XooZoo-projektin havaintojen seurauksena tulevan lukuvuoden 2005-2006 opinto-oppaista päätettiin luoda kaksi eri versiota: oma opasversio ensimmäisen vuoden opiskelijoille ja vanhemmille opiskelijoille. Myös englanninkielinen opas tuotettiin, mutta sitä ei muunnettu XML-muotoon ajanpuutteen vuoksi.

Xoo-jatko -projekti käynnistettiin huhtikuussa 2005 ja se päättyi kesäkuussa 2005. Työryhmän hallintoon kuuluivat tiedekunnan opintoasiainpäällikkö ja XooZoo-projektia opastaneet tekniset asiantuntijat. Lisäksi työryhmään kuului tiedekunnassa työskentelevä suunnittelija sekä XooZoo-projektissa työskennellyt osa-aikainen XML-tukihenkilö, joka on myös tämän tutkimuksen kirjoittaja. Lisäksi oppaan sisällön tuottamiseen ja jäsentämiseen osallistuivat tiedekunnan amanuenssit sekä muutamia yksittäisiä sisällöntuottajia tiedekunnan sisältä ja ulkopuolelta.

Xoo-jatko -projektin aikana oppaan lähdemateriaali tuotettiin XML-muotoon noudattaen opinto-oppaalle luotuja rakennemäärittäjä. XML-muotoon saattaminen tapahtui osin automaattisesti, osin manuaalisesti. Xoo-jatko -projektin kuluessa sisältöä ja tekstimuotoiluja koskevat uudet vaatimukset tarkentuivat, joten XooZoo-projektin aikana laaditut rakennemäärittäjät kokivat lieviä muutoksia.

Vanhan oppaan laatimiseen liittyneitä teknologisia ongelmia ratkaistiin automatisoimalla Korppi-järjestelmästä saatavien opintojaksokuvausten siirto opinto-oppaaseen. Henkilötietojen oppaaseen kopioimisen automatisointia ei toteutettu, vaan ne muunnettiin manuaalisesti XML-muotoon opasta varten.

Dokumenttien hallinta perustui verkkopalvelimelle luotuun hakemistorakenteeseen, jossa molempien opasversioiden tiedostoilla oli omat hakemistonsa. Varsinaista dokumenttienhallintajärjestelmää ei ollut käytössä.

Oppaan julkaisu perustui verkkopalvelimelle toteutettuun järjestelmään, joka mahdollisti opinto-oppaiden reaaliaikaisen tietojen päivittymisen. Tämä tarkoitti käytännössä sitä, että lähdedokumenttien päivittyessä opas-versiosta luotiin pyydettyä uusi julkaistu kooste sekä PDF- että XHTML-muodoissa. Tämä helpotti oppaan laadintaa siten, että sisältöä muokannut henkilö saattoi välittömästi nähdä miltä tehdyt muutokset näyttivät valmiissa oppaassa. Jatko-



projektin lopulla julkaisujärjestelmälle kehitettiin myös yksinkertainen WWW-selaimella käytettävä hallintakäyttöliittymän prototyyppi.

Seuraavassa luvussa kuvataan tarkemmin opinto-oppaan kehittämishankkeen myötä tapahtuneita muutoksia oppaan laadintaprosessissa. Kuvauksessa käytetään sisällönhallinnan alueelle kehitettyä viitekehystä (Salminen 2003b).

## 5 TAPAUSTUTKIMUS

Tässä luvussa esitellään ensin dokumenttien hallinnan alueelle kehitetty viitekehys, jonka mukaisesti tapaus kuvataan tarkemmin. Kuvaus perustuu materiaaliin, jota kerättiin aikavälillä lokakuu 2004 – syyskuu 2005 tutkielman laatijan toimesta. Viitekehysten avulla pyritään selventämään opinto-oppaan laadinta- ja koostamisprosessia ennen kehittämishanketta ja sen jälkeen.

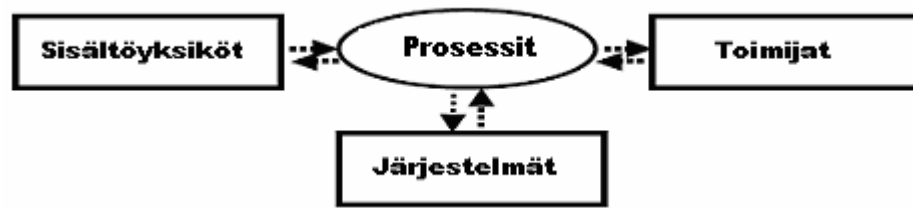
### 5.1 Viitekehys ja tiedonkeruumenetelmät

Opinto-oppaan kehittämishanketta analysoidaan Salmisen, Lyytikäisen & Tiitisen (2000) ja Salmisen (2003a) esittelemän dokumenttien hallinnan mallin avulla. Mallista on laadittu muunnelma, joka soveltuu paremmin sisällönhallinnan alueelle (Salminen 2003b). Jälkimmäistä mallia on käytetty muun muassa Eläketurvakeskuksen (Eläketurvakeskus 2005) sisällönhallinnan kehittämissuunnitelman arvioinnissa ja monissa muissa suomalaisia organisaatioita koskevissa sisällönhallintaan liittyvissä tapauksissa (Honkaranta, Salminen & Peltola, 2005).

Opinto-oppaan kehittämistapauksen arvioinnissa käytetään mallin jälkimmäistä versiota, sillä alkuperäinen malli sitoo viitekehyksessä mallinnettavan informaation dokumenteiksi. Jälkimmäisessä mallissa tallennettu informaatio on määritelty laiveammin sisältöyksiköiksi. Tämä sopii paremmin esimerkkitapauksen arviointiin, sillä kaikki opinto-oppaan lähdemateriaali ei ole puhtaasti dokumentiksi luokiteltavaa tietoa, vaan mukana on esimerkiksi erilaisten järjestelmien ja yksittäisten ohjelmistojen tuottamia sisältökoosteita. Jaottelun avulla havainnollistetaan opinto-oppaan laadintaan liittyviä muutoksia kehittämishankkeen jälkeen.

Tapaustutkimuksen tutkimusmateriaalina on käytetty XooZoo-projektissa ja sen jälkeen toteutetussa Xoo-jatko -projektissa tuotettuja dokumentteja ja raportteja, haastatteluja, jatkoprojektin aikana käytyjä sähköpostikeskusteluita, projekteihin liittyviä palaverimuistioita sekä etenkin tutkijan omaa havainnointia kehittämishankkeen aikana. Tutkielman kirjoittaja osallistui opinto-oppaan kehittämishankkeeseen koko sen keston ajan. XooZoo-projekti ja Xoo-jatko -projekti nähdään arvioinnissa yhtenä kokonaisuutena. Tutkimusmateriaalia ei analysoitu tilastollisten menetelmien avulla, vaan luvun 5 analyysi perustuu kvalitatiiviseen eli laadulliseen pohdintaan viitekehyksen avulla. Näkökulma pyritään pitämään mahdollisimman objektiivisena tutkielman kirjoittajan kehittämishankkeeseen osallistumisesta huolimatta.

Sisällönhallinnan viitekehys esitellään graafisesti kuviossa 6. Viitekehyksessä on kahdentyyppisiä kokonaisuuksia: prosesseja sekä resursseja. *Prosessit* ovat toimintoja, jotka koostuvat erilaisista aktiviteeteista joita yksi tai useampi toimija suorittaa. Voidaan ajatella, että organisaatiossa on prosesseja, jotka koostuvat aktiviteeteista. Esimerkkinä prosessista voi olla vaikkapa opinto-oppaan koostaminen, johon liittyy tiettyjä aktiviteetteja, kuten sisällöntuotanto ja lähdedokumenttien yhdistely opinto-oppaaksi. *Resurssit* on jaettu kolmeen eri tyyppiin riippuen niiden suhteesta aktiviteetteihin. Resursseja ovat toimijat, sisältöyksiköt ja järjestelmät. *Toimijat* ovat aktiviteettien suorittajia. Toimija voi olla joko organisaatio tai henkilö. Toimija voi olla myös ohjelmistoagentti, joka edustaa henkilöä. *Sisältöyksiköt* ovat tallennettua tietoa, joka tuotetaan tai jota käytetään aktiviteeteissa. Suuri osa sisältöyksiköistä on dokumentteja. *Järjestelmät* kuvaavat laitteistoja ja ohjelmistoja, joita käytetään hyväksi aktiviteettien suorittamiseen. (Salminen 2003b)



Kuvio 6. Sisällönhallintaympäristön osa-alueet

Katkoviivoilla kuvatut nuolet kuvaavat tietovirtoja prosessien ja resurssien välillä. Resursseja voidaan luonnehtia tiedon säilytyspaikoiksi, joihin eri prosessien aktiviteeteissa tuotettu informaatio voidaan tallentaa, ja joista aktiviteetit voivat hakea informaatiota. Aktiviteettien tarvitsema ja tuottama informaatio on säilötyinä dokumentteihin ja erilaisiin muihin sisältöyksiköihin, järjestelmiin, organisaation toimintatapoihin ja organisaation työntekijöiden aivoihin sekä kokemuksiin. Informaatio on täten mitä tahansa tietoa, joka voidaan tallentaa johonkin säilytyspaikkaan ja käyttää hyväksi jossakin aktiviteetissa tai jonka jokin aktiviteetti voi tuottaa ja tallentaa jonkin toisen aktiviteetin käytettäväksi myöhemmin, mahdollisesti jossakin muussa ympäristössä. Suhteessa sisältöyksiköihin, toimijoita voidaan kutsua käyttäjiksi tai tiedontuottajiksi (Salminen 2003b).

## 5.2 Opinto-oppaan laatiminen ennen kehittämishanketta

Seuraavissa alaluvuissa esitellään opinto-oppaan laatiminen jaoteltuna edellä esiteltyihin osa-alueisiin ennen opinto-oppaan kehittämishanketta. Tarkoituksena on kuvata mahdollisimman tarkasti alkutilassa vallinnut tilanne toimijoiden, prosessien, sisältöyksiköiden ja järjestelmien osalta.

### 5.2.1 Toimijat

Opinto-oppaan laadintaan osallistuneet toimijat muodostuivat IT-tiedekunnan sisäisistä ja ulkoisista toimijoista. Sisäiset toimijat koostuivat IT-tiedekuntaa koskevien tekstisisältöjen tuottajista, koostajasta ja hyväksyjistä. Tärkeimpiä sisäisiä toimijoita olivat ainelaitosten amanuenssit ja tiedekunnan opintoasiainpäällikkö. Etenkin tutkintorakenteisiin liittyvien tekstisisältöjen hyväksyjinä toimivat laitostasolla laitosneuvostot ja tiedekuntatasolla tiedekuntaneuvosto. Opinto-oppaan laatimista varten oli nimitetty työryhmä, joka oli päävastuussa oppaan laadinnasta. Täysin tyhjentäviä sisällöntuotannon vastuualueita ei kuitenkaan ollut työryhmän sisällä määritelty ja niistä oli osittain erilaisia näkemyksiä. Pääasiallinen vastuu opinto-oppaan laatimisesta oli tiedekunnan opintoasiainpäälliköllä. Sisäisiin toimijoihin kuului myös tekninen asiantuntija, joka hoiti oppaan sisältöjen koostamisen, opintojaksokuvausten ja henkilötietojen kopioinnin sekä huolehti oppaan paino- ja verkkoversioiden julkaisuista.

Ulkoisia toimijoita olivat IT-tiedekunnan kanssa yhteistyössä olevat muut tiedekunnat ja Jyväskylän yliopiston hallinto sekä opiskelijapalvelut. Muut tiedekunnat ja laitokset sekä erilaiset opiskelijajärjestöt tuottivat esimerkiksi sivuaineopintoihin ja muuhun opiskelijaelämään liittyvää sisältöä oppaaseen. Hallinnon ja opiskelijapalveluiden toimijat tuottivat opinto-oppaaseen sisältöä esimerkiksi yliopisto-opiskelun hallinnollisiin asioihin liittyen.

### 5.2.2 Prosessit

Opinto-oppaan laadintaan liittyvät pääasialliset prosessit olivat:

- tekstisisältöjen laatiminen ja ylläpito,
- tekstisisältöjen hyväksyminen,
- oppaan koostaminen ja

- oppaan julkaiseminen.

Tekstisisältöjen laatiminen tarkoitti varsinaista sisällöntuotantoa eli oppaan tekstien tuottamista, oikolukua ja sisältöversioiden päivittämistä sisäisten ja ulkoisten toimijoiden toimesta. Ainelaitosten osuuksien lisäksi sisältökokonaisuuksia ei ollut määritelty tietyille vastuuhenkilöille, mikä sirpaloitti lähdedokumenttien laatimista. Yksi henkilö saattoi tuottaa tekstipaloja eri puolille opasta ja toimitti tekstit monessa osassa oppaan koostajalle. Lisäksi vastuualueista oli tietyissä asioissa erilaisia näkemyksiä sisällöntuottajien keskuudessa. Tutkintorakenteita koskevien tekstisisältöjen hyväksyminen oli laitosneuvostojen ja tiedekuntaneuvoston hoitama prosessi. Sisältöversioiden hyväksyminen neuvostoissa vei aikaa, sillä usein neuvostojen jäsenet eivät olleet yksimielisiä tuotetuista sisällöistä. Etenkin keväällä 2004 havaittiin, että jopa oppaan asiasisältöjen jäsenystä ja lukujen järjestystä käsiteltiin paljon muun muassa tiedekuntaneuvoston istunnoissa.

Oppaan koostamiseen kuuluvia aktiviteetteja olivat opintojaksokuvausten sekä henkilötietojen kopiointi oppaan liitteiksi ja lähdedokumenttien koostaminen yhtenäiseksi oppaaksi. Sisällöntuottajat toimittivat lähdetekstit erillisinä Word-dokumentteina, jotka opasryhmän tekninen asiantuntija koosti yhtenäiseksi kokonaisuudeksi. Tekstinkäsittelyohjelmien tyylimääritysten kanssa syntyi ongelmia, sillä sisällöntuottajilla ei ollut käytössä yhtenäistä dokumenttipohjaa. Jokainen lähdedokumentti sisälsi erilaisia tyylejä, jotka myös kopioituivat koostettuun versioon. Dokumentin kasvaessa tämä aiheutti pahimmassa tapauksessa tekstinkäsittelyohjelman kaatumisen liiallisten tyylimääritysten vuoksi.

Opintojaksokuvausten kopiointi oppaan liitteeksi hoidettiin hakemalla kuvaukset Korppi-järjestelmästä ja kopioimalla ne manuaalisesti WWW-selaimen näytöltä opastiedostoon. Mikäli opintojaksojen tiedot muuttuivat Korppi-järjestelmässä, jouduttiin toimenpide aina uusimaan. Henkilötietolistaukset kopioitiin niin ikään manuaalisesti oppaan liitteeksi Jyväskylän yliopiston sähköi-

sestä puhelinluettelosta avaamalla luettelo näytölle ja kopioimalla tiedot oppaaseen.

Oppaan julkaisuun liittyviin aktiviteetteihin kuului valmiiksi koostetun oppaan eli Word-dokumentin muuntaminen PDF- ja HTML-muotoihin. PDF-versio tuotettiin oppaan painettavaa paperiversiota varten ja HTML-versio verkkojulkaisua varten.

### **5.2.3 Sisältöyksiköt**

Oppaan laadintaan liittyviä pääasiallisia sisältöyksiköitä olivat Word-lähdedokumentit, jotka sisälsivät tuotetun sisällön eivätkä ottaneet juurikaan kantaa kokonaisen oppaan loogiseen rakenteeseen. Lähdedokumentit sisälsivät aina jonkin tietyn asiakokonaisuuden, joka sitten kopioitiin varsinaiseen oppaaseen tiettyyn kohtaan. Oppaan laadinnassa käytettyjä sisältöyksiköitä olivat omalla tavallaan myös WWW-selaimen näytöltä Korppi-järjestelmästä kopioidut opintojaksokuvaukset, jotka kopioitiin oppaan liitteiksi.

### **5.2.4 Järjestelmät**

Dokumenttienhallinta perustui verkkopalvelimelle luotuun hakemistorakenteeseen, jossa työstettävää opasmateriaalia säilytettiin ja muokattiin. Oppaan laadinnasta ja julkaisusta vastaavilla toimijoilla oli luku- ja kirjoitusoikeudet verkkopalvelimen opas-hakemistoon. Pääasiallinen työkalu oppaan laadinnassa oli Word-tekstinkäsittelyohjelma, jolla tekstisisällöt tuotettiin ja jonka avulla sisällöt koostettiin oppaaksi. Pääasiallinen kommunikointiväline oppaan laadinnasta vastaavien toimijoiden välillä oli sähköposti.

Korppi-järjestelmä oli tärkeä oppaan kannalta, sillä sinne tallennetut opintojaksokuvaukset voitiin hakea WWW-selaimella erilaisia hakutermejä käyttäen järjestettynä halutulla tavalla näytölle ja edelleen kopioitavaksi oppaaseen. Lisäksi oppaan laadinnassa hyödynnetyihin jo olemassa oleviin järjestelmiin

kuului WWW:n kautta käytettävä Jyväskylän yliopiston sähköinen puhelinluettelo, josta IT-tiedekunnan henkilökunnan tiedot kopioitiin oppaan liitteeksi.

Oppaan julkaisuissa käytettiin kolmannen osapuolen ohjelmistoja, joiden avulla oppaasta muunnettiin eri julkaisuversiot. Julkaisuun ei liittynyt varsinaisia ongelmakohtia, sillä Wordin käytön yleisyyden ansiosta olemassa oli lukuisia toimivia kaupallisia ja ilmaisia ohjelmistoja, jotka tukivat erilaisten julkaisujen tuottamista suoraan Word-dokumenteista.

### **5.3 Opinto-oppaan laatiminen kehittämishankkeen jälkeen**

Seuraavissa alaluvuissa kuvataan opinto-oppaan laatimisprosessin tila opinto-oppaan kehittämishankkeen jälkeen.

#### **5.3.1 Toimijat**

Opinto-oppaan kehittämishankkeen jälkeen oppaiden laadintaan liittyvät toimijat olivat pääosin samoja kuin alkutilassa, mutta sisäisten toimijoiden ja etenkin työryhmän jäsenten vastuualueita oli tarkennettu muun muassa XooZoo-projektin parannusehdotuksien pohjalta. Sisältöjä tuotettiin edelleen tiedekunnan sisällä ja ulkopuolella. Ulkoisten toimijoiden järjestelyihin ei ollut mahdollisuutta vaikuttaa, joten heidän kohdallaan tilanne oli vastaava alkutilaan nähden.

Sisäisten toimijoiden vastuualueiden tarkennus tarkoitti sitä, että nyt esimerkiksi ainelaitosten tekstien tuottamisesta vastaavat toimijat olivat selkeämmin molempien laitosten amanuenssit. Tiedekuntaan nimettiin myös erikseen kansainvälisistä asioista vastaava henkilö, joka vastasi oppaan kansainvälisiin asioihin liittyvistä asioista. Lisäksi jatko-opinnoista, kieli- ja viestintäopinnoista, ja yhteistyökumppaneiden tarjoamista opinnoista vastasivat tietyt vastuuhenkilöt.



Muista opintoihin ja opiskeluun ja tutkintoihin liittyvistä yleisistä asioista vastasi tiedekunnan opintoasiainpäällikkö.

Kehittämishankkeen aikana työryhmään kuului erillinen XML-tukihenkilö, joka vastasi lähdedokumenttien muuntamisesta XML-muotoon. Työryhmään kuului myös kaksi teknistä asiantuntijaa, jotka vastasivat XML-muotoisen oppaan PDF- ja XHTML-julkaisuista. Lisäksi työryhmään kuului vielä kolmas tekninen asiantuntija, joka avusti Linux-palvelinympäristön käytössä ja tarjosi apua muun muassa automaattisten päivitysten laadintaan. Tutkintorakenteisiin liittyvien sisältöjen hyväksyminen tapahtui laitos- ja tiedekuntaneuvostoissa. Viime kädessä oppaiden sisältöjen hyväksyjänä toimi opintoasiainpäällikkö.

### **5.3.2 Prosessit**

XML:n käyttöönoton myötä laadintaprosessi pysyi sisällöntuottajien kannalta pitkälti samana. Laadintaprosessi koostui edelleen sisältöjen laatimisesta, hyväksymisestä, sisältöjen koostamisesta ja oppaan julkaisusta. Merkittävimmät uudistukset laadintaprosessissa olivat lähdedokumenttien muuntaminen XML-muotoon ja julkaisuprosessin automatisointi.

Tekstisisältöjen laadinnassa ei sisällöntuottajien kannalta tapahtunut suuria muutoksia lukuun ottamatta vastuualueiden selkiytymistä. Sisällöntuotannon vastuualueita oli nyt tarkennettu, joten oppaan laadintaa koordinoiva opintoasiainpäällikkö tiesi paremmin kenen puoleen kääntyä tiettyjen sisältöjen suhteen. Tekstisisällöt laadittiin edelleen Word-dokumentteina, lukuun ottamatta opintoasiainpäällikköä, joka laati omat osuutensa suoraan XML-muotoon. XML-tukihenkilö vastasi Word-muodossa tuotettujen sisältöjen muuntamisesta XML-muotoon. Lähdedokumentit lähetettiin XML-tukihenkilölle sähköpostitse.

Eräs suuri muutos laadintaprosessissa oli se, että oppaista oli päätetty julkaista kaksi eri versiota: oma opas ensimmäisen vuoden opiskelijoille ja oma vanhemmille opiskelijoille. Tähän päädyttiin tiedekunnassa keväällä 2005 läpikäy-

dyn tutkintorakenneuudistuksen myötä ja XooZoo-projektissa selvitettyjen oppaan sisältöä koskevien parannusehdotuksien pohjalta (XooZoo 2005b). Uusille opiskelijoille ei katsottu olevan tarpeellista kertoa vanhoista tutkintorakenteista, koska se ei koskettanut heitä. Tällä pyrittiin selkiyttämään uusille opiskelijoille tarjottavaa informaatiota. Muutoksen myötä havaittiin, että molemmat opasversiot käyttivät osittain samoja lähdedokumentteja, joita voitiin käyttää uudelleen molemmissa opasversioissa. Tämä tuli huomioida koostamisprosessissa sekä lähdedokumenttien laadinnassa.

Oppaan loogisen rakenteen järjestystä ei enää käsitelty tiedekunta- eikä laitosten neuvostoissa, vaan lopullinen vastuu oppaan kokonaisuudesta oli tiedekunnan opintoasiainpäälliköllä. Tällä pyrittiin välttämään viivästyksiä oppaan laadinnassa, joita aikaisemmin oli ollut juuri kyseisestä syystä. Tutkintorakenteet ja hallinnolliset sisällöt kuitenkin hyväksyttiin ensin neuvostoissa, mutta niiden lopullisesta sijainnista oppaassa vastasi opintoasiainpäällikkö.

Opintojaksokuvausten kopiointi oppaisiin automatisoitiin. Opintojaksojen kuvaukset saatiin XML-muodossa Korppi-järjestelmän XML-rajapinnasta. Rajapinta oli toteutettu Suomen virtuaaliyliopiston (2005) kehittämää korkeakoulujen välistä opintotietopalvelua varten. Henkilötietojen kopiointi oppaaseen oli tärkeä asia, jonka automatisointia ei ehditty toteuttaa. Henkilötiedot muunnettiin manuaalisesti XML-muotoon oppaaseen koostamista varten.

Oppaiden koostamisprosessissa tapahtui myös suuria muutoksia. XML:n ansiosta opasversioiden lähdedokumenttien koostaminen pystyttiin hoitamaan varsin pitkälti automaattisesti. Manuaaliseen tekstipalasten kopiointiin Word-ohjelman näytöltä toiselle ei enää ollut tarvetta sen jälkeen, kun Word-dokumenttien sisältö oli muunnettu XML-muotoon. Koostaminen tarkoitti tekstisisältöjen, opintojaksokuvausten ja henkilötietolistojen koostamista yhdeksi XML-tiedostoksi määrittelemällä manuaalisesti uudelle rakennetulle julkaisu-järjestelmälle, mitkä lähdedokumentit koosteeseen haluttiin. Määrittely tarkoitti

käytännössä koosteen XML-muotoisen määrittelydokumentin kirjoittamista. Tämän jälkeen julkaisujärjestelmä koosti opasversion ja loi siitä PDF- sekä XHTML-julkaisut. Koostaminen tapahtui täten puoliautomaattisesti.

Opinto-oppaiden julkaisuprosessi koki myös suuria muutoksia. Oppaiden julkaisua helpottava julkaisujärjestelmä vastasi uuden opasversion koostamisesta ja julkaisusta eri muodoissa sekä koosteen automaattisesta päivittymisestä lähdesisältöjen muuttuessa. Opasversioiden julkaisua varten verkkopalvelimelle oli luotu Internetissä näkyvä WWW-portaali (Opinto-opas 2005), josta käyttäjä saattoi katsella oppaiden PDF- tai XHTML-versioita ja oppaan julkaisutietoja WWW-selaimella.

### 5.3.3 Sisältöyksiköt

Sisältöyksiköiden kohdalla tapahtui paljon muutoksia XML:n käyttöönoton seurauksena. Kaikki lähdemateriaali muunnettiin XML-muotoon. Opintojaksojen kuvaukset saatiin Korpista valmiiksi XML-muodossa, ja ne voitiin siirtää oppaaseen automaattisesti XSLT-muunnoksen (Clark 1999) avulla. Henkilötiedot kopioitiin edelleen manuaalisesti näytöltä WWW-tulosteesta, mutta alkutilaan nähden muutoksena oli henkilötietojen muuntaminen XML-muotoon.

Kaikki oppaan koostamisessa käytetyt lähdedokumentit, mukaan lukien tekstisisällöt, opintojaksojen kuvaukset ja henkilötiedot noudattivat XooZoo-projektissa luotua ja Xoo-jatko -projektin aikana tarkennettua luku.dtd:tä. Luku.dtd:n semantiikka oli lähellä XHTML-määritystä (Pemberton, Austin, Axelson ym. 2000), semantiikan ollessa kuitenkin kuvattu suomen kielellä. Opintojaksokuvaukset noudattivat alun perin omaa määritystään, mutta ne muunnettiin luku.dtd:n mukaisiksi koostamista varten. Luku.dtd on tämän tutkielman liitteenä (Liite 2).

Lähdesisällöt oli jaoteltu luvuittain siten, että yksi XML-muotoinen lähdedokumentti vastasi yhtä lukua opasversiossa. Molemmat opasversiot käyttivät

osittain samoja lähdedokumentteja. Tällä pyrittiin minimoimaan toisteisen tiedon tallentaminen verkkopalvelimella, koska tiettyjä lähdedokumentteja käytettiin uudelleen molemmissa opasversioissa. Toisteiselta tiedolta ei kuitenkaan täysin välttytty, sillä lähdedokumenttien raekoko asetti omia rajoitteitaan lähdesisältöjen uudelleenkäytölle: yksi lähdedokumentti vastasi aina yhtä kokonaista lukua oppaassa. Jossakin luvussa saattoi kuitenkin olla toisen opasversion kannalta epärelevanttia informaatiota. Täten jokainen lähdedokumentti ei ollut suoraan käytettävissä molemmissa opasversioissa.

#### 5.3.4 Järjestelmät

Kehittämishankkeen jälkeen merkittävimpiä käytännön työkaluja oppaan laadinnassa olivat sisällöntuottajien edelleen käyttämä Wordtekstinkäsittelyohjelma, XML-tukihenkilön ja teknisten asiantuntijoiden käyttämät erilaiset XML-työkalut kuten XMLSpy (Altova 2005) ja Eclipsekehitysympäristö (Eclipse 2005) sekä kehittämishankkeen jäsenien välisessä kommunikoinnissa käytetty sähköposti. Erilaisten sisällöntuotannossa käytettyjen ohjelmistojen lukumäärä lisääntyi siis kahdella.

Erilaisia opinto-oppaan kannalta oleellisia järjestelmiä ja muita ohjelmistoja olivat esimerkiksi Korppi, josta saatiin opintojaksokuvaukset sekä Jyväskylän yliopiston sähköinen puhelinluettelo, josta tiedekunnan henkilötiedot kopioitiin oppaisiin. Lisäksi kehittämishankkeen aikana oli käytetty lukuisa määrä erilaisia pienempiä ohjelmia, määrittäjiä ja sovelluksia. Näitä tarvittiin jatkoprojektin aikana kehitetyn oppaan uuden koostamis- ja julkaisujärjestelmän toteutuksessa. Yksittäisiä ohjelmia oli käytössä yhteensä yhdeksän kappaletta.

Tällaisia ohjelmia olivat esimerkiksi koostamisprosessissa käytetyt XSLT-prosessorit. Erillisen asennuksen vaativia ohjelmia oli yhteensä viisi kappaletta; kuten esimerkiksi LaTeX-ladontajärjestelmä (LaTeX 2005) ja verkkopalvelimen ohjelmisto. Koostamisprosessissa tarvittavia XSLT-muunnostiedostoja oli käy-

tössä yhteensä 19 kappaletta. DTD-määrittelyksiä oli kahdeksan kappaletta, joista kaikkia ei kuitenkaan käytetty. LaTeX-ladontajärjestelmän avulla hoidettu PDF-muunnos vaati yhteensä 17 erilaista määrittelytiedostoa, jotka sisälsivät ohjeita sisältöjen käsittelyyn muunnoksessa. Erilaisia komentojonotiedostoja ja skriptejä tarvittiin julkaisujärjestelmässä yhteensä 13 kappaletta.

Integraatio Korppi-järjestelmän kanssa oli lähes kokonaan automaattinen. Opintojaksokuvaukset haettiin verkon yli Korpista, jonka XML-rajapinta palautti kuvaukset Suomen virtuaaliyliopiston laatiman määrittelyksen mukaisina XML-dokumentteina. Tästä syystä opintojaksokuvaukset jouduttiin muuntaamaan haun jälkeen XSLT:n (Clark 1999) avulla opinto-oppaalle sopivaan muotoon. Opintojaksokuvaukset haluttiin pitää mahdollisimman ajan tasaisina, joten julkaisujärjestelmä ohjelmoitiin hakemaan opintojaksokuvaukset Korpista kerran viikossa.

Jotta halutut opintojaksot saatiin haetuksi, täytyi hakuohjelmalle syöttää tiettyjä parametreja, jotka hakuohjelma välitti edelleen Korpin tietokannanhallintajärjestelmälle. Kaikkia tarvittavia opintojaksokuvauksia ei kuitenkaan saatu haetua yhdellä haulla. Tämä johtui Korpin vaatimien parametrien asettamista rajoitteista. Hakuohjelma ohjelmoitiin tekemään useita peräkkäisiä hakuja, joiden palautteet koostettiin yhteen. Joitain yksittäisiä opintojaksoja jouduttiin koostamaan eri hauilla, jotta kaikki tarpeelliset opintojaksot saatiin oppaaseen mukaan.

Käytännössä hakuohjelmaa käytettiin komentoriviltä, jolloin opintojaksokuvausten haku vaati tietämystä komentoriviohjelmoinnista. Helppokäyttöistä käyttöliittymää opintojaksokokonaisuuksien hakuun ja koostamiseen ei ollut. Integraatio vaati myös muutamia opinto-opas-hanketta varten tehtyjä muutoksia Korpin XML-rajapintaan: esimerkiksi hakujen nopeuttamiseksi joitakin Korpin palauttamia tietoja jouduttiin karsimaan ja joitain tietoja taas lisäämään, jotta kaikki tarvittavat tiedot saatiin haussa mukaan. Koko integraatioprosessi saatiin

lähes kokonaan automatisoitua, lukuun ottamatta manuaalisen muokkauksen tarvetta opintojaksokoosteissa. Tämä johtui käytännössä siitä, että Korppiin oli tallennettu joidenkin opintojaksojen kohdalle väärää tietoa, jotka automaattisesti myös kopioituivat oppaaseen. Virhe oli opintojaksosta vastaavan henkilön eikä teknologiaan liittyvä puute.

Joidenkin yksittäisten opintojaksojen tapauksissa ilmeni ylimääräisiä tulostuksia, koska hakuparametrit vastasivat sille annettuja arvoja useammassa kuin yhdessä haussa. Tapaus koski esimerkiksi sellaisia opintojaksoja, jotka luennoitiin useamman kerran täsmälleen samanlaisina. Useamman tulostuksen estämiseksi näitä opintojaksoja jouduttiin poistamaan manuaalisesti. Lisäksi joku opintojakso saattoi tulostua sellaiseen kohtaan mihin ei haluttu. Tämä johtui opintojakson koodista, jonka perusteella sen paikka koosteessa määräytyi. Tällaiset opintojaksot jouduttiin manuaalisesti siirtämään oikeaan paikkaan.

Dokumenttienhallinta perustui Linux-alustaiselle verkkopalvelimelle luotuun hakemistorakenteeseen, jossa molemmille opasversioille oli omat erilliset hakemistonsa. Varsinaista dokumenttienhallintajärjestelmää ei ollut käytössä joh-tuen muun muassa kustannussyistä ja Xoo-jatko -projektin aikaresursseista.

XooZoo-projektin aikana oli luotu erilaisia DTD-määrittäjiä oppaan sisällön tarkkaa merkkaukseen varten. Oppaiden erilaisten julkaisuun liittyvien tietojen merkkaukseen varten oli laadittu oma DTD. Tällaisia tietoja olivat esimerkiksi oppaan laatijoiden nimet, painopaikkatiedot ja julkaisunumerotiedot.

Oppaan sisällöntuotannon ja ulkoasun kannalta tärkein rakennemäärittäjä oli tekstisisältöjen merkkaukseen varten laadittu luku.dtd-määrittäjä, joka sisälsi rakenteellisia ja ulkoasullisia määrittäjiä. Tällaisia määrittäjiä olivat otsikkotasot, kappaleet, listaukset, kuvat, hyperlinkit ja tekstimuotoilut kuten esimerkiksi lihavoinnit. Luku.dtd oli lopulta pääasiallinen DTD, jota opinto-oppaiden lähdedokumenttien merkkauksessa käytettiin.

Metatietoja varten oli laadittu myös oma DTD, jota oli tarkoitus käyttää esimerkiksi tekstilukujen metatietojen syöttöä varten. Metatietoihin oli tarkoitus syöttää esimerkiksi tekstiversion laatijan nimi, päiväys ja version tila. Lisäksi tekstin kommentoinnille ja hyväksynnälle oli annettu mahdollisuus. Meta.dtd:tä käytettiin jossain määrin lähdesisältöjen metatietojen syöttämiseen.

Tiedekunnan tutkintojen tarkempaa merkkeä varten oli laadittu DTD-määrittys. Tämän tarkoituksena oli merkata tiedekunnan tarjoamat erilaiset tutkinnot ja niihin liittyvät tiedot, jotta ne olisi mahdollista poimia tehokkaasti oppaan sisällön seasta. Tutkinnot.dtd:n todettiin Xoo-jatko -projektin aikana kuitenkin olevan todella keskeneräinen, joten sitä ei käytetty oppaiden merkkämisessä lainkaan.

Oppaan liitteenä oleville opintojaksokuvauksille luotiin myös DTD. Tätä oli tarkoitus käyttää hyödyksi automatisoitaessa opintojaksokuvausten siirtoa Korpista oppaaseen. Tätä määrittystä ei kuitenkaan tarvittu, sillä Korppi tarjosi opintojaksokuvaukset valmiiksi XML-muodossa. Tämän mahdollisti Korppi-järjestelmään kehittämishankkeen ulkopuolella rakennettu rajapinta, joka tiettyjen parametrien avulla pyytäessä palautti haluttujen opintojaksojen kuvaukset XML-muodossa.

Niin ikään liitteenä oppaassa oleville tiedekunnan henkilötiedoille oli luotu DTD henkilötietojen automatisoinnin tueksi tulevaisuutta ajatellen. Tätäkään ei lopulta tarvittu, koska henkilötiedot kopioitiin manuaalisesti näytöltä luku.dtd:n mukaiseen muotoon.

Lähdedokumenttien koostamisesta vastaava järjestelmä oli pieni XML/XSLT -sovellus. Kooste-määrittymän mukaiseen XML-tiedostoon voitiin luetella koosteeseen halutut lähdedokumentit ja niiden järjestys. Molempia opasversioita varten oli omat koostemäärittymänsä, jotka osittain käyttivät samoja tiedostoja. Molempien opasversioiden koostemäärittymät ovat tämän tutkielman liitteinä (3 ja 4). XML-muotoiseen koostetiedostoon oli mahdollisuus merkata lähdedo-

kumentit esimerkiksi oppaan esiluvuiksi, varsinaisiksi tekstiluvuiksi tai liitteiksi. Tämä vaikutti lopullisessa oppaan tulostuksessa esimerkiksi lukujen numerointiin ja otsikointiin. Yksi lähdedokumentti vastasi aina yhtä kokonaista lukua oppaassa. Kun koostemääritys oli valmis, XSLT-muunnos kokosi lähdedokumentit yhdeksi XML-dokumentiksi, eli opasversioksi.

Oppaan PDF-julkaisussa käytettiin LaTeX-ladontajärjestelmää (LaTeX 2005). PDF-julkaisua varten opas muunnettiin XSLT-muunnoksella (Clark 1999) ensin LaTeXin ymmärtämään muotoon, jonka jälkeen LaTeX tulosti oppaasta PDF-version. XHTML-julkaisu hoidettiin XSLT-muunnoksella suoraan koostetusta opas-tiedostosta. Molemmissa elektronisissa julkaisuissa, sekä PDF- että XHTML-versioissa, oli oppaan luettavuutta ja käytettävyyttä helpottava automaattisesti luotu sisällysluettelo, jonka otsikot toimivat hyperlinkkeinä oppaan sisältölukuihin.

Verkkopalvelimelle rakennetun julkaisujärjestelmän avulla opasversioiden julkaisua voitiin hallita WWW-selaimella käytettävän hallintakäyttöliittymän välityksellä. Julkaisujärjestelmä mahdollisti esimerkiksi opasversioiden automaattisen päivityksen. Lähdedokumenttien päivittyessä järjestelmä kutsui ohjelmaa, joka loi uuden työversion työn alla olevasta opasversiosta esikatselua varten. Mikäli muokkaaja oli tyytyväinen tulokseen, hän saattoi julkaista oppaasta virallisen version hallintakäyttöliittymän avulla yhdellä hyperlinkin painalluksella. Täten mahdolliset päivityksen yhteydessä sattuneet virheet oli mahdollista korjata ennen varsinaisen julkaisun päivittymistä.

Järjestelmä päivitti sekä opasversion PDF- että XHTML-versiot. Täten julkaistujen oppaiden tiedot olivat nopeasti ajan tasalla. Lisäksi hallintakäyttöliittymän avulla voitiin suorittaa automatisoitu opintojaksokuvausten haku Korpista. Hakuohjelman parametreja ei kuitenkaan voinut muokata hallintakäyttöliittymän avulla. Hallintakäyttöliittymän avulla voitiin vain käynnistää ohjelmoitu haku. Haun muokkaamiseksi oli muokattava hakuohjelman komentorivikäsky-



jä. Hallintakäyttöliittymän yhteyteen oli tarkoitus liittää kaikki automatisoivissa olevat julkaisujärjestelmän toiminnot, jotta nämä kaikki olisivat käyttäjän hallittavissa helposti yhdessä paikassa WWW-selaimen avulla.

Taulukkoon 1 on koottu kehittämishankkeen myötä tapahtuneet olennaisimmat muutokset opinto-oppaan laadintaprosessissa. Muutokset on jaoteltu osaluokittain Salmisen (2003b) esittämän viitekehyksen mukaan. Ne asiat, joita taulukossa ei ole mainittu, säilyivät pääpiirteittäin samanlaisina myös kehittämishankkeen jälkeen.

Taulukko 1. Yhteenvedo muutoksista opinto-oppaan laadintaprosessin osa-alueissa

Osa-alue	Alkutila	Jälkitila
Toimijat	<ul style="list-style-type: none"> <li>- Tiedekuntaneuvosto oppaan loogisen rakenteen hyväksymisessä mukana.</li> <li>- Työryhmän jäsenten vastuualueet osittain epäselvät.</li> <li>- Tekninen asiantuntija vastuussa oppaan koostamisesta ja julkaisusta.</li> </ul>	<ul style="list-style-type: none"> <li>- Opintoasiainpäällikkö oppaan loogisen rakenteen hyväksyjänä.</li> <li>- Työryhmän jäsenten vastuualueet selkeämpiä.</li> <li>- XML-tukihenkilö vastasi lähdedokumenttien muuntamisesta XML-muotoon.</li> <li>- Tekniset asiantuntijat vastasivat julkaisujärjestelmän laadinnasta.</li> </ul>
Prosessit	<ul style="list-style-type: none"> <li>- Laadittiin yksi opas kaikille opiskelijoille.</li> <li>- Opintojaksokuvausten manuaalinen kopiointi.</li> <li>- Oppaan koostaminen manuaalinen toimenpide.</li> <li>- Julkaisu erilaisia kolmansien osapuolien työkaluja hyödyntäen.</li> </ul>	<ul style="list-style-type: none"> <li>- Laadittiin kaksi opasversiota.</li> <li>- Opintojaksokuvausten kopiointi automaattista.</li> <li>- Oppaan koostaminen puoliautomaattista.</li> <li>- Uusi julkaisujärjestelmä, joka tuotti automaattisesti PDF- ja XHTML-versiot.</li> </ul>
Sisältöyksiköt	<ul style="list-style-type: none"> <li>- Lähdedokumentit Word-muodossa.</li> <li>- Lähdedokumenteista koottiin yksi Word-tiedosto eli opinto-opas.</li> <li>- Korppi-näytöt opintojaksokuvausten manuaalisessa kopioinnissa.</li> </ul>	<ul style="list-style-type: none"> <li>- Word-lähdedokumentit muunnettiin XML-muotoon .</li> <li>- XML-muotoisten lähdedokumenttien raekoko vastasi yhtä kokonaista lukua opasversiossa.</li> <li>- Lähdedokumentit osittain samoja molemmissa opasversioissa.</li> <li>- Korppiyhteyden automatisointi.</li> </ul>
Järjestelmät	<ul style="list-style-type: none"> <li>- Word (sisällöntuottajien työkalu, oppaan tallennusmuoto).</li> <li>- Korppi (opintojaksokuvausten kopiointi manuaalisesti).</li> </ul>	<ul style="list-style-type: none"> <li>- Word (sisällöntuottajat).</li> <li>- XML-työkalut (XMLSpy, Eclipse, tekstisisältöjen muuntaminen XML-muotoon XML- ja XSLT -prosessorit).</li> <li>- Korppi (opintojaksokuvausten kopiointi automaattisesti kerran viikossa).</li> <li>- Uusi julkaisujärjestelmä oppaiden koostamista varten (LaTeX PDF-, ja XSLT XHTML-versioiden tuottamisessa).</li> </ul>

Seuraavassa luvussa opinto-oppaan koostamismalli kuvataan yksityiskohtaisemmin, ja sitä verrataan Heikkisen (2000) esittelemään dokumenttien koostamismalliin. Arvioinnissa pyritään tutkimaan millä tavoin tapaustutkimus poikkeaa aiemmin esitetystä koostamisesta, ja mitä heikkouksia ja vahvuuksia opinto-oppaan koostamismalliin liittyy. Lisäksi tarkoituksena on tutkia, löytyykö opinto-oppaan koostamismallista uusia XML-dokumenttien koostamiseen liittyviä ilmiöitä.

## 6 OPINTO-OPPAAN KOOSTAMINEN

Tässä luvussa opinto-oppaan koostamismalli kuvataan sekä graafisesti että sanallisesti. Opinto-oppaan koostamismallia verrataan Heikkisen (2000) esittämään koostamismalliin ja arvioidaan opinto-oppaiden lähdedokumenttien raekoon vaikutusta koostamiseen. Lisäksi pohditaan, kuinka integraatio muiden järjestelmien kanssa onnistui. Integraatiolla muiden järjestelmien kanssa tarkoitetaan esimerkiksi joissain muissa järjestelmissä tallennettuna olevien tietojen automaattista siirtämistä koostettavaan dokumenttiin. Luvussa esitetään myös joitain käytännön kokemuksia opinto-oppaan kehittämishankkeen ajalta.

### 6.1 Arvioinnin kohteet

Tapaustutkimuksen arvioinnissa tärkeimpänä arvioitavana asiana on opinto-oppaan koostamismallin vertailu Heikkisen (2000) esittämään XML-dokumenttien koostamismalliin. Muita pohdittavia asioita ovat muun muassa kuinka pitkälle tutkittavan tapauksen koostaminen voitiin automatisoida ja millaisia ongelmia koostamiseen liittyi. Koostamisen automatisoinnin kannalta tärkeä seikka on automaattisesti tuotettujen XML-muotoisten lähdedokumenttien manuaalisen ylläpidon ja muokkaamisen tarve. Lähdedokumenttien osalta tarkastellaan myös lähdedokumenttien raekokoa ja sen merkitystä XML-dokumenttien koostamisessa. Tutkittava tapaus on tyypiltään melko tavallinen monikanavajulkaisu, jossa XML-muodossa olevasta lähdemateriaalista julkaistaan versiot paperi- sekä verkkojulkaisua varten. Tämä tekee tapaustutkimuksesta ja sen havainnoista entistä yleistettävämmän.

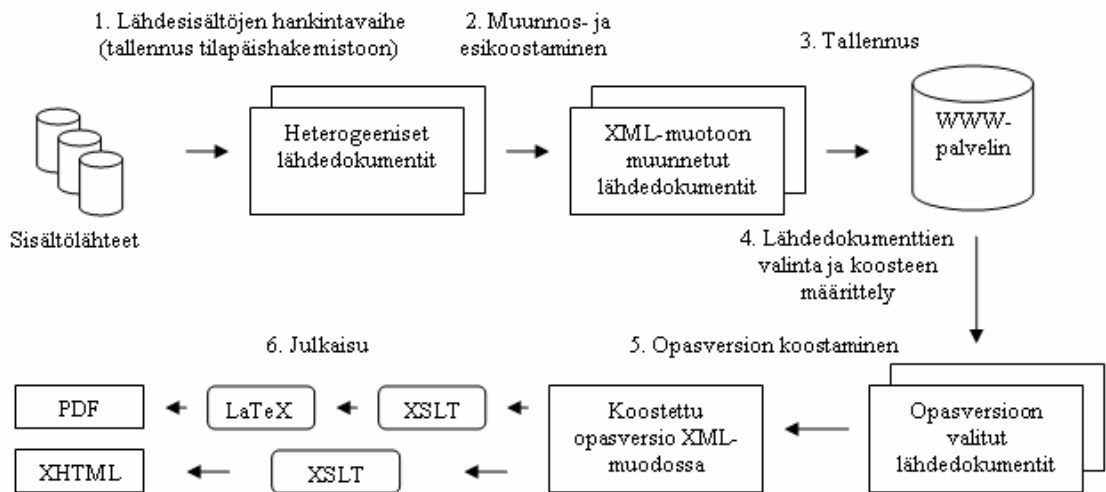
## 6.2 Opinto-oppaan koostamismalli

Seuraavaksi opinto-oppaan koostamismalli kuvataan pääpiirteittäin sekä graafisesti että sanallisesti. Kuvaus alkaa lähdesisältöjen hankintavaiheesta (content discovery). *Lähdesisältöjen hankintavaiheella* tarkoitetaan sisältöjen hankkimista niiden alkuperäisessä muodossaan erilaisista tietojärjestelmistä ja dokumenttikokoelmista koostettaessa uutta dokumenttia. (Mcintosh 2000)

Kuviossa 7 on kuvattu graafisesti opinto-oppaan koostamismalli kehittämissankkeen jälkeen. Koostamismallista voidaan erottaa kuusi päävaihetta, jotka karkealla tasolla kuvattuna ovat:

1. lähdesisältöjen hankintavaihe, jossa kaikki tarvittavat lähdesisällöt koottiin ja tallennettiin WWW-palvelimelle tilapäishakemistoon odottamaan jatkokäsittelyä,
2. lähdesisältöjen muunnos- ja esikoostamisvaihe, jossa kootut heterogeeniset lähdesisällöt muunnettiin XML-dokumenteiksi osin automaattisesti ja osin manuaalisesti, ja jossa osa lähdesisällöistä myös esikoostettiin,
3. lähdedokumenttien tallennusvaihe, jossa XML-muotoon muunnetut lähdedokumentit tallennettiin verkkopalvelimelle joko automaattisesti tai manuaalisesti oikeisiin hakemistoihin säilytystä varten,
4. lähdedokumenttien valinta ja koosteen määrittelyvaihe, joka tehtiin kirjoittamalla XML-muotoiseen koostedokumenttiin, mitkä lähdedokumentit opasversioon haluttiin koostaa,
5. opasversion koostamisvaihe, jossa julkaisujärjestelmä automaattisesti koosti lähdedokumentit opasversioksi ja

6. opasversion julkaisuvaihe, jossa julkaisujärjestelmä kutsui eri julkaisu-  
muodoista huolehtivia ohjelmia muodostaen opasversiosta erilaiset jul-  
kaisu-



Kuvio 7. Opinto-oppaan koostamismalli

Kuviossa 7 nuolet kuvaavat koostamisprosessin kulkusuuntaa ja samalla prosessin eri vaiheita. Lieriöt kuvaavat tietovarastoja. Suorakaiteet kuvaavat dokumentteja niiden eri vaiheissa. Pyöreäreunaiset suorakaiteet kuvaavat opasversion julkaisuun käytettyjä ohjelmia. Graafisessa kuvauksessa on pyritty noudattamaan Heikkisen (2000) esittämää koostamismallin notaatiota mallien vertailun helpottamiseksi.

Opinto-oppaan koostamismalli kuvataan vaiheittain yksityiskohtaisemmin seuraavissa alaluvuissa. Koostamisprosessi oli molempien opasversioiden kohdalla samanlainen.

### 6.2.1 Lähdesisältöjen hankinta

Opinto-oppaan koostamismallin ensimmäisessä vaiheessa, lähdesisältöjen hankintavaiheessa, kaikki opasversioita varten tarvittavat lähdesisällöt koottiin erilaisista sisältölähteistä. Erilaisia sisältölähteitä ovat tässä tapauksessa sisällöntuottajien tuottamat Word-dokumentit, opintoasiainpäällikön tuottamat XML-dokumentit, Korppi-järjestelmän tuottamat XML-muotoiset opintojaksokuvaukset ja Jyväskylän yliopiston sähköisestä puhelinluettelosta kopioidut IT-tiedekunnan henkilötiedot. Täten sisältölähteinä toimivat joko dokumentit tai erilliset tietojärjestelmät. Hankintavaiheessa kootut lähdesisällöt tallennettiin tilapäishakemistoihin kuviossa 7 kuvatulle WWW-palvelimelle jatkokäsittelyä varten.

Lähdesisältöjen hankinta tapahtui osin automaattisesti ja osin manuaalisesti. Korppi-järjestelmästä saadut opintojaksokuvaukset koottiin hakuohjelman avulla automaattisesti verkon kautta määrittelemällä hakuohjelmalle tietyt hakuparametrit. Hakuohjelman välittämien parametrien avulla Korppi osasi palauttaa halutut opintojaksokuvaukset XML-tiedostoina, jotka tallennettiin WWW-palvelimelle tilapäishakemistoon. Kaikkien tarvittavien opintojaksokuvausten saamiseksi hakuja jouduttiin tekemään useita. Kunkin haun tuloksena oli yksi XML-tiedosto. Manuaalisesti koottuja lähdesisältöjä olivat sisällöntuottajien tuottamat Word-dokumentit, jotka sisälsivät opasversioiden varsinaiset tekstisisällöt. Word-dokumentit tallennettiin WWW-palvelimelle omaan hakemistoon. Osa tekstisisällöistä tuotettiin valmiiksi XML-muotoon ja siten myös tallennettiin jo hankintavaiheessa XML-muodossa WWW-palvelimelle. Tällaisia XML-tiedostoja ei tallennettu tilapäishakemistoon vaan suoraan tietyn opasversion lähdedokumenttien säilytyshakemistoon. Manuaalisesti koottuja lähdesisältöjä olivat myös Jyväskylän yliopiston (2005) sähköisestä puhelinluettelosta kootut IT-tiedekunnan henkilötiedot. Henkilötiedot tallennettiin WWW-palvelimelle suoraan XML-muotoon kolmeen erilliseen tiedostoon, tiedekunnan

laitosten mukaan jaoteltuna odottamaan jatkokäsittelyä. Jaottelulla tavoiteltiin henkilötietolistojen helpompaa ylläpitoa tiedekunnan laitoksilla.

### **6.2.2 Lähdesisältöjen muuntaminen ja esikoostaminen**

Tässä vaiheessa tilapäishakemistoihin tallennetut lähdesisällöt muunnettiin luku.dtd:n mukaisiksi XML-dokumenteiksi osin automaattisesti ja osin manuaalisesti. Joitakin lähdedokumentteja myös esikoostettiin tässä vaiheessa. Tällaisia olivat esimerkiksi opintojaksokuvaukset ja IT-tiedekunnan henkilötiedot. Word-dokumentit muunnettiin manuaalisesti luku.dtd:n mukaisiksi lähdedokumenteiksi XML-työkalujen avulla. Korpista saatuja XML-muotoisia opintojaksokuvauksia jouduttiin ensin esikoostamaan, jotta useiden hakujen tulosedokumentit saatiin samaan XML-tiedostoon. Tämän jälkeen yhteenkuuluvat opintojaksokuvaukset muunnettiin luku.dtd:n mukaiseksi XML-dokumentiksi XSLT-sovelluksen avulla. Toimenpide oli lähestulkoon automaattinen muutama manuaalisesti tehtyä muutosta lukuun ottamatta. Tällaisia muutoksia olivat joissakin tapauksissa useampaan kertaan esiintyneiden opintojaksojen ylimääräisten ilmentymien poistamiset. Muunnosvaiheessa opintojaksokuvauksia jouduttiin siistimään erillisellä skriptillä. Siistimistä vaativat esimerkiksi opintojaksokuvausten sekaan kirjoitetut HTML-elementtitunnisteet. Siistimistöimenpide oli automaattinen. Myös henkilötiedot esikoostettiin tässä vaiheessa. Kolmeen eri tiedostoon jaetut henkilötiedot koostettiin yhdeksi XML-tiedostoksi, jotta henkilötiedot vastaisivat yhtä kokonaista liitettä opasversioissa.

### **6.2.3 Lähdedokumenttien tallennus WWW-palvelimelle**

Tallennusvaiheessa kaikki edellisessä vaiheessa muunnetut ja esikoostetut lähdedokumentit tallennettiin osin automaattisesti ja osin manuaalisesti WWW-palvelimelle oikeisiin hakemistoihin odottamaan varsinaista opasversion koostamista. Manuaalisesti tallennettavia lähdedokumentteja olivat varsinaiset tekstisisällöt, joiden yhteydessä oli tiedettävä mihin hakemistoon ne piti tallentaa.



Molemmille opasversioille oli luotu omat hakemistonsa, ja osa lähdedokumenteista oli yhteiskäyttöisiä molemmissa opasversioissa. Tällaiset lähdedokumentit tallennettiin ainoastaan toisen opasversion hakemistoon. Molemmissa opasversioissa oli liitteenä toisistaan poikkeavat opintojaksokuvaukset, jotka oli erikseen esikoostettu edellisessä vaiheessa. Opintojaksokuvausten tallennus oikeaan hakemistoon hoitui automaattisesti molempien opasversioiden kohdalla. Myös esikoostetut henkilötiedot tallennettiin tässä vaiheessa automaattisesti oikeaan hakemistoon. WWW-palvelimen opinto-opas-järjestelmälle varatusta hakemistosta otettiin säännöllisin väliajoin varmuuskopioita mahdollisten virhetilanteiden varalle.

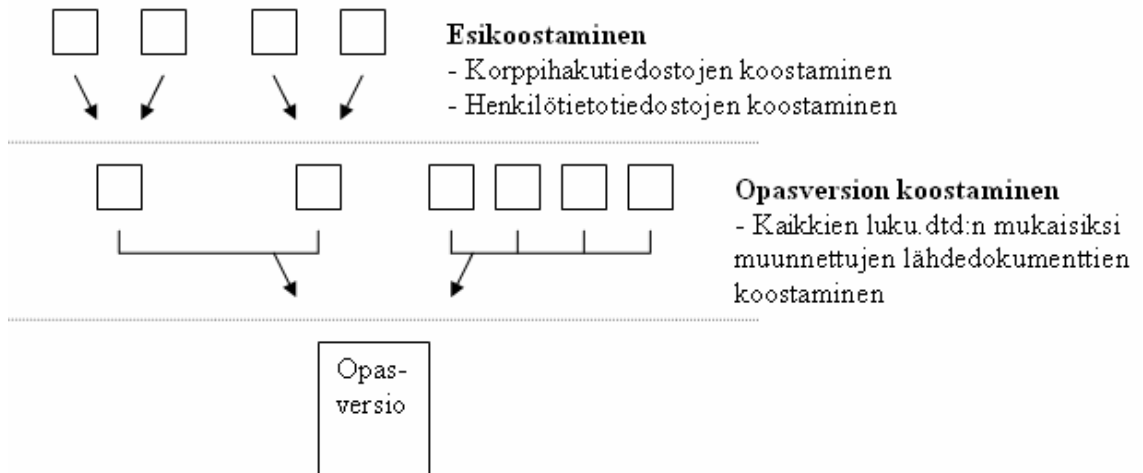
#### **6.2.4 Lähdedokumenttien valinta ja opaskoosteen määrittely**

Tässä vaiheessa edellisissä vaiheissa kootut lähdesisällöt oli muunnettu luku.dtd:n mukaisiksi XML-lähdedokumenteiksi. Lähdedokumentit olivat tallennettuina oikeissa hakemistoissa ja valmiita opasversioiden koostamista varten. Vaiheessa määriteltiin, mikä opasversio haluttiin luoda, ja mitä lähdedokumenteja siihen kuului. Käytännössä tämä tapahtui kirjoittamalla XML-muotoinen dokumentti, jossa lueteltiin mitkä lähdedokumentit opasversioon haluttiin mukaan ja mikä niiden järjestys oppaassa oli. Yksi lähdedokumentti vastasi yhtä kokonaista lukua oppaassa. Lisäksi koostetiedostossa määriteltiin millaisia lukuja lähdedokumenteista haluttiin oppaassa; esimerkiksi jotkin luvut haluttiin numeroimattomiksi esiluvuiksi, toiset numeroiduiksi ja joistakin luvuista tuli oppaan liitteitä. Nämä asiat merkattiin koostetiedostoon, jotta julkaisujärjestelmä osasi käsitellä lähdedokumenteja oikein. Molempien opasversioiden koostetiedostot tallennettiin opasversioiden omiin hakemistoihin WWW-palvelimella (Liite 3, liite 4).

### 6.2.5 Opasversion koostaminen

Tässä vaiheessa luku.dtd:n mukaisista lähdedokumenteista koostettiin yksi XML-dokumentti eli opasversio. Julkaisujärjestelmä käsitteli edellisessä vaiheessa luodun XML-muotoisen koostetiedoston, jonka perusteella tarvittavat lähdedokumentit saatiin opasversioon mukaan oikeaan järjestykseen oikealla tavalla merkattuna. Tuloksena syntyi XML-muotoinen, luku.dtd:n mukainen opinto-opasversio, jossa kaikki lähdedokumentit olivat koottuna yhteen tiedostoon. Opasversio oli valmis erilaisten julkaisujen tuottamista varten. Koostettavia lähdedokumentteja ensimmäisen vuoden opiskelijoiden oppaassa oli yhteensä 24 kappaletta, joista kaksi olivat esilukuja, 17 oli varsinaisia tekstilukuja, kolme oli liitteitä ja kaksi oppaan takalukuja. Vanhempien opiskelijoiden oppaassa koostettavia lähdedokumentteja oli yhteensä 25 kappaletta, joista kaksi olivat esilukuja, 18 oli varsinaisia tekstilukuja, kolme oli liitteitä ja kaksi oppaan takalukuja. Molempien opasversioiden numeroimattomina takalukuina olivat Jyväskylän yliopiston kampuksen kartan tekstiselitykset muunnettuna XML-muotoon sekä opasversion päivitystiedot.

Kuviossa 8 on esitetty molemmat koostamisvaiheet: esikoostamisvaihe ja varsinaisen opasversion koostamisvaihe. Esikoostamisvaiheessa Korpista haetut opintojaksokuvaukset ja manuaalisesti XML-muotoon muunnetut henkilötiedot koostettiin isommiksi kokonaisuuksiksi, eli yhtä kokonaista opasversion liitettä vastaavaan muotoon. Tavoitteena oli saada kaikista opasversioon koostettavista lähdedokumenteista rakenteeltaan yhdenmukaiset. Varsinaisessa opasversion koostamisvaiheessa kaikki lähdedokumentit olivat luku.dtd-määrityksen mukaisia niin, että yksi tiedosto vastasi yhtä oppaan lukua tai liitettä. Varsinaiset tekstiluvut muunnettiin manuaalisesti valmiiksi oikeaan muotoon, joten niitä ei tarvinnut esikoostaa opintojaksokuvausten ja henkilötietojen tapaan.



Kuvio 8. Koostamisvaiheet

Kuviossa 8 neliöt kuvaavat koostettavia lähdedokumentteja. Esikoostamisvaihe ja varsinainen koostamisvaihe on erotettu toisistaan katkoviivoilla. Nuolet kuvaavat lähdesisältöjen ja lähdedokumenttien koostumista toistensa kanssa aina isommaksi kokonaisuudeksi. Alimmaisena oleva suorakulmio kuvaa koostettua opasversiota.

### 6.2.6 Opasversion julkaisu

Julkaisuvaiheessa opasversiosta muunnettiin PDF- ja XHTML-julkaisut LaTeX-ladontajärjestelmän ja XSLT-sovelluksen avulla, joita julkaisujärjestelmä automaattisesti kutsui. PDF-versiota varten opasversio jouduttiin ensin muuntaamaan XSLT-sovelluksen avulla LaTeX-ladontajärjestelmän ymmärtämään muotoon, jonka jälkeen LaTeX-järjestelmä muodosti oppaasta PDF-muotoisen julkaisun. Julkaisujärjestelmä huolehti näiden toimenpiteiden kutsumisesta automaattisesti. XHTML-version julkaisusta huolehti XSLT-sovellus, joka tulosti koko oppaan kerralla WWW-selaimen näytölle. XHTML-muotoiseen julkaisuun luotiin myös erillisellä XSLT-sovelluksella oppaan verkkoversion käyttöä helpottava sisällysluettelo, jonka otsikot toimivat hyperlinkkeinä oppaan sisältöön.

Opasversioista oli tarjolla myös toisenlainen XHTML-versio, joka oli paloitetu pienempiin osiin erillisellä ohjelmalla. Tämä versio oli tarkoitettu esimerkiksi hitaammille verkkoyhteyksille ja käyttäjille, jotka halusivat selata opasta pienemmissä osissa kerrallaan. Opinto-oppaiden XHTML- ja PDF-versiot julkaistiin WWW-palvelimelle luodussa opasportaalissa, joka oli julkinen kaikille Internetin käyttäjille. Oppaiden PDF-versiot toimitettiin myös painoon, jossa niitä julkaistiin myös painetut versiot.

### **6.3 Arviointi**

Seuraavissa alaluvuissa arvioidaan opinto-oppaan koostamismallia. Mallia vertaillaan ensin tämän tutkielman luvussa 3 kuvattuun Heikkisen (2000) esittelemään dokumenttien koostamismalliin. Arvioinnissa kiinnitetään huomiota myös koostamismallin automaattisuutteen sekä opasversion koostamisessa käytettyjen muunnettujen lähdedokumenttien raekokoon. Lisäksi opintooppaan koostamismallista pyritään lisäksi nostamaan esille havaittuja heikkouksia ja vahvuuksia yleisemminkin.

#### **6.3.1 Koostamismallien vertailu**

Opinto-oppaan koostamismalli poikkeaa Heikkisen (2000) esittelemästä dokumenttien koostamismallista erityisesti siinä, että opinto-oppaan tapauksessa lähdesisältöjä koottiin hankintavaiheessa useista erilaisista lähteistä ja tietovarastoista osin automaattisesti ja osin manuaalisesti. Heikkisen (2000) mallissa oli valmiiksi olemassa dokumenttikokoelma, jossa lähdedokumentit olivat rakenteisessa muodossa. Opinto-oppaan koostamismallissa ei ollut Heikkisen (2000) mallin tapaista alustusvaihetta, jossa lähdemateriaali olisi pilkottu katkelmiksi ja edelleen ryhmitelty katkelmakokoelmiksi automaattisesti. Opintooppaiden lähdemateriaali koostui kokonaisista XML-dokumenteista, eikä nii-

den palasista eli katkelmista kuten Heikkisen (2000) mallissa. Lisäksi opinto-oppaan tapauksessa lähdemateriaalia ei koostamisvaiheessa muunnettu Heikkisen (2000) tapaan yleisen DTD:n mukaiseksi koostedokumentiksi, vaan opinto-oppaan merkkauksessa käytettiin luku.dtd:tä.

Heikkisen (2000) koostamismallin toiminta oli myös automatisoitu pidemmälle. Heikkisen (2000) mallissa koko koostamisprosessia voitiin hallita käyttöliittymän avulla monipuolisesti ja käyttäjäystävällisesti, kun taas opinto-oppaan koostaminen vaati koostajalta manuaalista kirjoitustyötä ja tietämystä XML-kielestä. Opinto-oppaan hallintakäyttöliittymän avulla voitiin avustaa käyttäjää opinto-oppaan koostamisprosessin teknisimmissä toimenpiteissä, mutta koostamisprosessin muokkaaminen esimerkiksi opintojaksokuvausten haun osalta vaati käyttäjältä kokemusta jopa komentoriviohjelmoinnista. Opinto-oppaan julkaisujärjestelmän käytettävyys jätti täten toivomisen varaa.

Opinto-oppaan koostamismallissa huomio kohdistuu myös manuaalisen työn määrään. Heikkisen (2000) mallissa manuaalista työtä vaati ainoastaan koostamisprosessin ohjaus käyttöliittymän välityksellä. Heikkisen (2000) mallissa lähdesisällöt olivat saatavilla valmiissa dokumenttikokoelmassa, eikä niitä tarvinnut koota erikseen erilaisista sisältölähteistä tai tietovarastoista. Opinto-oppaan tapauksessa suuri osa lähdesisällöistä jouduttiin ensin kokoamaan manuaalisesti eri sisältölähteistä, muuntamaan XML-muotoon ja tallentamaan oikeaan hakemistoon WWW-palvelimella. Manuaalisen työn runsautta opinto-oppaan koostamismallissa voidaan pitää puutteena, sillä manuaalisen työn lisääntyessä työhön kulutetun ajan määrä ja alttius tehdä inhimillisiä virheitä kasvaa (Bhavani & John 1997). Lisäksi koostamismallien välillä oli erilaisia tarpeita lähdesisältöjen ajantasaisuuden suhteen: opinto-oppaan tapauksessa opintojaksokuvaukset haettiin Korpista viikottain, jotta tiedot oppaassa olivat koko ajan mahdollisimman ajan tasaisia. Tällaista määrääjain suoritettavaa lähdesisältöjen automaattista päivitystoimintoa ei ollut Heikkisen (2000) koostamismallissa.

Valittaessa lähdedokumentteja koostettavaan opinto-opasversioon käyttäjän täytyi osata muokata XML-dokumenttia, tietää mitä lähdedokumentteja WWW-palvelimelle oli tallennettu ja mihin, sekä valita sieltä oikeat lähdedokumentit. Heikkisen (2000) mallin graafinen käyttöliittymä tarjosi kattavat hakutoiminnot ja käytettävämmän tavan koostettavien lähdedokumenttien valintaan. Valittujen lähdedokumenttien koostaminen tulosdokumentiksi oli molemmissa malleissa varsin automaattista ja samankaltaista, samoin erilaisten julkaisujen tuottaminen. Molemmissa malleissa järjestelmä huolehti automaattisesti tulosdokumentin koostamisesta ja julkaisujen tuottamisesta. Opinto-oppaan tapauksessa lähdedokumentteihin jouduttiin kuitenkin tekemään muutamia manuaalisia hienosäätöjä ennen opasversioiden koostamista. Esimerkiksi joissakin tapauksissa opasversion PDF-julkaisua varten jouduttiin tekemään pakotettuja sivunvaihtoja, jotta tekstit saatiin sijoitettua täsmällisesti halutulla tavalla painoversiota ajatellen. Tällaisia tapauksia olivat esimerkiksi pitkät, useammalle sivulle ulottuvat taulukot, joiden sijainti haluttiin määrittää absoluuttisesti. Pakotettu sivunvaihto tehtiin merkkaamalla haluttuun kohtaan tyhjä sivunvaihtoa tarkoittava elementtitunniste, jonka kohdalle tultaessa julkaisujärjestelmä osasi vaihtaa sivua automaattisesti. Tätä varten jouduttiin kuitenkin käytännössä ensin julkaisemaan koevedos oppaan PDF-versiosta, jotta nähtiin, mihin kohtaan pakotettuja sivunvaihtoja tarvittiin.

### 6.3.2 Koostamisprosessin automatisointi

Vaikka opinto-oppaan koostaminen vaati paljon manuaalisia toimenpiteitä, saatiin tärkeitä toimenpiteitä myös automatisoitua. Opintojaksokuvausten siirron automatisointi oli erityisen tärkeä ominaisuus, sillä tiedot olivat peräisin toisesta tietojärjestelmästä, jonka päivittämisestä vastasivat eri henkilöt kuin opinto-oppaan työryhmän jäsenet. Oppaan sivuista noin  $\frac{1}{4}$  koostui opintojaksokuvauksista, joiden käsittely oli ennen kehittämishanketta vaatinut paljon manuaalista työtä. Julkaisujärjestelmä huolehti tämän suurehkon informaatiomäärän

käsittelystä automaattisesti, joka oli pois manuaalisen työn taakasta verrattuna opinto-oppaan koostamiseen ennen kehittämishanketta.

Integraatio Korppi-järjestelmän ja opinto-oppaan välillä saatiin automatisoitua XML:n avulla ongelmitta, lukuun ottamatta joitakin Korppi-järjestelmän tuottamia virheellisiä sisältöjä. Liikuntakurssin tapauksessa jouduttiin opintojaksojen järjestämisestä vastaavaa XSLT-sovellusta varta vasten muokkaamaan, jotta kyseinen opintojakso saatiin tulostumaan toiseen paikkaan. IT-tiedekunnan määrittelemiin opintojaksojen koodeihin perustuva lajittelu tulosti liikuntakurssin paikkaan, johon sitä ei oppaan kokonaisuuden kannalta haluttu. Täten jouduttiin tekemään toimenpiteitä ainoastaan yhden tapauksen vuoksi. Tämäkin seikka voidaan katsoa kokonaisuuden kannalta pieneksi, joskin tarkkuutta vaativaksi toimenpiteeksi.

Joitakin opintojaksoja jouduttiin siistimään, sillä Korppi-tulosteessa opintojaksokuvausten sekaan syötetyt HTML-elementtitunnisteet tulostuivat virheellisinä. Siistimistoimenpide toteutettiin ohjelmallisesti merkkijonokäsittelykieltä käyttäen. Toimenpide oli sinänsä automaattinen, mutta se loi tarpeen käsitellä ohjelmallisesti Korpista saatuja XML-dokumentteja tapauskohtaisella tavalla, joten se voidaan laskea ongelmaksi integraatiossa.

Voidaan todeta, että automaattisesti tuotettujen XML-lähdedokumenttien manuaalisen muokkaamisen tarve oli vähäistä. Yleisesti ottaen Korppiin rakennettu rajapinta helpotti suuresti integraatiota Korpin ja opinto-oppaan välillä, nimenomaan XML:n ansiosta.

Lähdedokumenttien manuaalisen valintamäärittelyn jälkeen oppaan julkaisu-järjestelmä kykeni huolehtimaan lähdedokumenttien koostamisesta ja monikanavajulkaisusta täysin automaattisesti, mikä helpotti huomattavasti oppaan laadintaprosessia. Enää ei ollut tarvetta kopioida tietoja Word-ohjelman näytöltä toiselle. Lähdemateriaali oli nyt jaettuna pienempiin osiin verrattuna alkutilaan, jossa kaikki lähdemateriaali oli yhdessä Word-tiedostossa. Tämä helpotti

erityisesti lähdemateriaalin ylläpitoa ja uusien opasversioiden koostamista lähdedokumenttien päivittyessä. Julkaisujärjestelmän WWW-selaimella käytettävä hallintakäyttöliittymä helpotti myös opasversioiden hallintaa niiden työstövaiheessa. Käyttäjä saattoi hallita WWW-selaimen avulla opasversion koostamiseen ja julkaisuun liittyviä automatisoituja toimenpiteitä WWW:stä tutulla toimintalogiikalla.

Kaikkea olennaista ei kuitenkaan saatu automatisoitua, josta tärkeänä esimerkkinä henkilötietojen siirron automatisointi. Henkilötietojen siirtoa ei toteutettu ajanpuutteen vuoksi. Kaiken kaikkiaan opinto-oppaan julkaisujärjestelmää voitiin kutsua puoliautomaattiseksi dokumenttien koostamisjärjestelmäksi.

### 6.3.3 Lähdedokumentit ja niiden raekoko

Viimeinen arvioitava huomio kiinnittyy koostettaviin lähdedokumentteihin ja etenkin niiden raekokoon. ”*Raekoolla*, eli *granulariteetilla* tarkoitetaan digitaalisen tiedon yhteydessä hallittavien tietoyksikköjen kokoa” (Tyrväinen 2005, 22). Tässä tutkielmassa raekoolla tarkoitetaan sitä, kuinka pieniin osiin XML-muotoon muunnetut lähdedokumentit tulisi pilkkoa, ja miksi, jotta raekoko olisi tarkoituksenmukainen. XML-kieli mahdollistaa kunkin elementtitunnisteen ohjelmallisen käsittelyn. Tämä puolestaan mahdollistaa kyseisten sisältöpalasten tehokkaan uudelleen käytön uusissa dokumenteissa. Tätä ominaisuutta ei hyödynnetty opinto-oppaan koostamismallissa. Lähdedokumenttien yhtä kokonaista sisältölukua opasversiossa vastaava raekoko ja julkaisujärjestelmän tapa koostaa opasversiot eivät mahdollistaneet lähdedokumenttien pilkkomista pienemmiksi ja tehokkaammin uudelleen käytettäväksi katkelmiksi Heikkisen (2000) mallin tapaan.

Opinto-oppaan lähdedokumenttien raekoko aiheutti myös toistaiseen tiedon tallentamiseen liittyviä ongelmia: jotkut lähdedokumentit olivat suurimmilta osiltaan sopivia käytettäväksi molemmissa opasversioissa, mutta saattoivat sisältää



joitain keskenään erilaisia sisältöjä. Yhdenkin kappaleen olennainen poikkeavuus saattoi estää saman lähdedokumentin uudelleenkäytön molemmissa opasversioissa. Esimerkiksi tiedekunnan tutkintoihin liittyvät kuvaukset oli tallennettu samaan lähdedokumenttiin, jota haluttiin uudelleenkäyttää molemmissa opasversioissa. Ensimmäisen vuoden opiskelijoille tarkoitettuun oppaaseen ei kuitenkaan haluttu tutkintoihin kuuluvia syventäviä opintoja koskevia osuuksia, kun taas vanhempien opiskelijoiden oppaaseen ne ehdottomasti haluttiin. Täten tutkintokuvaukset jouduttiin tallentamaan erillisiin XML-tiedostoihin suppeampana ja laajempaan versioina. Lähdedokumenttien sisällöt olivat suppeammalta osalta identtiset. Molemmissa opasversioissa käytettiin samoja tekstisisältöjä, mutta vanhempien opiskelijoiden opasversioon haluttiin lisäksi tietoa tutkintojen syventävistä opinnoista. Julkaisujärjestelmän toteutuksen vaatima yhden kokonaisen sisältöluvun raekoko ei mahdollistanut tutkintokuvausten osittaista uudelleenkäyttöä molemmissa opasversioissa, ja näin jouduttiin tallentamaan samaa informaatiota kahteen eri lähdedokumenttiin.

Opinto-oppaan koostamismallissa lähdedokumentteja ei pilkottu katkelmiksi, eikä lähdedokumentteja ollut mahdollista hakea Heikkisen (2000) mallin mukaisesti dokumenttikokoelmasta käyttöliittymän avulla. Koostettavat lähdedokumentit oli määriteltävä kirjoittamalla niiden tiedostonimet erilliseen XML-muotoiseen koostemäärittystiedostoon. Heikkisen (2000) mallissa järjestelmä huolehti automaattisesti katkelmien ryhmittelystä sisällön mukaan, ja tarjosi kattavat hakumahdollisuudet katkelmien etsimiseksi graafisen käyttöliittymän avulla. Opinto-oppaan tapauksessa käytetyn luku.dtd:n hyvin ulkoasuun ja muotoiluun keskittyneet merkkauksmahdollisuudet eivät tarjonneet riittävän tarkkaa merkkaustapaa, jolla lähdedokumentit olisi voitu merkata niiden sisältöä paremmin kuvaavammalla tavalla.

XooZoo-projektin aikana laaditun meta.dtd-määrittelyn avulla voitiin lähdedokumentteihin liittää metatietoja, kuten kommentteja, tekijätietoja, versionumero ja dokumentin tila. Meta.dtd:n käyttö opasversioiden tuotannossa oli kui-

tenkin lähinnä nimellistä. Sisältöpalasilla ei täten ollut olemassa juuri minkäänlaisia metatietoja, joiden perusteella niitä olisi voitu hakea ohjelmallisesti Heikkisen (2000) mallin tapaan. Koostaessa käyttäjän opasversiota oli tiedettävä tarkkaan mitä lähdedokumentti sisälsi. Toisaalta, koska Heikkisen (2000) mallin kaltaista toimintoa sisältöpalasten hakuun ei ollut käytössä, lähdedokumenttien pienemmällä raekoolla ja lähdedokumenttien kattavilla metatiedoilla ei olisi ollut edes suurempaa merkitystä. Lähdedokumentteja olisi pienemmänkin raekoon tapauksessa jouduttu määrittelemään manuaalisesti koosteeseen. Tässä tapauksessa myös manuaalisesti hallittavien lähdedokumenttien lukumäärä olisi kasvanut, joka puolestaan olisi vaikeuttanut hakemistorakenteeseen perustuvaa lähdedokumenttien hallintaa. Voidaankin vetää johtopäätös, että koostettaessa XML-dokumentteja ei lähdedokumenteille ole olemassa yhtä optimaalista raekokoa, vaan raekoon määrittely tulisi tehdä tapauskohtaisesti.

Opinto-oppaan tapauksessa lähdedokumentit olivat lukumääränsä ja raekonsa puolesta kohtalaisen helposti hallittavissa. Lisäksi asiaa auttoi myös se, että suurin osa lähdesisällöistä oli muunnettu XML-muotoon manuaalisesti, jolloin lähdesisällöt olivat käyneet teknisille tukihenkilöille tutuiksi. Ulkopuolisen henkilön olisi ollut hankalaa koostaa opasversio tutustumatta ensin kunkin lähdedokumentin sisältöön. Dokumenttien hallinnan sijaan oppaan laadinnassa oli keskitytty enemmänkin sisällönhallintaan. Dokumenttien ohjelmallinen hallinta onkin erityinen jatkokehityskohde opinto-oppaan koostamiseksi, etenkin jos useammat sisällöntuottajat alkavat tulevaisuudessa tuottaa tekstejä XML-muodossa. Tehokkaampaa olisi, mikäli sisällöntuottajat päivittäisivät muutokset itse suoraan lähdedokumentteihin.

#### 6.4 Käytännön kokemuksia opinto-oppaan kehittämishankkeesta

Seuraavaksi esitetään joitain kehittämishankkeen aikana havaittuja käytännön kokemuksia. Kokemukset eivät liity varsinaisesti dokumenttien koostamiseen, vaan yleisemmin rakenteisten dokumenttien parissa työskentelyyn.

Opinto-oppaan kehittämishanke oli kohtalaisen laaja projekti: se kesti lähes vuoden, ja siihen osallistui tiiviisti noin kymmenen ihmistä ja vähintään saman verran ihmisiä sisällöntuotannossa. On syytä huomata, että XooZoo-projektin alkaessa kehittäjinä toimi henkilöitä, joilla ei ollut juurikaan käytännön kokemusta rakenteisten dokumenttien kanssa työskentelystä, vaan tietämys kasvoi projektin edetessä. Kyseessä oli täten monelta osin myös oppimisprojekti. Tehtävää oli paljon, eikä projektin kaikissa osa-alueissa ollut edes suoranaisesti kyse rakenteisista dokumenteista. XooZoo-projektin aikana saatiin silti tehtyä kattava pohjatyö, jonka tuloksiin projektin toimeksiantaja oli tyytyväinen, ja jota voitiin lähteä kehittämään eteenpäin Xoo-jatko -projektissa. Xoo-jatko -projekti oli toteutuspainotteisempi XooZoo-projektiin verrattuna. Sen aikana toteutettiin opinto-oppaalle uusi XML:ään perustuva julkaisujärjestelmä ja muunnettiin lähdemateriaali XML-muotoon.

Kehittämishankkeen teki mielenkiintoiseksi myös se, että hanke toteutettiin IT-tiedekunnan ja sen kahden ainelaitoksen yhteistyönä, ja siihen osallistui monia ihmisiä erilaisin teknologisin taustoin. XML-kieli ei ollut kaikille tuttu teknologia. Näin päästiin havainnoimaan XML:n käyttöönottoa uutena teknologiana kohdeorganisaatiossa. Lisäksi XooZoo-projektin aikana havaittiin, kuinka hankalaa XML:n hyödyllisyyttä ja etuja on demonstroida etenkin hankkeen alkupuolella. Projektin alussa kohdealueen tarpeet täytyy tutkia tarkoin esimerkiksi keskustelemalla toimeksiantajan kanssa, ennen kuin XML-määrityksiä voidaan alkaa suunnitella.

Kehittämishankkeen aikana saattoi myös havaita, kuinka rakenteisten dokumenttien parissa työskennellessä ihmisillä saattaa olla hyvinkin erilaisia näke-

myksiä esimerkiksi merkkaukäytännöistä. Eräs työryhmän jäsen muun muassa totesi sähköpostin välityksellä käydyssä keskustelussa:

”on kieltämättä mielenkiintoista nähdä, kuinka monella tavalla tietyn tekstirakenteen voi tulkita. Ei ihme, että skeemasuunnitteluprojektit ovat niin työläitä.”

Opinto-oppaan kehittämishankkeen aikana saatujen kokemusten myötä voidaan myös todeta DTD-suunnittelun olevan jatkuva prosessi. Projektin edetessä vaatimukset tarkentuvat ja uusia vaatimuksia syntyy, jotka vaativat muutoksia myös DTD-määrittelyyn iteratiivisesti.

Sisällöntuotannon kehittämisen kannalta ei tässä hankkeessa vielä päästy tärkeään tavoitteeseen, jossa kaikki sisältö tuotettaisiin suoraan XML-muotoon. Työryhmän jäsenten kiireellisten aikataulujen ja hankkeen rajallisten resurssien vuoksi sisällöntuotannon suurempi muutos jäi tulevaisuuden jatkokehitykseen. Tapahtuessaan se tulee vaatimaan ominaisuuksia ja resursseja esimerkiksi käytännön työkaluilta, dokumenttien hallinnan suunnittelulta, julkaisujärjestelmän jatkokehitykseltä, lähdedokumenttien raekoon uudelleen suunnittelulta ja työryhmän jäsenten koulutukselta XML:n osalta.

Yleisesti ottaen opinto-oppaan kehittämishankkeen tuloksiin oltiin tyytyväisiä niin työryhmän sisällä kuin IT-tiedekunnassakin ja positiivista palautetta etenkin oppaan sisällöstä ja verkkojulkaisun ulkoasusta saatiin myös Jyväskylän yliopiston hallinnolta.

## 7 YHTEENVETO

Tässä tutkielmassa tarkasteltiin XML-dokumenttien koostamista. Dokumenttien koostamisella tarkoitetaan uusien dokumenttien koostamista olemassa olevista tai uusista lähdedokumenteista tai niiden osista esimerkiksi erilaisia julkaisuja varten. XML-dokumenttien koostamisella tarkoitetaan lähdemateriaalin koostamista ja muuntamista XML-muotoon. Lähdemateriaali voi olla myös jo valmiiksi XML-muodossa, mikä helpottaa koostamisprosessia: merkkauksen ansiosta päästään käsiksi kokonaista dokumenttia pienempiin sisältöosiin ja niiden ohjelmallinen käsittely helpottuu. Mikäli lähdedokumentit eivät ole valmiiksi XML-muodossa, ne täytyy muuntaa XML-muotoon joko automaattisesti tai manuaalisesti.

Tutkielma oli tapaustutkimus, jossa tarkasteltiin Jyväskylän yliopiston informaatioteknologian tiedekunnan XML-muotoisen opinto-oppaan koostamismallia. Koostamismalli syntyi vuosien 2004-2005 aikana toteutetun opinto-oppaan kehittämishankkeen tuloksena, ja sitä kokeiltiin käytännössä koostettaessa luvun 2005-2006 opinto-oppaita XML-muotoon muunnetuista lähdedokumenteista, jotka olivat peräisin erilaisista sisältölähteistä.

Tapaustutkimuksen kohdealue kuvattiin luvussa 4 ja kehittämishanketta analysoitiin dokumenttien hallinnan alueelle suunnitellun viitekehyksen avulla luvussa 5. Tutkielman teoriaosuus käsitti Heikkisen (2000) väitöskirjassa esitellyn XML-dokumenttien koostamismallin esittelyn luvussa 3, johon opinto-oppaan koostamismallia verrattiin luvussa 6. Tutkielmassa esiteltiin lisäksi rakenteisten dokumenttien ja dokumenttien hallinnan käsitteet sekä XML-kielen perusteet luvussa 2.

Tutkimuksen päätavoitteena oli opinto-oppaan koostamismallin vertailu aiemmassa tutkimuksessa esiteltyyn koostamismalliin sekä esittää havaintoja mahdollisista uusista XML-dokumenttien koostamiseen liittyvistä käytännön

ilmiöistä. Tutkielman muita tavoitteita olivat XML-dokumenttien koostamisen ja sisältöjen uudelleenkäytön käsitteiden selvittäminen ja kehittämishankkeen aikana havaituista käytännön kokemuksista raportointi.

Vertailtujen koostamismallien erona oli koostamistoimintojen automaattisuus: opinto-oppaan koostaminen vaati runsaasti manuaalista kirjoitustyötä ja jopa ohjelmointitaitoja, kun taas Heikkisen (2000) esittelemän koostamismallin toiminnot olivat pidemmälle automatisoituja ja käyttäjäystävällisempiä sekä kokonaan koostajan ohjattavissa käyttöliittymän avulla. Molempien mallien perustoiminta ja lopputulokset eli koostetut dokumentit olivat kuitenkin hyvin samankaltaisia.

Tutkielman tulosten perusteella lähdedokumenttien raekoko on syytä huomioida XML-dokumenttien koostamisessa: usein on tarpeen uudelleenkäyttää vain osa dokumentista, joten kyseinen osa tulisi voida selkeästi erottaa muusta dokumenttisisällöstä. Opinto-oppaan tapauksessa lähdedokumentteja ei pilkottu katkelmiksi Heikkisen (2000) mallin mukaan, vaan koostamisessa käytettiin kokonaisia lähdedokumentteja. Tämä johti joissakin tapauksissa toisteisen tiedon tallentamiseen, mutta toisaalta taas raekoon ansiosta lähdedokumenttien hallinta oli käytettävissä olevien työkalujen avulla helppoa. Tutkielmassa havaittiin myös, ettei ole olemassa yhtä absoluuttista raekokoa, joka soveltuisi kaikkiin mahdollisiin koostamistilanteisiin. Lähdedokumenttien raekoko tulisi aina määritellä tapauskohtaisten tarpeiden mukaan koostettaessa XML-dokumentteja.

Koostamismallien lähdemateriaaleilla havaittiin eroavaisuuksia: opinto-oppaan tapauksessa osaa lähdemateriaalista oli tarve päivittää automaattisesti määräajoin. Lisäksi opinto-oppaan lähdemateriaali koottiin erimuotoisina erilaisista sisältölähteistä, kun taas Heikkisen (2000) mallissa oli olemassa valmis dokumenttikokoelma, jossa kaikki lähdemateriaali oli rakenteisessa muodossa. Heikkisen (2000) mallissa lähdedokumentit eivät kuitenkaan noudattaneet yh-

tenäistä rakennemäärittystä, jonka johdosta lähdedokumentit ensin pilkottiin katkelmiksi ja luokiteltiin sisällön mukaan jonka jälkeen koosteeseen valitut katkelmat muunnettiin yleisen DTD:n mukaiseksi tulosedokumentiksi.

Tutkielmassa havaittiin myös, että XML-teknologia helpotti tapaustutkimuksessa kuvattua kahden tietojärjestelmän integraatiota etenkin XSLT-kielellä toteutettujen XML-dokumenttien automaattisten muunnosten ansiosta. Opintojaksokuvaukset saatiin Korpista valmiiksi XML-muodossa, jolloin ne oli helppo muuntaa automaattisesti XSLT:n avulla opinto-oppaalle sopivaan muotoon.

Kehittämishankkeen aikana saatujen käytännön kokemusten perusteella voidaan todeta rakenteisiin dokumentteihin liittyvien kehittämishankkeiden vaativan paljon määrittelytyötä ja keskustelua toimeksiantajan kanssa, ennen kuin konkreettisia tuloksia voidaan odottaa. Pelkästään DTD-määrittysten suunnittelu vaatii tarkkaa kohdealueen tarpeisiin ja elektronisten dokumenttien vaatimukseen kohdistuvaa tutkimustyötä koko hankkeen ajan.

Tutkimusmateriaalina käytettiin opinto-oppaan kehittämishankkeen aikana syntyneitä dokumentteja ja käytännön sovelluksia joita analysoitiin kvalitatiivisesti viitekehyksen avulla. Pääosin tietoa kerättiin kuitenkin havainnoimalla käytännön ilmiöitä opinto-oppaan koostamisessa koko kehittämishankkeen ajan. Tutkielman kirjoittajan osallistuminen kehittämishankkeeseen vaikutti luonnollisesti tapaustutkimuksen analyysiin. Kirjoittaja pyrki olemaan mahdollisimman objektiivinen, mutta tiivis mukanaolo kehittämishankkeessa ei voi olla vaikuttamatta myös havaintojen kuvailuun. Kirjoittaja oli itse tekemässä ratkaisuja kehittämishankkeen aikana, jolloin hän ei välttämättä ole paras mahdollinen henkilö arvioimaan toteutuksia mahdollisimman objektiivisesti. Kehittämishankkeen ulkopuolinen henkilö olisi mahdollisesti kiinnittänyt huomioita asioihin, joihin kirjoittaja ei ole ottanut kantaa. Tämä on syytä huomata etenkin tutkimustulosten yleistettävyyttä ja luotettavuutta pohdittaessa. Lisäksi opinto-oppaan koostamismalli kehitettiin huomioiden ainoastaan yhden tiedekunnan

tarpeet etenkin DTD-suunnittelussa. Koostamisessa käytetyt teknologiat kuten XML ja XSLT ovat kuitenkin yleisesti monikanavajulkaisuissa ja vastaavissa hankkeissa käytettyjä teknologioita, joten niiden hyödyntämisestä saatuja tuloksia ja havaintoja voidaan pitää yleistettävänä.

Jatkotutkimusaiheena opinto-oppaan lähdedokumenttien raekokoa voisi yrittää kehittää siten, että lähdesisältöjä voitaisiin uudelleenkäyttää tehokkaammin ja välttää tutkielmassa todettu toisteisen tiedon tallentamisen ongelma. Lähdedokumenttien pilkkomista katkelmiksi sopivalla raekoolla Heikkisen (2000) mallin tapaan voisi myös pohtia. Lisäksi opinto-oppaan koostamismallin toimintaa voitaisiin automatisoida pidemmälle ja käyttöliittymää kehittää käyttäjäystävällisempään suuntaan, joiden käytännön toteutuksia ja kannattavuutta voitaisiin miettiä jatkotutkimuksessa.



## LÄHDELUETTELO

- Adler S., Berglund A., Caruso J. ym. (toim.). 2001. Extensible Stylesheet Language (XSL) Versio 1.0 [online]. W3-järjestön suositus 15.10.2001. [Viitattu 2.12.2005]. Saatavilla WWW-muodossa <URL:<http://www.w3.org/TR/xsl/>>.
- Alcorn W., Kaplan K. & Tucker J. 2004. Development of XML industry standards for information exchange and commerce (electronics recycling industry example). Teoksessa D. Dickinson, S.H. Matthews & P. Muscanelli (toim.) Proceedings of the 2004 IEEE International Symposium on Electronics & the Environment Scottsdale, AZ Usa, May 10-13. IEEE Computer Society, 281-286.
- Altova 2005. Altova XMLSpy 2005 [online]. Altova, Inc [viitattu 10.11.2005]. Saatavilla [www-osoitteessa](http://www.altova.com/products_ide.html) <[http://www.altova.com/products\\_ide.html](http://www.altova.com/products_ide.html)>.
- Antila J. A. & Pohjonen L. S. 2001. XML yrityksen sovellusten integroinnissa. Jyväskylän yliopisto, Tietojärjestelmätieteen pro gradu -tutkielma.
- Anttila J. 2001. Dokumenttien hallinta. Helsinki: Oy Edita Ab
- Ashman H. 2000. Electronic document addressing: dealing with change. ACM computing surveys (CSUR) 32(3), 201-212.
- Bartocci E., Mariani L. & Merelli E. 2003. An XML view of the world. Teoksessa O. Camp, J. Filipe, S. Hammoudi & M. Piattini (toim.) Proceedings of the 5th international conference on enterprise information systems angers, France, April 23-26. ICEIS Press, 19-27. Saatavilla WWW-osoitteessa <<http://www.bioagent.net/WWWPublications/Download/aixo03.pdf>>.

- Bhavnani S. K. & John B. E. 1997. From sufficient to efficient usage: an analysis of strategic knowledge. Teoksessa S. Pemberton (toim.) Proceedings of the SIGCHI conference on Human factors in computing systems 1997 Atlanta, Georgia, United States, March 22-27. New York, NY, USA: ACM Press, 91-98.
- Boiko B. 2002. Content management bible. New York, U.S.A: Hungry Minds, Inc.
- Bray T., Paoli J., Sperberg-McQueen C.M., Maler E., Yergeau F. & Cowan J. (toim.). 2004. Extensible Markup Language (XML) 1.1. [online] W3-järjestön suositus 15.4.2004. [viitattu 19.4.2004]. Saatavilla WWW-muodossa <<http://www.w3.org/TR/xml11/>>.
- Brown A. 2003. XML in serial publishing: past, present and future. OCLC Systems & Services 19(4), 149-154.
- Clark J. (toim.). 1999. XSL Transformations (XSLT) Versio 1.0 [online]. W3-järjestön suositus 16.11.1999. [Viitattu 10.5.2005]. Saatavilla WWW-muodossa <<http://www.w3.org/TR/xslt>>.
- Dunn M. 2003. Single-source publishing with XML. IT Professional 5(1), 51-54.
- Eclipse 2005. Eclipse.org Main Page [online]. Eclipse foundation [viitattu 10.11.2005]. Saatavilla www-osoitteessa <<http://www.eclipse.org/>>.
- Fallside D. C., Walmsley P. (toim.). 2004. XML Schema osa 0 [online]. W3-järjestön suositus 28.10.2004. [Viitattu 2.12.2005]. Saatavilla WWW-muodossa <<http://www.w3.org/TR/2004/REC-xmlschema-0-20041028/>> .
- Eläketurvakeskus 2005. Eläketurvakeskus [online]. Suomen eläketurvakeskus [viitattu 10.11.2005]. Saatavilla www-osoitteessa <<http://www.etk.fi/>>.

Heikkilä J., Hiltunen L., Hämäläinen T., Kyppö J., Kärkkäinen T, Mäkinen S., Puuronen S., Saariluoma P., Sakkinen M., Salminen A., Savela J. & Veijalainen J. 2005. Jyväskylän yliopiston informaatioteknologian tiedekunnan toimintakertomus 2004. Lievestuore: ER-paino Oy. Saatavilla WWW-osoitteessa

<<http://www.it.jyu.fi/Henkilostolle/Toimintakertomus.pdf>>

Heikkinen B. 2000. Generalization of document structures and document assembly. PhD Thesis, University of Helsinki Department of computer science, Report A-2000-2.

Hirsjärvi S., Remes P. & Sajavaara P. 2000. Tutki ja kirjoita. Helsinki: Kirjayhtymä.

Honkaranta A. 2003. From genres to content analysis. Experiences from four case organizations. Jyväskylä Studies in Computing, 31. University of Jyväskylä. Ph.D. Thesis.

Honkaranta A., Salminen A., Peltola T. 2005. Challenges in the redesign of content management: A case of FCP. International journal of cases on electronic commerce, 1 (1), 53-69.

Honkaranta A. & Tyrväinen P. 2001. Possibilities and constraints for managing and reusing information content of structured documents: The case of operation and maintenance manuals. Teoksessa A. Mørch, A. L. Opdahl, B. Solveig, & R. E. Moe (toim.) Proceedings of information systems research seminar in Scandinavia (IRIS) 24 conference Bergen, Norway, August 11-14. University of Bergen, Department of Information Science, 381-396.

Honkaranta A. & Tyrväinen P. 2005. Content management in organizations. Teoksessa M. Khosrowpour (toim.), Encyclopedia of information science and technology. Hershey, U.S.A.: Idea Group Publishing, Inc., 550-555

- Jenssen A. E. & Sandahl T. I. 1996. Conflicts between the possibilities and the reality in the field of structured electronic documents: Experiences from a largescale SGML-project. Teoksessa Bo Dahlbom ym. (toim.) Proceedings of the 19th Information systems Research seminar In Scandinavia, Lökeberg (Göteborg), August 10-13. 935-955.
- Jyväskylän yliopisto 2005. Jyväskylän yliopiston henkilöhaku [online]. Jyväskylän yliopisto [viitattu 25.11.2005]. Saatavilla [www-osoitteessa <http://www.cc.jyu.fi/numerohaku/>](http://www.cc.jyu.fi/numerohaku/).
- Korhonen R. 2003. Rakenteisten liiketoimintadokumenttien visualisointi. XSL:n soveltaminen organisaatioiden välisiin XML-dokumentteihin. Jyväskylän yliopisto, Tietojärjestelmätieteen pro gradu -tutkielma.
- Korppi 2005. Korppi-järjestelmä [online]. Jyväskylän yliopisto [viitattu 10.11.2005]. Saatavilla [www-osoitteessa <https://korppi.jyu.fi/kotka/help/tietoja.jsp>](https://korppi.jyu.fi/kotka/help/tietoja.jsp).
- Koulopoulos T.M., Frappaolo C. 1995. Electronic document management systems. A Portable Consultant. McGraw-Hill, 1995.
- LaTeX 2005. LaTeX project: LaTeX - A document preparation system [online]. LaTeX project [viitattu 22.11.2005]. Saatavilla [www-osoitteessa <http://www.latex-project.org/contact.html>](http://www.latex-project.org/contact.html).
- Lehtonen M., Petit R., Heinonen O. & Lindén G. 2002. A dynamic user interface for document assembly. Teoksessa E. Munson, R. Furuta & J.I. Maletic (toim.) Proceedings of the 2002 ACM symposium on Document engineering McLean, Virginia, USA, November 8-9. New York, NY: ACM Press, 134-141.

- Lehtonen M. 2001. Semi-automatic document assembly with structured source data. University of Helsinki Department of Computer Science, Master's Thesis.
- Levy D. M. 1993. Document reuse and document systems. *Electronic publishing - origination, dissemination and design*, 6 (4), 339-348.
- Li H. 2000. XML and industrial standards for electronic commerce. *Knowledge and information systems* 2(4), 487-497.
- Lyytikäinen V. 2004. Contextual and structural metadata in enterprise document management. PhD thesis, University of Jyväskylä.
- McIntosh M. 2000. Content management using the rational unified process®. Rational software white paper. Cupertino, CA: Rational software corporation.
- Microsoft 2005. Microsoft Office Online: Word 2003 Home Page [online]. Microsoft corporation [viitattu 26.11.2005]. Saatavilla [www-osoitteessa <http://office.microsoft.com/en-us/FX010857991033.aspx>](http://office.microsoft.com/en-us/FX010857991033.aspx).
- Opinto-opas 2005. Informaatioteknologian tiedekunta - Opinto-oppaat 2005-2006 [online]. Jyväskylän yliopisto [viitattu 10.11.2005]. Saatavilla [www-osoitteessa <http://opinto-opas.it.jyu.fi/2005-2006/>](http://opinto-opas.it.jyu.fi/2005-2006/).
- Pemberton S., Austin D., Axelsson J. ym. (toim.). 2000. The Extensible HyperText Markup Language (Second Edition) A Reformulation of HTML 4 in XML 1.0 [online]. W3-järjestön suositus 26.1.2000. [Viitattu 27.10.2005]. Saatavilla WWW-muodossa [<http://www.w3.org/TR/xhtml1/>](http://www.w3.org/TR/xhtml1/).
- Päivärinta T. & Munkvold B.E. 2005. Enterprise content management: an integrated perspective on information management. Teoksessa *System sciences (toim.) Proceedings of the 38th Annual Hawaii International*

Conference on System Sciences Big Island, Hawaii, January 3-6. Washington, DC, USA: IEEE Computer Society, 96-99.

Rockley A. 2002. Managing enterprise content: a unified content strategy. New Rider Publishing.

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>>.

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. 2003b. Towards digital government by XML standardization: methods and experiences. Teoksessa XML users group Finland (toim.), Proceedings of the XML Finland 2003: Open Standards, XML and the Public Sector, Kuopio, October, 29-30. Espoo: XML Users Group Finland.

Salminen A. 2004a. XML-kieli ITK221 XML-kieli- opintojakson opintojaksomateriaali. Osa 1. Jyväskylä: Jyväskylän yliopisto, tietojenkäsittelytieteiden laitos.

Salminen A. 2004b. XML-kieli. ITK221 XML-kieli- opintojakson opintojaksomateriaali. Osa 2. Jyväskylä: Jyväskylän yliopisto, tietojenkäsittelytieteiden laitos.

- Salminen A. 2004c. XML-kieli. ITK221 XML-kieli- opintojakson opintojaksomateriaali. Osa 3. Jyväskylä: Jyväskylän yliopisto, tietojenkäsittelytieteiden laitos.
- Salminen A. 2005. Building digital government by XML. Teoksessa System Sciences (toim.) Proceedings of the thirty-eighth annual Hawaii international conference on system sciences. Los Alamitos, CA: IEEE Computer Society
- Schamber L. 1996. What is a document? Rethinking the concept of uneasy times. *Journal of the American Society for Information Science* 47 (9), 669-671.
- Suomen virtuaaliyliopisto 2005. Suomen virtuaaliyliopisto / Etusivu 2005 [online]. Suomen virtuaaliyliopisto [viitattu 10.11.2005]. Saatavilla [www-osoitteessa <http://www.virtuaaliyliopisto.fi>](http://www.virtuaaliyliopisto.fi).
- Sutton M. J. D. 1996. Document management for the enterprise: principles, techniques, and applications. New York: Wiley.
- Tyrväinen P. 2005. Organisaation sisältöjen hallinta. ITKD51 Sisällönhallinta organisaatioissa - opintojakson opintojaksomateriaali. Jyväskylä: Jyväskylän yliopisto, tietojenkäsittelytieteiden laitos.
- Vartiainen T. 2005. Moral conflicts in a project course in information systems education. PhD Thesis, University of Jyväskylä, Jyväskylä studies in computing, 49.
- Vaughan-Nichols S. J. 2003. XML raises concerns as it gains prominence. *IEEE Computer* 36(5), 14-16.
- W3C 2005. [online]. World Wide Web Consortium [viitattu 10.11.2005]. Saatavilla [www-osoitteessa <http://www.w3.org/>](http://www.w3.org/).

- Weitzman L. 2004. Meta-design for sensible information. *Interactions* 11(2), 71-73.
- Wilkinson R. 1994. Effective retrieval of structured documents. Teoksessa W. Bruce Croft & C.J. Van Rijsbergen (toim.) *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval* Dublin, Ireland July 03-06. Springer-Verlag New York, Inc. New York, NY, USA, 311-317.
- Yin R. K. 1994. *Case study research: design and methods*. Second Edition. Thousand Oaks, California: Sage Publications.
- XooZoo. 2005a. Laatumisprosessin kartoitus. Vaiheraportti (versio 4.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- XooZoo. 2005b. Opinto-oppaan sisällön ja käytön kartoitus. Raportti - julkinen (versio 1.0). Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos.
- XooZoo 2005c. XooZoo-projekti, Jyväskylän yliopisto, 2004-2005 [online]. Jyväskylän yliopisto [viitattu 10.11.2005]. Saatavilla [www-osoitteessa <http://projekti.it.jyu.fi/2004/xoozoo/>](http://projekti.it.jyu.fi/2004/xoozoo/).



## LIITE 1: XOOZOO-PROJEKTIN TOIMEKSIANTO

1 YIRITYKSEN TIEDOT	<b>Yrityksen nimi:</b> Jyväskylän yliopisto Informaatioteknologian tiedekunta	<b>Yrityksen toimiala:</b> Julkinen hallinto Tutkimus ja koulutus
	<b>Postitoimiosoite:</b> Jyväskylän yliopisto. Informaatioteknologian tiedekunta. PL 35 (AG 2.krs), 40014 Jyväskylän yliopisto, Suomi	
	<b>Muut projektin kannalta merkittävät toimipaikat:</b> Agora Center, Virtuaaliyliopisto, Tietojenkäsittelytieteiden laitos, Tietotekniikan laitos.	
	<b>Yrityksen henkilöstömäärä:</b>  7 (toimisto), n. 1900 opiskelijaa, 200 jatko-opiskelijaa.	<b>Projektiin osallistuvien henkilöiden määrä:</b>  Tiedekunnan toimisto: 4 TKTL+TTL: 2
2 PROJEKTIAIHEEN ALUSTAVAT TIEDOT	<b>Projektityön nimi:</b> XOO (Opinto-Oppaan hallinnan, sisällön, ja rakenteen kehittäminen).	
	<b>Lyhyt kuvaus projektista:</b> <p>Tiedekunnan opinto-opas kootaan vuosittain, ja julkaistaan sekä painettuna, että Internet-sivuilla. Oppaan käyttäjiä ovat muun muassa tiedekunnan omat uudet opiskelijat (perusvalinnan ja muuntokoulutusvalinnan kautta valitut), omat vanhat opiskelijat, oma opettajakunta, oma hallintoväki, sivuaineopiskelijat, yliopiston muu hallintoväki (opet, potentiaaliset uudet opiskelijat), ja niin edelleen. Oppaan sisältö on laaja; se sisältää monenlaista materiaalia eri lähteistä: kurssikuvauksia ja aikatauluja Korpista, tiedekunnan ja laitosten esittelyä, ohjeita, tutkintorakennekuvauksia, henkilökuntaluetteloita, tietoa muiden laitosten tarjonnasta ja yliopiston yhteisestä opinto-ohjauksesta ja niin edelleen. Oppaan sisältötiedot tulevat useista eri lähteistä; laitoksilta, muista yliopistoista, tiedekunnasta, ja Korppi-järjestelmästä. Oppaasta laaditaan lisäksi myös erillinen englanninkielinen versio, jossa paljon samoja tietoja. Oppaan laatimiseen liittyy sekä sisällöllisiä, että teknisiä haasteita.</p> <p>Opinto-oppaan laatimiseen liittyy kaksi keskeistä kehittämistarvetta, joihin projektiryhmältä toivotaan apua. Ensimmäinen ongelma-alue on sisällön jaottelu ja laadintaprosessi: sisältöosien tuottamisen vastuuttaminen, hankinta, tarkastaminen, versiointi, ja hyväksyttäminen moninaisilla sisällön tuottaja- ja hyväksyjätahoilla. Nykykäytäntöjä on tarvetta selvittää ja kehittää. Oppaan sisällön voisi modularisoida (mahdollisesti määritellä rakenteeseen muotoon), ja laadintaprosessia tukivälineineen voisi kehittää. Toinen ongelma on oppaan sisältö ja käytettävyys: sisältääkö opas tarpeeksi tietoa, puuttuuko olennaista tietoa, onko oppaassa liikaa tietoa? Onko tieto esitetty tarkoituksenmukaisessa muodossa, ja ovatko käytetyt julkaisuodot (Internet, Opinto-opaskirja) käyttäjäryhmien tarpeiden mukaisia?</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			

**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 esitettyjen menetelmien käyttöön (menetelmävalinta ei ole lopullinen). Projektiryhmällä on mahdollisuus saada tukea myös Korppi-liitäntöjen toteuttamiseen sekä paperijulkaisuprosessiin. Kehittämisessä tarvittava tiedekunnan prosessien ja opinto-oppaan tuottaja-käyttäjäroolien kartoitus 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 (ks. <http://www2.vy.fi/index.php?profile=strategiatyo&language=fin&pageref=0>). IT-tiedekunta on käynnistämässä myös toista kehittämisprojektia (MemoX), jonka kanssa projektilaiset voivat tehdä yhteistyötä muun muassa nykyisten ja tulevien toimintatapojen selvitys- ja suunnittelutyössä.

	<p><b>Aiheen työstämiseen käytettävät mahdolliset toteutusvälineet ja menetelmät:</b></p> <p>Menetelmät: Maler &amp; 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><b>Opiskelijoilta toivottava tiedollinen ja taidollinen osaaminen sekä esitysryhmän ihannekoostumuksesta esimerkiksi osaamisalueiden suhteen:</b></p> <p>Rakenteisten dokumenttien alueen, ja eritoten XML -kielen, sisältöjen hallinnan kehittämiseen liittyvien menetelmien (ks. menetelmät) tuntemus tarjoaa hyvän pohjan projektityöskentelylle; osalla ryhmäläisistä voisi olla tämän alueen tunte- musta. Muita projektityössä tarvittavia taitoja tai kiinnostuksen alueita ovat: do- kumenttien ja sisältöjen hallinta, käytettävyyden kehittäminen, organisaatioiden toiminnan kehittäminen, vaatimusmäärittely, mallintaminen ja suunnittelu, käyt- täjäystävällisten dokumentaatioiden ja tuottamisprosessien suunnittelu. Projekti- työn sisältö on monipuolinen; käyttäjähaastatteluista, opinto-oppaan käytettävyy- den ja sisällön määrittelystä ja oppaan tuottamisen nykytila-analyysistä aina uusi- en ratkaisujen suunnitteluun, ja pilotoimiseen saakka. Optimaalisessa ryhmässä voisi olla sekä teknisesti että ”käyttäjäläheisesti” orientoituneita henkilöitä.</p>
	<p><b>Projektin menestykselliseen hoitamiseen edellytettävä kielitaito:</b></p> <p>Suomen kieli riittää. Englannin kielen osaamisesta on etua tarvittavien teknisten määritysten ja ohjelma-oppaiden tulkinnassa.</p>
	<p><b>Projektiin osallistumisen estävät rajoitukset (Esimerkiksi opiskelijan työsuhde kilpailevaan yritykseen):</b></p>
4 MUUTA HUOMIOITTA- VAA	<p>Projektiryhmä saa apua ja tukea työskentelynsä sekä Korppi-henkilöstöltä (Vesa Lappalainen) että rakenteisten dokumenttien menetelmien ja työvälineiden osalta TKTL-laitokselta (Anne Honkaranta). Tiedekunnan toimisto huolehtii perehdy- tyksestä aihepiiriin, ja on käytettävissä niin nykytilan ja ongelmakohtien kartoit- tukseen, kuin tulevien ratkaisujen ideointiinkin. Projektin tulos edistää tiedekun- nan toimintaa, ja hyödyttää siten sekä tiedekunnan hallintoa että kaikkia tiede- kunnassa opiskelevia. On mahdollista, että projektiryhmäläisille tarjoutuu mah- dollisuus jatkaa työskentelyä kesä- tai gradutöiden merkeissä.</p>

**LIITE 2: OPINTO-OPPAAN LUKU.DTD**

```

<!-- ===== -->
<!-- opinto-oppaan luku.dtd -->
<!-- ===== -->
<!-- XooZoo-projektin luku.dtd:n versio 1.1. Alunperin muokattu Annen Honkarannan
(AHo) laatimasta pohjasta -->
<!-- Tiina Penttinen & Pekka Rinne -->
<!-- Lisayksia TeX/PDF-kasittelya varten -->
<!-- Miika Nurminen -->
<!-- ===== versiohistoria ===== -->
<!-- lisatty alaluku3, koska tarvitaan otsikkotaso 4 kayttoon. 06062005/AHo -->
<!-- lisatty luvuille, kuville ja taulukoille attribuutti numero numeroiden tallennuspai-
kaksi numerointeja varten /06062005/AHo -->
<!-- lisatty taulukolle attribuutti reuna, joka on normaali, nakymaton tai paksu, vrt.
html-tulosteessa border=0, 1 tai 2 -->
<!-- eli taulukon reuna-arvolla pitaisi voida saataa taulukon reunan paksuutta tai nakyy-
ko se ollenkaan tulosteessa -->
<!-- 23052005 /Anne Honkaranta -->

<!-- lisatty taulukon sarakkeelle attribuutit sarakeyli ja riviyli pdf-kasittelya varten. si-
vunvaihto-elementti lisatty lukutasolle pdf-kasittelya varten. PRi 18.5.2005 -->
<!-- lisatty taulukon sarakkeeseen attribuutti tyyppi pdf-kasittelya varten. MN 2.5.2005
-->
<!-- V1.1 lisatty kappaleelle valiotsikko-elementti, kappaleen kappaletyyppi-
attribuuteista poistettiin tiivistelma ja johdanto ja sen nimiset elementit tehtiin luvulle ja
alaluvuille. Luvuille ja alaluvuille laitettiin myos attribuutti taso skeemamuunnoksia var-
ten. Lisaksi lisatty luvulle, alaluku1:lle ja alaluku2:lle meta-dtd-liittyma 25.2.2005 TPe
ja PRi. Lisatty rivinvaihtoelementti kappaleelle muotoilun tarkentamista varten seka
taulukolle attribuutti tyyppi PDF-muunnosta varten. Tyyppena tavallinen tai erikois.
Taulukon sarakkeelle lisatty mahdollisuus merkata teksti suoraan korostukseksi. PRi
12.4.2005 -->
<!-- V 1.0 katselmoinnissa paatetyn perusteella kaikki id-attribuutit muutettu ID-
tyyppiseksi ja poistettu opas-elementti (oli juurena) 24.2.2005 TPe -->
<!-- V 1.0 Listan otsikko voi nyt sisaltaa korostuksen ja korostukselle lisatty attribuutti
kursivoitu, korostuksen attribuutilta otettu oletusarvo pois ja attribuutti on muutettu pa-
kolliseksi -->
<!-- V 0.33 Entiteetit (lakipykala, euro) poistettu -->
<!-- V 0.32 korjailtu muutamia attribuuttimaarityksia. Lisatty kieli-attribuutti kappaleel-
le takaisin -->
<!-- kappale-elementin kappaletyypille lisatty testausta varten arvo tavallinen -
->
<!-- hakutermin elementin termiid-attribuutti vaihdettu implied-tyyppiseksi -->
<!-- 2.2.2005/ AHo + vastuuhenkilon nimi-elementti jaettu etu- ja sukunimi -
elementeiksi 3.2.2005 PRi -->
<!-- V 0.31 1.2.2005 kieli-attribuutti "fi | en" vaihdettu kappaleelta lukutasoille (luku,
alaluku1, alaluku2). Ristiviittaus mahdollistettu luku-, alaluku- ja kappaleetasoille. Li-
saksi hakutermeille lisatty ID-tyyppinen tunniste.-->

```

```

<!-- V 0.3 27.1.2005 testauksen jälkeen nimetty versioksi 0.3. testidatana kaytetty opin-
to-oppaan oikeaa sisaltoa PRi & TPe -->
<!-- V 0.23 26.1.2005 kuviot-elementti poistettu, avainsanat-elementti muutettu haku-
termit-elementiksi. PRi & TPe -->
<!-- V 0.22 26.1.2005 kuviot-etiteetti muutettu kuviot-elementiksi, koska editori ei
osannut kasitella kuviot-entiteettia tapauksissa, joissa tarvittiin useita ilmentymia kuvi-
oista. PRi & TPe -->
<!-- V 0.21 26.1.2005 korjauksia katselmoinnin pohjalta: kappale-elementtiin lisatty
kommentti-elementti seka kappale-elementille luotu kieli attribuutti fi | en. ID-
attribuutti lisatty luvuille (aliluvut mukaanlukien), kappaleelle ja taulukolle. Luvuille ja
kappaleille luotu avainsanat elementti.taulukon sarakkeeseen luotu mahdollisuus lisata
lista. PRi & TPe -->
<!-- V 0.2 25.1.2005 Annen kanssa tarkastettu ja kayty lapi. Sarakkeeseen voi lisata nyt
suoraan PCDATAA. PRi & TPe-->
<!-- V 0.13 25.1.2005 (yksinkertaistettu edellisesta versiosta. parametri-entiteetteja va-
hennetty) PRi & TPe -->
<!-- V 0.12 25.1.2005 (Korjailtu Anne Honkarannan kanssa, muutoksia kappale-
elementtiin ja          vastuuhenkiloelementtiin) PRi & TPe & AHO
-->
<!-- V 0.11 20.1.2005 (Korjailtu Anne Honkarannan kanssa) PRi & TPe & AHO-->
<!-- V 0.1 aloitettu ke 19.1.2005 PRi & TPe -->
<!-- ===== -->

<!-- ===== -->
<!--          entiteetit -->
<!-- ===== -->

<!-- ===== -->
<!--          luku          -->
<!-- ===== -->
<!ENTITY % meta SYSTEM "meta.dtd">
%meta;
<!ENTITY % kappale SYSTEM "kappale.dtd">
%kappale;

<!ELEMENT luku (dokutieto?, otsikko, tiivistelma?, johdanto?, vastuuhenkilo*, (kap-
pale | kuvio | lista | taulukko | sivunvaihto)*, alaluku1*, allekirjoitus?)>
<!ATTLIST luku
id ID #IMPLIED
taso CDATA #FIXED "1"
kieli (fi | en ) "fi"
numero CDATA #IMPLIED
>

<!ELEMENT alaluku1 (dokutieto?, otsikko, tiivistelma?, johdanto?, vastuuhenkilo*,
(kappale | kuvio | lista | taulukko | sivunvaihto)*, alaluku2*, allekirjoitus?)>
<!ATTLIST alaluku1
id ID #IMPLIED

```

```

taso CDATA #FIXED "2"
kieli (fi | en ) "fi"
numero CDATA #IMPLIED
>

```

```

<!ELEMENT alaluku2 (dokutieto?, otsikko, tiivistelma?, johdanto?, vastuuhenkilo*,
(kappale | kuvio | lista | taulukko | sivunvaihto)*, alaluku3*, allekirjoitus?)>
<!ATTLIST alaluku2
id ID #IMPLIED
taso CDATA #FIXED "3"
kieli (fi | en ) "fi"
numero CDATA #IMPLIED
>

```

```

<!ELEMENT alaluku3 (dokutieto?, otsikko, tiivistelma?, johdanto?, vastuuhenkilo*,
(kappale | kuvio | lista | taulukko | sivunvaihto)*, allekirjoitus?)>
<!ATTLIST alaluku3
id ID #IMPLIED
taso CDATA #FIXED "4"
kieli (fi | en ) "fi"
numero CDATA #IMPLIED
>

```

```

<!ELEMENT otsikko (#PCDATA | korostus)*>
<!ATTLIST otsikko
numero CDATA #IMPLIED
>

```

```

<!ELEMENT vastuuhenkilo (titteli?, nimi, yhteystieto?, tarkennus?)>
<!ELEMENT titteli (#PCDATA)>
<!ELEMENT nimi (etunimi, sukunimi)>
<!ELEMENT etunimi (#PCDATA)>
<!ELEMENT sukunimi (#PCDATA)>
<!ELEMENT yhteystieto (#PCDATA)>
<!ELEMENT tarkennus (#PCDATA)>
<!ELEMENT allekirjoitus (#PCDATA)>
<!ELEMENT tiivistelma (#PCDATA)>
<!ELEMENT johdanto (#PCDATA)>

```

```

<!-- ===== -->
<!--      kappale -omassa tiedostossaan -->
<!-- ===== -->

<!-- ===== -->
<!--      kuvio      -->
<!-- ===== -->

```

```

<!ELEMENT kuvio (kuvanimi?, kuva, kuvateksti?)>

```

```

<!ATTLIST kuvio
id ID #IMPLIED
>

```

```

<!ELEMENT kuvanimi (#PCDATA)> <!-- kuvan nimi, esim. Agora -->
<!ELEMENT kuva EMPTY>
<!ATTLIST kuva
src CDATA #REQUIRED
leveys CDATA #IMPLIED
korkeus CDATA #IMPLIED
numero CDATA #IMPLIED
>

```

```

<!ELEMENT kuvateksti (kappale+)> <!-- pidemmat tekstit kuvasta tahan -->

```

```

<!-- ===== -->
<!-- lista -->
<!-- ===== -->

```

```

<!ELEMENT lista (otsikko | listakohta | alilista)*>
<!ATTLIST lista
id ID #IMPLIED
tyyppi (pampula | viiva | numero | kirjain) "pampula"
>

```

```

<!ELEMENT listakohta (#PCDATA | kappale | korostus)*>
<!ELEMENT alilista (otsikko | listakohta)*>

```

```

<!-- ===== -->
<!-- taulukko -->
<!-- ===== -->

```

```

<!--
tavallinen: pdf-tulostuksessa 1 sivulle mahtuva taulukko, jonka tarkan paikan ladonta-
ohjelma maaraa erikois: monisivuinen taulukko
-->

```

```

<!ELEMENT taulukko (taulukkonimi?, rivi+)>
<!ATTLIST taulukko
id ID #IMPLIED
tyyppi (tavallinen | erikois) "tavallinen"
reuna (normaali | nakymaton | paksu) "normaali"
numero CDATA #IMPLIED
>

```

```

<!ELEMENT taulukkonimi (#PCDATA)>
<!ELEMENT rivi (sarake)+>
<!ELEMENT sarake (#PCDATA | otsikko | kappale | lista | korostus)*>

```

```
<!-- tavallinen: 1-rivinen sarake
      monirivi:  monirivinen sarake, jonka leveys määritetään automaattisesti
      kapea:     monirivinen, n. 2cm leveä sarake
-->

<!-- sarakeyli vrt. colspan riviyli vrt. rowspan + CDATA = tieto kuinka monta ri-
via/saraketta menneen yli-->

<!ATTLIST sarake
  tyyppi (tavallinen | monirivi | kapea) "tavallinen"
  sarakeyli CDATA #IMPLIED
  riviyli CDATA #IMPLIED
>

<!-- ===== luku.dtd loppuu ===== -->
```



### LIITE 3: ENSIMMÄISEN VUODEN OPISKELIJOIDEN OPAS- VERSION KOOSTEMÄÄRITYS

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE opaskooste SYSTEM "../skriptit/opaskooste.dtd">
<?xml-stylesheet type="text/xsl" href="../skriptit/opaskooste2luku.xsl"?>
<opaskooste>
  <meta>
    <laatimispvm>
      <paiva>13</paiva>
      <kuukausi>05</kuukausi>
      <vuosi>2005</vuosi>
    </laatimispvm>
    <laatija>
      <etunimi>Eija</etunimi>
      <sukunimi>eijan testi</sukunimi>
      <organisaatio>JY/TKTL</organisaatio>
    </laatija>
    <opasvariantti>Fuksiopas1</opasvariantti>
  </meta>
  <esiluvut>
    <tiedosto>varadekaanin_tervehdys.xml</tiedosto>
    <tiedosto>ensimmaisen_viikon_ohjelma.xml</tiedosto>
  </esiluvut>
  <tekstiluvut>
    <luku>
      <aihekuvaus>Yliopiston esittelyä</aihekuvaus>
      <tiedosto>../opas/jyvaskylan_yliopiston_esittely.xml</tiedosto>
    </luku>
    <luku>
      <aihekuvaus>Tiedekunnan ja laitosten esittelyä</aihekuvaus>
      <tiedos-
to>../opas/tiedekunnan_ja_laitosten_esittely.xml</tiedosto>
    </luku>
    <luku>
      <tiedosto>opintojen_kaynnistaminen.xml</tiedosto>
    </luku>
    <luku>
      <tiedosto>opiskelu.xml</tiedosto>
    </luku>
    <luku>
      <tiedosto>../opas/opintoneuvonta.xml</tiedosto>
    </luku>
    <luku>
      <tiedosto>erilaiset_opiskelijavalinnat.xml</tiedosto>
    </luku>
  </luku>

```

```

        <tiedosto>../opas/opinto_oikeudet.xml</tiedosto>
</luku>

<luku>
    <aihekuvaus>Tutkintoasiaa</aihekuvaus>
    <tiedosto>tutkinnot.xml</tiedosto>
</luku>
<luku>
    <tiedosto>tktl.xml</tiedosto>
</luku>
<luku>
    <tiedosto>tfl.xml</tiedosto>
</luku>
<luku>
    <tiedos-
to>../opas/yhteistyokumppaneiden_opinnot.xml</tiedosto>
</luku>
<luku>
    <tiedos-
to>kieli_ja_viestintaopintojen_merkitys_fuksi.xml</tiedosto>
</luku>
<luku>
    <tiedosto>fuksi_todistukset_ja_valmistuminen.xml</tiedosto>
</luku>
<luku>
    <tiedosto>../opas/maisterin_tutkinnon_jalkeen.xml</tiedosto>
</luku>
<luku>
    <tiedosto>../opas/kansainvalistymisen.xml</tiedosto>
</luku>
<luku>
    <tiedos-
to>../opas/muuta_tarpeellista_ja_hyodyllista.xml</tiedosto>
</luku>
<luku>
    <tiedos-
to>muiden_tiedekuntien_opiskelijoiden_sivuaineopinnot.xml</tiedosto>
</luku>
</tekstiluvut>
<liitteet>
    <tekstiliite>
        <liitetiedos-
to>../opas/opintosuoritusjohtosaanto.xml</liitetiedosto>
</tekstiliite>
<tekstiliite>
    <liitetiedosto>../puhlu/kaikki_henkilotiedot.xml</liitetiedosto>
</tekstiliite>
<kurssikuvausliite>

```

<huomautus><kappale>Tämä liite sisältää tietoja IT-tiedekunnan kandidaattiopintoihin kuuluvistalukuvuonna 2005-2006. Kurssien tarkemman aikataulun löydät Korpista sivulta:

<rivinvaihto />

<korostus tyyp-

pi="www">https://korppi.jyu.fi/kotka/course/student/organisationList.jsp</korostus>,<rivinvaihto />

kun kirjoitat hakukenttään kurssin koodin.

<rivinvaihto></rivinvaihto>Korpista löytyvät tiedot myös muusta opetustarjonnasta.</kappale></huomautus>

<liitetiedosto>it\_kurssit.xml</liitetiedosto>

</kurssikuvausliite>

<!-- ytp\_kurssit.xml ja matikan\_opinnot.xml -tiedostojen sisältö vajavaista, ei kannattane ottaa oppaaseen. 9.6.2005 Pekka Rinne

<kurssikuvausliite>

<liitetiedosto>ytp\_kurssit.xml</liitetiedosto>

</kurssikuvausliite>

<kurssikuvausliite>

<liitetiedosto>matikan\_opinnot.xml</liitetiedosto>

</kurssikuvausliite-->

</liitteet>

<takaluvut>

<tiedosto>../opas/kartan\_selitykset.xml</tiedosto>

<tiedosto>date.xml</tiedosto>

</takaluvut>

</opaskooste>

## LIITE 4: VANHEMPIEN OPISKELIJOIDEN OPASVERSION KOOSTEMÄÄRITYS

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE opaskooste SYSTEM "../skriptit/opaskooste.dtd">
<?xml-stylesheet type="text/xsl" href="../skriptit/opaskooste2luku.xsl"?>
<opaskooste>
  <meta>
    <laatimispvm>
      <paiva>2</paiva>
      <kuukausi>06</kuukausi>
      <vuosi>2005</vuosi>
    </laatimispvm>
    <laatija>
      <etunimi>Pekka</etunimi>
      <sukunimi>Rinne</sukunimi>
      <organisaatio>JY/TKTL</organisaatio>
    </laatija>
    <opasvariantti>Vanhojen opas</opasvariantti>
  </meta>
  <esiluvut>
    <tiedosto>dekaanin_tervehdys.xml</tiedosto>
    <tiedosto>opas_johdanto.xml</tiedosto>
  </esiluvut>
  <tekstiluvut>
    <luku>
      <aihekuvaus>Yliopiston esittely</aihekuvaus>
      <tiedosto>jyvaskylan_yliopiston_esittely.xml</tiedosto>
    </luku>
    <luku>
      <aihekuvaus>Tiedekunnan ja laitosten esittely</aihekuvaus>
      <tiedosto>tiedekunnan_ja_laitosten_esittely.xml</tiedosto>
    </luku>
    <luku>
      <tiedosto>muutokset.xml</tiedosto>
    </luku>
    <luku>
      <tiedosto>siirtymavaiheen_ohjeistus.xml</tiedosto>
    </luku>
    <luku>
      <tiedosto>vanhojen_opiskelu.xml</tiedosto>
    </luku>
    <luku>
      <tiedosto>opinto_oikeudet.xml</tiedosto>
    </luku>
    <luku>
      <tiedosto>opintoneuvonta_vanhat.xml</tiedosto>
  </tekstiluvut>
</opaskooste>

```

```

</luku>
<luku>
  <aihekuvaus>Tutkintoasiaa</aihekuvaus>
  <!-- tama tutkinnot.xml on vanhojen oppaan tutkinnoista. fuksi-
oppaan tutkinnot.xml siirretty fuksiopas-kansioon, jossa
ainoastaan fuksien uudet tutkinnot. Fuksi-oppaan tutkinnot.xml ei
siis muutu. -->
  <tiedosto>tutkinnot_vanha.xml</tiedosto>
</luku>

<luku>
  <tiedosto>tktl_vanha.xml</tiedosto>
</luku>
<luku>
  <tiedosto>tll_vanha.xml</tiedosto>
</luku>

<luku>
  <tiedos-
to>kieli_ja_viestintaopintojen_merkitys_vanha.xml</tiedosto>
</luku>
<luku>
  <tiedosto>kansainvalistymisen.xml</tiedosto>
</luku>
<luku>
  <tiedosto>yhteistyokumppaneiden_opinnot.xml</tiedosto>
</luku>
<luku>
  <tiedosto>vanha_todistukset_ja_valmistuminen.xml</tiedosto>
  <!-- siirretaanke opas-kansioon? -->
</luku>

<luku>
  <tiedosto>maisterin_tutkinnon_jalkeen.xml</tiedosto>
</luku>
<luku>
  <tiedosto>jatkokoulutus.xml</tiedosto>
</luku>

<luku>
  <tiedos-
to>vanhojen_muuta_tarpeellista_ja_hyodyllista.xml</tiedosto>
</luku>
<luku>
  <tiedos-
to>muiden_tiedekuntien_opiskelijoiden_sivuaineopinnot_vanhat.xml</tiedosto>

```

```

        </luku>
    </tekstiluvut>
    <liitteet>
        <tekstiliite>
            <liitetiedosto>opintosuoritusjohtosaanto.xml</liitetiedosto>
        </tekstiliite>
        <tekstiliite>
            <liitetiedosto>../puhlu/kaikki_henkilotiedot.xml</liitetiedosto>
        </tekstiliite>
        <kurssikuvausliite>
            <huomautus><kappale>Tämä liite sisältää tietoja IT-tiedekunnan opin-
toihin kuuluvistalukuvuonna 2005-2006. Kurssien tarkemman aikataulun löydät Korpis-
ta sivulta:
<rivinvaihto />
<korostus tyyp-
pi="www">https://korppi.jyu.fi/kotka/course/student/organisationList.jsp</korostus>,<r
ivinvaihto />
kun kirjoitat hakukenttään kurssin koodin.
<rivinvaihto></rivinvaihto>Korpista löytyvät tiedot myös muusta opetustarjonnas-
ta.</kappale></huomautus>
            <liitetiedosto>it_kurssit.xml</liitetiedosto>
        </kurssikuvausliite>
        <!-- ei ollut valmiita taloustieteen eikä matikan kursseja Korpissa
10.6.2005 Pekka Rinne
        <kurssikuvausliite>
            <liitetiedosto>../fuksiopas/ytp_kurssit.xml</liitetiedosto>
        </kurssikuvausliite>
        <kurssikuvausliite>
            <liitetiedosto>../fuksiopas/matikan_opinnot.xml</liitetiedosto>
        </kurssikuvausliite-->
    </liitteet>
    <takaluvut>
        <tiedosto>kartan_selitykset.xml</tiedosto>
    <tiedosto>date.xml</tiedosto>
    </takaluvut>
</opaskooste>

```