

LAPPEENRANNAN TEKNILLINEN YLIOPISTO

Tietotekniikan osasto

Verkko-oppimateriaalien uudelleenkäytettävyys ja siirrettävyys oppimisalustalta toiselle

Aihe on hyväksytty Tietotekniikan osaston osastoneuvostossa 26.1.2005

Tarkastajat: Prof. Arto Kaarna (LTY)
Prof. Juha Puustjärvi (HY)

Ohjaaja: Heli Korpinen (EKAMK)

Savonlinnassa 26.9.2005

Hannu Matikainen
Moinsalmentie 1763
57230 Savonlinna
Puh. 050 310 6015

TIIVISTELMÄ

Lappeenrannan teknillinen yliopisto

Tietotekniikan osasto

Hannu Matikainen

Verkko-oppimateriaalien uudelleenkäytettävyys ja siirrettävyys oppimisalustalta toiselle

Diplomityö

2005

82 sivua, 26 kuvaa ja 5 taulukkoa

Tarkastajat: Professori Arto Kaarna ja professori Juha Puustjärvi

Hakusanat: verkko-oppimateriaali, oppimisaihiot, oppimisalustat, uudelleenkäytettävyys

Työssä tarkastellaan oppimisalustoilla käytettäviä testikysymyksiä ja oppisisältöjä uudelleenkäytettävyyden ja siirrettävyyden näkökulmasta. Tavoitteena on selvittää, minkälaisia välineitä määrämuotoisten sisältöjen tuottamista varten on olemassa, ja miten niillä luodut materiaalit ovat siirrettävissä työvälineestä toiseen ja oppimisalustoille. Tutkittavana ovat sisällön paketointiin tarkoitettut SCORM- ja IMS Content Packaging -spesifikaatiot sekä kysymysten esittämiseen tarkoitettu IMS Question and Test Interoperability -spesifikaatio.

Pääpaino työssä on selvittää, miten oppisisällöt ovat siirrettävissä WebCT- ja Moodle-alustojen välillä. Tähän liittyen on luotu ohjeistus, miten yksittäiset tiedostot, kysymykset ja Sisältökokoaisuus-työkalulla koostetut sisällöt siirretään WebCT:ltä Moodleen.

ABSTRACT

Lappeenranta University of Technology

Department of Information Technology

Hannu Matikainen

Verkko-oppimateriaalien uudelleenkäytettävyys ja siirrettävyys oppimisolustalta toiselle

Thesis for the Degree of Master of Science in Technology

2005

82 pages, 26 figures and 5 tables

Examiners: Professor Arto Kaarna and professor Juha Puustjärvi

Keywords: e-learning content, learning objects, learning management systems, re-usability

The objective of this study is to examine the test questions and learning contents used in learning management systems from re-usability's and transferability's perspective. The purpose is to find out what kind of tools exist for producing standard contents, and how one can export the materials, which are created by them, from one tool to another and to learning management systems.

The SCORM- and IMS Content packaging -specifications that are designed for content packaging, as well as IMS Question and Test Interoperability -specification that is designed for starting questions, are studied.

The major issue in this study is to find out how one can export the learning contents from learning management system WebCT to another learning management system Moodle. In connection with that one has created a guidance where it is explained how individual files, questions and contents created with Content Module tool are exported from WebCT to Moodle.

SISÄLLYSLUETTELO

1 JOHDANTO.....	1
1.1 Työn tausta ja rajaukset.....	1
1.2 Tutkimusongelma ja tutkimuksen tavoite	2
2 OPPIMISAIHIOT	3
2.1 Määritelmä	3
2.2 Oppimisasihioiden pedagoginen käyttö.....	4
2.2.1 Aihoiden luokittelu.....	4
2.2.2 Aihioille asetettavia vaatimuksia.....	6
2.3 Oppimisasihion koko ja koostumus	7
2.4 Oppimisasihoita kohtaan esitettyä kritiikkiä	10
3 OPPIMISYMPÄRISTÖÖN LIITTYVIÄ TEKNISIÄ RATKAISUJA	11
3.1 Oppimisympäristö-käsite	11
3.2 Oppimisalustat.....	11
3.3 Oppimateriaalin hallintajärjestelmät	13
3.4 Oppimateriaalipankit.....	13
4 STANDARDOINTI	15
4.1 Standardoinnin etuja.....	15
4.2 Spesifikaatioiden ja standardien tuottajat.....	16
4.2.1 AICC - Aviation Industry CBT Committee	18
4.2.2 ARIADNE - Alliance of Remote Instructional Authoring & Distribution Network for Europe.....	19
4.2.3 IMS Global Learning Consortium.....	20
4.2.4 ADL - Advanced Distributed Learning Initiative	20
4.2.5 Standardointiorganisaatiot.....	21
5 SISÄLLÖNTUOTANTOON LIITTYVIÄ MÄÄRITYKSIÄ	23
5.1 Oppimisasihioiden metatiedon kuvaaminen	23
5.2 IMS Question and Test Interoperability – kysymykset ja testit	26
5.2.1 Yleistä.....	26
5.2.2 Spesifikaation käyttöalue	26
5.2.3 Kysymystyyppit.....	28
5.2.4 Kysymysten ja testien tallentaminen	29
5.3 IMS Content Packaging – sisällön paketointi	31
5.3.1 Sisältöpakettien komponentit	31

5.3.2 Julistusosan elementit.....	32
5.4 SCORM-referenssimalli.....	35
5.4.1 Sisällönkoostamismalli.....	36
5.4.2 Käytönaikainen ympäristö.....	39
5.4.3 Oppimisaihioiden esittämisjärjestys ja niiden välinen navigointi.....	41
6 SISÄLLÖN TUOTTAMINEN OPPIMISALUSTOILLE.....	44
6.1 WebCT.....	44
6.2 Moodle.....	45
6.3 Kysymykset ja testit.....	45
6.3.1 QAed-editori.....	46
6.3.2 Questionmark Perception.....	47
6.3.3 Respondus.....	48
6.3.4 Hot Potatoes.....	49
6.4 Sisällön paketointi.....	51
6.4.1 RELOAD Editor.....	52
6.4.2 eXe – eLearning XHTML editor.....	57
6.4.3 Coursegenie.....	59
7 SISÄLLÖN SIIRTÄMINEN.....	62
7.1 Yksinkertainen tapa siirtää materiaalia kurssilta toiselle.....	62
7.2 Kysymysten siirtäminen.....	63
7.2.1 Siirtäminen WebCT-muodossa.....	63
7.2.2 IMS QTI -muodon käyttäminen.....	66
7.3 Materiaalin siirtäminen sisällön paketoinnin avulla.....	67
8 SUOSITUS TOIMENPITEIKSI SISÄLLÖN LUOMISESSA JA SIIRROSSA.....	69
8.1 Kysymysten ja testien luominen.....	69
8.2 Sisällön luominen.....	70
8.2.1 Työkalut.....	70
8.2.2 Sisällön tuottamisen toimintamalli.....	71
8.3 Sisällön siirtäminen WebCT:ltä Moodleen.....	72
8.4 Spesifikaatiot sisällöntuotannossa.....	73
9 POHDINTA JA JOHTOPÄÄTÖKSET.....	75
9.1 Työn tavoitteisiin pääseminen ja tulokset.....	75
9.2 Jatkossa selvitettävää.....	76
LÄHDELUETTELO.....	77

KÄYTETYT LYHENTEET

ADL	Advanced Distributed Learning
AGR	AICC Guidelines & Recommendations
AICC	Aviation Industry CBT Committee
API	Application Programming Interface
ARIADNE	Alliance of Remote Instructional Authoring & Distribution Network for Europe
CBT	Computer-Based Training
CEN/ISSS	European Committee for Standardization - Information Society Standardization System
CEN/ISSS WS-LT	CEN/ISSS Work Shop on Learning Technology
CMI	Computer Managed Instruction
CMU	Content Migration Utility
EKAMK	Etelä-Karjalan ammattikorkeakoulu
EKKY	Etelä-Karjalan koulutuskuntayhtymä
HACP	HTTP-based AICC CMI Protocol
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
IEEE	Institute of Electrical and Electronics Engineers
IEEE LTSC	IEEE Learning Technology Standards Committee
IMS	Instructional Management System
IMS CMU	IMS Content Migration Utility
IMS CP	IMS Content Packaging
IMS LD	IMS Learning Design
IMS LRM	IMS Learning Resource Meta-data
IMS QTI	IMS Question and Test Interoperability

ISO/IEC	International Organization for Standardization - International Electro-technical Commission
ISO/IEC JTC	ISO/IEC Joint Technical Committee
KPM	Knowledge Pool System
LCMS	Learning Content Management System
LMS	Learning Management System
LO	Learning Object
LOM	Learning Object Metadata
OA	Oppimisaihio
PDF	Portable Document Format
RIO	Reusable Information Object
RLO	Reusable Learning Object
SCO	Sharable Content Object
SCORM	Sharable Content Object Reference Model
SCORM CAM	SCORM Content Aggregation Model
SCORM RTE	SCORM Run-Time Environment
SCORM SN	SCORM Sequencing and Navigation
TAO	Tietokoneavusteinen opetus
UKCMF	UK Common Metadata Framework
URL	Uniform Resource Locator
UTF	Unicode Transformation Format
VLE	Virtual Learning Environment
XML	Extended Markup Language

1 JOHDANTO

1.1 Työn tausta ja rajaukset

Työ tehdään Etelä-Karjalan ammattikorkeakoululle (EKAMK), joka on osa Etelä-Karjalan koulutuskuntayhtymää (EKKY). Työ liittyy verkko-opetukseen, jota EKKY:ssä on toteutettu 1990-luvun lopulta lähtien. EKKY:n opiskelijamäärä on lähes 7000 ja opetushenkilöstöä on noin 550, kun mukaan lasketaan ammattiopiston, ammattikorkeakoulun ja aikuisopiston opiskelijat ja opettajat.

Verkko-opetusta varten EKKY:ssä on käytössä kaksi oppimisalustaa: WebCT ja Moodle. Näistä WebCT on ollut vuodesta 1999 lähtien pääasiallisena oppimisympäristönä, ja Moodlea on testattu useilla kursseilla syksystä 2004 alkaen. WebCT on kaupallinen tuote, jonka vuotuinen lisenssihintaa on varsin korkea. Sen kehittäminen on aloitettu jo vuonna 1996 Pohjois-Amerikan yliopistoissa, ja tällä hetkellä kyseessä on varsin kehittynyt ja valmis tuote. Moodle puolestaan on ilmainen avoimen lähdekoodin periaatteella toteutettu oppimisalusta, jonka ensimmäinen versio tuli jakeluun elokuussa 2002.

WebCT:n kehityskaaren koko ajan painolastina on ollut se vanha teknologia, jolle WebCT on rakennettu. Osasyynä lienee yhteensopivuuden varmistaminen, jolloin vanhoillakin versioilla tehdyt kurssit on mahdollista siirtää uusilla alustoilla käytettäväksi. Siirtyminen uusimpiin versioihin ei kuitenkaan ole sujunut aina helposti; tietyt versiovaihdot ovat olleet melko työläitä.

Moodle elää vasta alkutaivaltaan, mutta toisaalta sen kehityksessä on voitu hyödyntää sitä kokemusta, mitä verkko-opetuksesta ja -opiskelusta on kertynyt muita alustoja käytettäessä. Moodlen kehitystyö onkin voitu aloittaa puhtaalta pöydältä viimeisintä tietämystä ja teknologiaa hyödyntäen. Avoimeen lähdekoodiin pohjautuvana Moodlen räätälöinti omiin tarkoituksiin on myös mahdollista. Tällä hetkellä kiinnostus Moodleen on suurta niin Suomessa kuin maailmanlaajuisestikin, esimerkiksi Suomessa sitä testataan ja käytetään useissa oppilaitoksissa. EKAMK:ssa opettajakunta on todennut Moodlen helppokäyttöiseksi ja verkko-opetukseen hyvin soveltuvaksi.

Alustojen hinta- ja tekninen kehitys osaltaan ratkaisevat, siirrytäänkö jatkossa pelkästään jomankumman käyttöön vai jatketaanko edelleenkin kahden alustan mallilla. Ottaen huomioon WebCT:n lisenssimaksut ja Moodlen maksuttomuuden, EKKY:ssä on luonnollisesti kiinnostusta viimeksi mainittuun. Tässä työssä ei kuitenkaan ole tarkoitus ottaa syvällisesti kantaa mainittujen

alustojen keskinäiseen paremmuuteen; kyseessä ei ole kaiken kattava alustavertailu teknisessä eikä pedagogisessakaan mielessä.

1.2 Tutkimusongelma ja tutkimuksen tavoite

Materiaalin tuottaminen verkkokursseille on aina ollut jonkinasteinen ongelma siinä suhteessa, että miten oppimateriaali kannattaisi teknisesti tehdä. Materiaalilla tässä tarkoitetaan lähinnä itsearviointitehtäviä, testikysymyksiä ja niistä koostettuja testejä sekä varsinaista sisältömateriaalia; näitä kutsutaan myös oppimisasihioksi. Keskeisin kysymys on, että miten materiaali olisi helposti uudelleenkäytettävissä eri kursseilla ja vieläpä eri oppimisalustoilla? Millä tavalla ja millä työvälineillä materiaali kannattaisi tehdä, ja mihin muotoon tallentaa siirrettävyys huomioiden?

Materiaalin tekeminen oikealla tavalla uudelleenkäytettävyys huomioiden ei ole itsestäänselvyys yhtä oppimisalustaakaan käytettäessä saati kun alustoja on useampia. Tällä hetkellä Moodle tukee SCORM-spesifikaatiota ja WebCT tukee IMS-spesifikaatiota, jotka molemmat periaatteessa mahdollistavat määrämuotoisten kurssisisältöjen (oppimisasihioiden) siirtämisen alustalta toiselle. Tässä työssä pyritään löytämään vastaukset seuraaviin kysymyksiin:

- Miten kurssisisällöt (sisältömateriaali, itsearviointitehtävät, testikysymykset ja testit) kannattaisi alun perin luoda, jotta siirrettävyys ympäristöstä toiseen onnistuisi parhaiten uudelleenkäytettävyys huomioiden?
- Voidaanko olemassa olevat kurssisisällöt siirtää alustalta toiselle ja miten se tehdään?
- Millä välineillä on mahdollista tehdä IMS-/SCORM-spesifikaatioiden mukaista kurssimateriaalia (oppimisasihiota)?
- Mitä eroa on IMS- ja SCORM-spesifikaatioilla; ovatko ne keskenään yhteensopivia tai miten helposti yhteen sovitettavissa?
- Mitkä ovat näiden spesifikaatioiden tulevaisuudennäkymät; onko jompikumpi mahdollisesti saavuttamassa valta-aseman?

Työn tavoitteena on luoda toimintamalli ja ohjeistus kurssisisältöjen luomiseen ja siirtämiseen alustalta toiselle. Kurssisisältöjen siirtämisessä käytetään esimerkkinä WebCT:tä ja Moodlea painopisteen ollessa WebCT:ltä Moodleen päin. Toimintamallissa otetaan esisijaisesti huomioon WebCT:n ja Moodlen välinen siirrettävyys, mutta pyritään myös yleisempään uudelleenkäytettävyyteen eri alustojen välillä mahdollisuuksien mukaan.

2 OPPIMISAIHIOT

2.1 Määritelmä

Oppimisaihiot (Learning Objects) otettiin käyttöön verkko-opetuksessa varsinaisesti vasta 2000-luvun alkupuolella, vaikka käsite sinänsä on peräisin 1990-luvun alusta. Käsitteen luojana pidetään yleisesti Autodesk-yhtiön futuristia Wayne Hodginsia, joka keksi Lego-palikoista analogian oppimisen rakennuspalikoihin (Jacobsen 2002). Toisaalta on todettava, että oppimisaihioissa on sama perusidea kuin hypertekstissä ja hypermediassa, jotka käsitteet Ted Nelson kehitti jo 1960-luvulla.

Oppimisaihion määritelmiä on lukuisia, joista seuraavana neljä esimerkkiä. IEEE:n Learning Object Metadata -standardin määritelmän mukaan oppimisaihio on ”mikä tahansa olio, digitaalinen tai ei-digitaalinen, mitä voidaan käyttää teknologisesti tuetussa oppimisessa” (IEEE LOM 2002a). Aiheeseen laajasti perehtynyt Wiley (2000) puolestaan määrittelee oppimisaihion olevan ”mikä tahansa digitaalinen resurssi, jota voidaan uudelleenkäyttää oppimisen tukena”. Määritelmä sisältää kaiken mikä on jaettavissa verkon kautta, ja voi olla kooltaan pieni tai suuri. Esimerkkinä pienistä uudelleenkäytettävistä digitaalisista resursseista Wiley mainitsee kuvat, lyhyet tekstipätkät, video- ja äänileikkeet sekä pienet www-sovellukset kuten Java-laskimet. Laajoja uudelleenkäytettäviä digitaalisia resursseja puolestaan ovat esimerkiksi kokonaiset www-sivut/sivustot, joissa on yhdistetty eri medialementtejä tai sovelluksia, ja jotka tarjoavat kokonaisen opetuksellisen tapahtuman.

Polsani (2003) muotoilee määritelmän seuraavasti: ”Oppimisaihio on itsenäinen oppisisältöyksikkö, jota voidaan uudelleenkäyttää opetukseen monenlaisissa asiayhteyksissä”. L’Allierin (1997) ja Masien (2003) mukaan oppimisaihiosta on löydyttävä kaikki tietyssä oppimistapahtumassa tarvittavat elementit. Oppimisaihio on siis ”pienin itsenäinen kokonaisuus, joka sisältää esimerkiksi tavoitteen oppimistapahtumalle, opiskeltavan sisällön ja lopuksi arvioinnin, jolla tutkitaan, onko tavoitteeseen päästy”.

Tarkasteltaessa edellä esitettyjä määritelmiä voidaan todeta, että IEEE:n määritelmä on liian yleinen; sen mukaan oppimisaihio voi periaatteessa olla mikä tahansa esine, vaikkapa valokuva tai tietokonelaite. Wileyn määritelmä on lähes yhtä laaja, vaikka se rajoittuu vain digitaalisiin resursseihin. Tätä määritelmää on kritisoitu (Polsani 2003) siitä, että miksi tekstejä, kuvia, ääniä, videoita, www-sivuja yms. pitää kutsua oppimisaihioiksi – eikö niitä voisi kutsua noilla mainituilla nimillä? Polsani itse esittää varsin käyttökelpoisen määritelmän huomioiden aihion käytön

opetuksellisesti erilaisissa asiayhteyksissä, mutta ei ole jostain syystä halunnut korostaa niiden digitaalisuutta. L’Allierin ja Masien näkökulma painottaa oppimisasihoiden pedagogista funktiota – sisältö on vain yksi osa kokonaisuudesta.

Tämän työn kannalta sopiva määritelmä on kaikkia edellisiä soveltaen: ”Oppimisasihio on yhden tai useamman digitaalisen mediaelementin muodostama kokonaisuus, jota voidaan uudelleen käyttää opetukseen erilaisissa asiayhteyksissä”. Tämä kattaa sekä asiasisällön että pedagogisten tavoitteiden kokoamisen uudelleenkäytettävistä ”rakennuspalikoista”.

Aikaisemmin mainitun LOM-standardin suomenkielisessä versiossa (IEEE LOM 200b) Learning Object on käännetty muotoon *oppisisältö*. Kyseistä termiä on kritisoitu (Ilomäki 2004, 10), koska se antaa kuvan oppimisasihion roolista pelkkänä tietosisältönä, tiedonlähteenä. Oppisisältö on hyvä suomenkielinen sana, joten sitä – osittain käsitteiden vakiintumattomuudesta johtuen – käytetään myöhemmin tekstissä oppimisasihoiden rinnalla. Näiden lisäksi esiintyy käsite *oppimateriaali*, joka tämän työn kontekstissa on synonyymi oppisisällölle ja oppimisasihioille. Oppimisasihiosta käytetään myös lyhyempää *aihio*-termiä. Englanninkielisessä kirjallisuudessa oppimisasihio-käsite tunnetaan ainakin nimillä *Learning Object* (LO), *Reusable Learning Object* (RLO) ja *Sharable Content Object* (SCO).

Oppimisasihioihin voidaan nähdä olevan kaksi pääasiallista lähestymistapaa: pedagoginen ja tekninen. Pedagogisessa tarkastelutavassa ahiolla on jokin tietty rooli oppimisprosessissa. Tekninen näkökulma puolestaan painottaa aihoiden standardointia, jolla mahdollistetaan niiden uudelleenkäytettävyys eli siirrettävyys työvälineohjelmasta ja/tai oppimisalustasta toiseen. Nämä eivät ole toisiaan poissulkevia, vaan pikemminkin toisiaan tukevia näkökulmia. Silander (2004) asettaa vielä sisällön erilleen kolmanneksi näkökulmaksi.

2.2 Oppimisasihoiden pedagoginen käyttö

2.2.1 Aihoiden luokittelu

Silander (2003, 37-42 ja 71-72) jakaa oppimisasihiot niiden pedagogisen funktion eli käytön tavoitteen mukaan seitsemään ryhmään:

1. *Aktivointi*: oppimisasihio, jolla herätetään oppija pohtimaan asioita ja omaksumaan uutta. Tämä voidaan tehdä aktivoivilla kysymyksillä.

2. *Kontekstin luonti, ongelman asettaminen*: aihio, jolla luodaan lähtökohta käsiteltävälle asialle; voi myös ohjata näkemään keskeiset ongelmat käsiteltävästä aiheesta.
3. *Hypoteesin/työskentelyteorian testaaminen*: aihio, jolla oppija voi testata omia selityksiään ja tehdä omat johtopäätöksensä; esimerkiksi interaktiivinen simulaatio.
4. *Tietolähde*: materiaalia tiedonrakentelulle; oppimisaihioita, jotka havainnollistavat opettavaa asiaa tai ilmiötä.
5. *Tiedonrakentelu*: oppimisen työkaluja, jotka ohjaavat tiedonrakenteluprosessia; työkaluja esimerkiksi oppimistehtävien työstämiseen ja ongelmanratkaisuun.
6. *Reflektio*: aihio, joka auttaa havainnoimaan omaa ajattelua ja toimintaa, ja jolla oppijat voivat suorittaa itsearviointia.
7. *Testaus/arviointi*: oppimisaihio, jolla testataan oppijan tietoja ja taitoja sekä arvioidaan oppimista ja osaamista.

Oppimisaihiot voidaan luokitella käytännönläheisemmin esimerkiksi EU:n CELEBRATE-hankeessa käytetyllä tavalla, taulukko 1. Silanderin edellä esittämien pedagogisten tavoitteiden toteuttamiseen voidaan käyttää erityyppisiä oppimisaihioita tai niiden yhdistelmiä; esimerkiksi kontekstin luonti voitaisiin tehdä vaikkapa kokeilu- tai tietolähdeaihioilla tai niitä yhdistämällä.

Taulukko 1. Oppimisaihioiden luokittelu (Jaakkola et al. 2004, 30-35).

Oppimisaihiotyyppi	Kuvaus
Arviointiaihiot	Tyypillisesti monivalinta-, lasku- tai aukkosanatehtäviä, jotka tietokone arvostelee.
Harjoitusohjelmat	Yksinkertaisia harjoituksia ja pelejä, joissa harjoitellaan tiettyä yksittäistä ilmiötä (esim. kertotaulun harjoittelu).
Tietolähteet	Tietoaineistot, joissa asia tai ilmiö on tavallisesti kuvailtu tekstein ja usein myös havainnollistettu kuvin, äänin, videoin ja animaation. Valtaosin www-sivustoja.
Sanastot	Virtuaalisia sanakirjoja ja tiettyyn sisältöön liittyviä sanastoja.
Oppaat	Käsikirjat ja tutoriaalit: oppaita konkreettisen toiminnan suorittamiseen. Esimerkiksi laitteiden käyttöohjeet, koulujen hakuoppaat ja toimintaohjeet (esim. työsuorituksen ohjaukseen).
Kokeiluaihiot	Vuorovaikutuksellisia simulaatioita ja mallinnustyökaluja, joilla mallinnetaan todellisia tai kuvitteellisia asioita tai prosesseja.
Avoin toiminta	Erilaisia avoimia tehtäviä ja luovia harjoituksia, joissa opiskelijan toiminnat ja niiden tulokset eivät ole tarkasti ennakoitavissa. Esimerkiksi tekstimuotoiset kysymykset ja tehtävät sekä sellaiset monipuoliset pelit, jotka eivät kuulu harjoitusohjelma-aihioihin.

Työvälineaihiot	Työkaluja, joilla käyttäjä voi luoda jotakin uutta, muokata jo luotua, tai olla vuorovaikutuksessa muiden kanssa. Esimerkiksi teksti-, kuva-, ääni- yms. editorit sekä chat, keskustelupalstat ja viestiohjelmat.
-----------------	---

2.2.2 Aihioille asetettavia vaatimuksia

Oppimisaihio on varsin atominen ja itsenäinen kokonaisuus, joka mahdollistaa niiden uudelleenkäytettävyyden (Silander 2003, 67). Aihoiden tuotannossa yksi keskeinen tavoite onkin niiden uudelleenkäytettävyys, eli aihioita on mahdollista käyttää erilaisissa asiayhteyksissä ja niitä voi yhdistellä toisiin materiaaleihin opetus- ja opiskelutilanteen mukaan (Ilomäki 2004, 15-16). Ilomäki antaa seuraavia tunnuspiirteitä opetuksellisesti hyvälle aihioille: Aihoiden on oltava *helposti löydettävissä*; tämä on perusedellytys aihoiden käytölle. Löydettävyyttä pyritään parantamaan aihioihin liitettyjen metatietojen avulla. Aihoiden laajamittaisen käytön edellytyksenä on, että ne ovat *saatavilla verkosta ilmaiseksi tai kohtuuhinnalla*. Tämä toteutuu usein jonkin portaalien kautta, esimerkkinä EU:n CELEBRATE-hankkeen oppimisaihioportaali.

Edelleen Ilomäen mukaan aihoiden pitäisi olla *ilman erityisiä teknisiä taitoja helposti käytettävissä*, jolloin opettaja pystyisi ottamaan aihiot käyttöön ilman erityistaitoja, ja vastaavasti oppijoiden olisi helppo käyttää niitä. Aihioita tulisi olla *mahdollista käyttää erilaisissa tilanteissa ja yhteyksissä*; tämän toteutumiseksi aihiot eivät saisi olla sidoksissa johonkin tiettyyn pedagogiseen malliin tai asiayhteyteen. Oppimisaihioiden perusluonteeseen kuuluu myös, että niiden tulisi olla *yhdistettävissä suuremmiksi kokonaisuuksiksi*. Lisäksi sisällön ajan tasalla pitämiseksi tärkeää on *päivitettyvyys*, jolloin aihio voidaan päivittää esimerkiksi sen tekijän tai aihiota hyödyntävän opettajan toimesta.

Silander (2004) luokittelee edellä luetellut aihoiden kriteerit aikaisemmin mainittujen kolmen näkökulman mukaisesti taulukossa 2. Esitetty luokittelu antaa hyvän kokonaiskuvan aihoiden uudelleenkäytettävyyden kriteereistä pedagogisten, sisällöllisten ja teknisten näkökulmien suhteen.

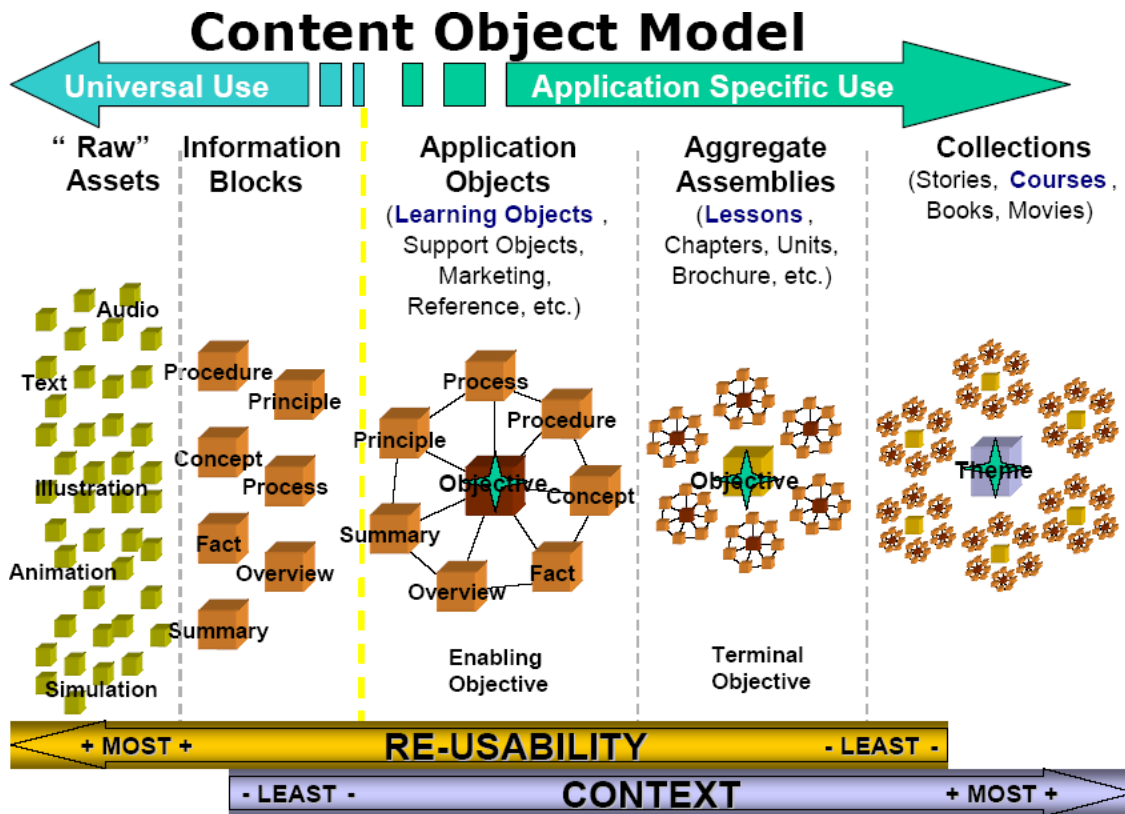
Taulukko 2. Oppimisaihion uudelleenkäytettävyyden kriteerit (Silander 2004).

	Sisältökonteksti	Pedagoginen konteksti	Tekninen konteksti
Kriteerit	1. Atominen ja itsenäinen kokonaisuus		
	2. Kontekstiriippumattomuus	5. Riippumattomuus pedagogisesta mallista	7. Monialustaisuus
	3. Useita sisältöesityksiä	6. Useita pedagogisia funktioita	8. Käytettävyys
	4. Päivitettävyys ja kumuloituva sisältö		9. Pedagoginen metatieto (kumulatiivinen)

Tässä työssä keskeinen kriteeri on taulukon kohta 7 *Monialustaisuus*, joka laajemmin käsitettynä tarkoittaa sisältöjen teknistä yhteensopivuutta (siirrettävyyttä) eri sovellusten ja järjestelmien välillä. Käytännön sisällöntuotannossa aihion tekijän olisi kuitenkin otettava kaikki mainitut kriteerit huomioon.

2.3 Oppimisaihion koko ja koostumus

Oppimisaihioilla ei ole mitään tiettyä ihannekokoa; pieniä aihioita on helpompi yhdistellä (joustavuus), kun taas laajempia aihioita (kokonaisia oppimateriaalipaketteja) voi olla helpompi hyödyntää (Jaakkola et al. 2004; Leeuwe 2001). Kuva 1 havainnollistaa, miten aihion koko vaikuttaa niiden uudelleenkäytettävyyteen ja kontekstiin. Kun pienempiä tietoyksiköitä kootaan suuremmiksi kokonaisuuksiksi, sitoutuminen kontekstiin kasvaa, eli niitä voidaan käyttää entistä harvemmissä asiayhteyksissä. Tämä heikentää aihion uudelleenkäytettävyyttä.



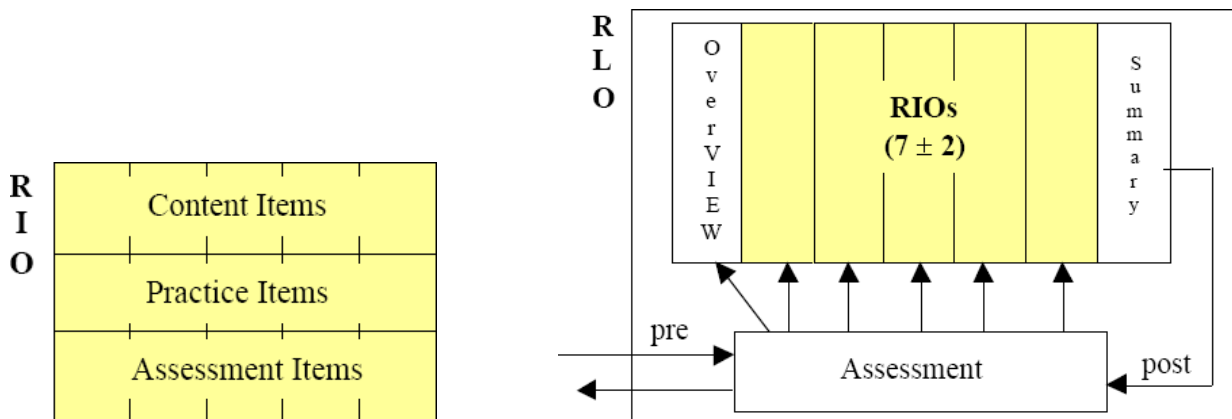
Kuva 1. Oppimisaihion koon vaikutus uudelleenkäytettävyyteen ja kontekstiin (Duval & Hodgins 2003).

Kuten kohdassa 2.1 *Määritelmä*, s. 3 kävi ilmi, niin oppimisasihioille on lukuisia erilaisia määritelmiä. Duval & Hodgins (2003) ovat pyrkineet ratkaisemaan tämän ongelman luomalla taksonomian, joka luokittelee oppimisasihiot niiden koosteasteen perusteella. Yleisimmin luokittelutasoja on esitetty eri lähteissä neljästä viiteen; kuvan 1 viisi tasoa ovat Duval & Hodginsin (2003) ja Masien (2003) mukaan seuraavat:

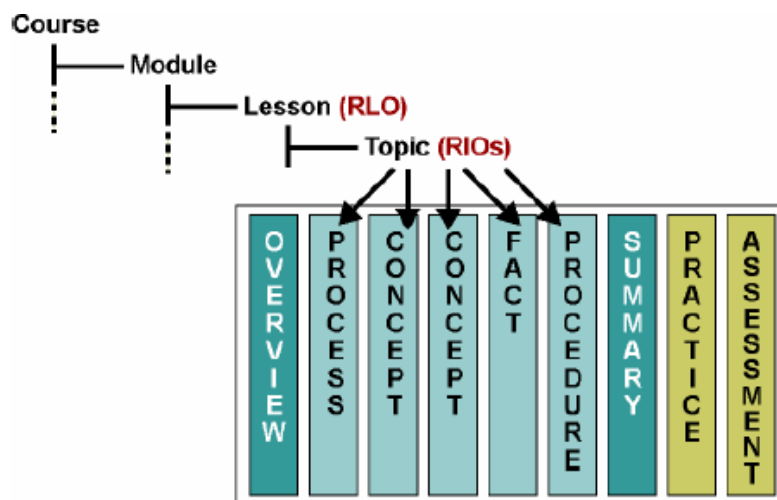
- *Medialementit* (Raw Assets, raakamateriaali) ovat oppimisasihion atomisimpia osasia, eli yksittäisiä digitaalisessa muodossa olevia kuvia, tekstejä, ääniä, videoita, animaatioita ja simulaatioita.
- *Sisältöobjektit* (Information Blocks) koostuvat erilaisista medialementeistä. Sisältö luokitellaan monissa lähteissä viiteen aihetyyppiin: *menettelytavat*, *periaatteet*, *käsitteet*, *prosessit* ja *faktat*. Kuvassa on lisäksi esitetty tyypit *yleiskatsaus* ja *yhteenveto*.
- *Sovellusalkohtaiset objektit* (Application Objects) on koottu sisältöobjekteista, ja mukana on asiasisällön lisäksi myös opetukselliset tavoitteet, harjoitukset ja palaute. Tämän koostetason objekteja kutsutaan Duval & Hodginsin mukaan oppimisasihioiksi.
- *Koosteet* (Aggregate Assemblies) ja *kokoelmat* (Collections) ovat edellisistä koottuja laajempia kokonaisuuksia muodostaen oppitunteja ja kursseja.

Kaksi ensimmäistä tasoa ovat sovellusalueesta riippumattomia, kun taas niitä seuraavat ylemmät tasot kohdistuvat tiettyyn opetusalueeseen. Uudelleenkäytettävyys saadaan varsinaisesti aikaiseksi kahta ensimmäistä tasoa hyödyntämällä. (Duval & Hodgins 2003)

Cisco Systemsin (2000) käyttämässä käsitteistössä sisältöobjekteja kutsutaan nimellä RIO (Reusable Information Object), joista RLO:t (Reusable Learning Objects) muodostuvat, kuva 2. Kukin RIO voi olla aiheyytältään joko *menettelytavat, periaatteet, käsitteet, prosessit* tai *faktat*. RIO sisältää sisällön lisäksi harjoitus- ja arviointielementit. Ciscon uusimmassa käsitteistössä RIO on pyritty korvaamaan topic-käsitteellä (aihe) ja RLO lesson-käsitteellä (oppitunti) (Cisco 2003). Kuvan 3 oppimisaihiohierarkia selvittää aihioajatusta yleistasolla, kyseessä ei ole pelkästään Ciscon käyttämä malli.



Kuva 2. RIO-RLO-rakenne (Cisco 2000).



Kuva 3. Oppimisaihiohierarkia (Cisco 2003).

2.4 Oppimisaihioita kohtaan esitettyä kritiikkiä

Muun muassa Jaakkola & Nirhamo (2003) ovat esittäneet kritiikkiä liian markkina- ja teknologislähtöiselle oppimisaihioajattelulle. Oppimisteknologian standardit ovat heidän mukaansa ensisijaisesti kiinnostuneita siitä, miten sisältöä voidaan helposti koostaa yhdistelemällä eri lähteistä peräisin olevaa materiaalia, eli miten saada aikaiseksi uudelleenkäytettäviä sisältöjä. On mahdollista rakentaa oppimisalustoja tai oppimateriaalin hallintajärjestelmiä, jotka voivat tarjota oppijalle sopivaa materiaalia sopivalla hetkellä. Tällaiset ns. mukautuvat oppimisjärjestelmät oppivat käyttäjistään ja pystyvät siten ennakoimaan, mitä oppija tarvitsee seuraavaksi.

Jaakkolan & Nirhamon havainto on, että mainitunlaisen sisällöntuotantotavan kannattajat ja mukautuvien oppimisjärjestelmien suunnittelijat eivät näytä tietävän, kuinka ihminen oppii; näiden tahojen toiminta perustuu virheellisiin ja vanhentuneisiin oppimiskäsityksiin. Teknologialähtöinen opetus esimerkiksi suosii esi- ja jälkitestejä yhdistettynä aikaisempaan käyttöhistoriaan, joiden perusteella järjestelmä jakaa oppijalle sopivan materiaalin. Oppija käy materiaalia läpi, kunnes saavuttaa vaaditun tason päästäkseen eteenpäin. Tämä tukee epäolennaisten asioiden muistamista, eikä anna oppijalle mahdollisuutta hallita omaa oppimistaan. Eri oppijoilla on erilaiset tarpeet, johon ”automatisoitu” opetus ei pysty vastaamaan.

Oppimateriaalin koostamisen idea toteutuu parhaiten silloin, kun oppimisaihioiden koko on suhteellisen pieni. Oppijan kannalta voi kuitenkin olla helpompi saada kokonaiskuva asiasta, jos oppimisaihion koko on suuri, esimerkiksi kurssin laajuinen. Hänellä saattaa olla vaikeuksia koota pienistä aihioista mielekästä kokonaisuutta. (Jaakkola & Nirhamo 2003). Aihiot on voitu toteuttaa eri työkaluilla, jolloin jokaisessa aihiossa saattaa olla esimerkiksi erilainen ulkoasu, esitystapa ja käsitteistö. Kussakin aihioissa voi vielä olla oma sisäinen navigointitapansa sen lisäksi, että oppimisalusta tarjoaa aihioille yleisen navigointitavan. Tällöin useista eri oppimisaihioista koostettu kokonaisuus saattaa pahimmillaan näyttää käyttäjälle yhdeltä sekasotkulta. (Bohl et al. 2002)

3 OPPIMISYMPÄRISTÖÖN LIITTYVIÄ TEKNISIÄ RATKAISUJA

3.1 Oppimisympäristö-käsite

Eräs keskeinen käsite verkko-opiskelussa on oppimisympäristö. Sillä tarkoitetaan paikkaa, tilaa, yhteisöä tai toimintakäytäntöä, joka edistää oppimista. Itse asiassa oppimisympäristö voi olla muukin kuin sähköinen ympäristö, vaikkapa luokkahuone. Oppimisympäristöä laajempi käsite on avoin oppimisympäristö, jossa pyritään saamaan aikaan joustavuutta ajan, paikan, menetelmien, toteutustapojen ja oppisisältöjen suhteen. Käytännössä tämä toteutuu parhaiten tietoverkossa, jossa olevaa oppimisympäristöä kutsutaan verkko-oppimisympäristöksi tai virtuaaliseksi oppimisympäristöksi. (Manninen 2000; Jaakkola et al. 2004, 36)

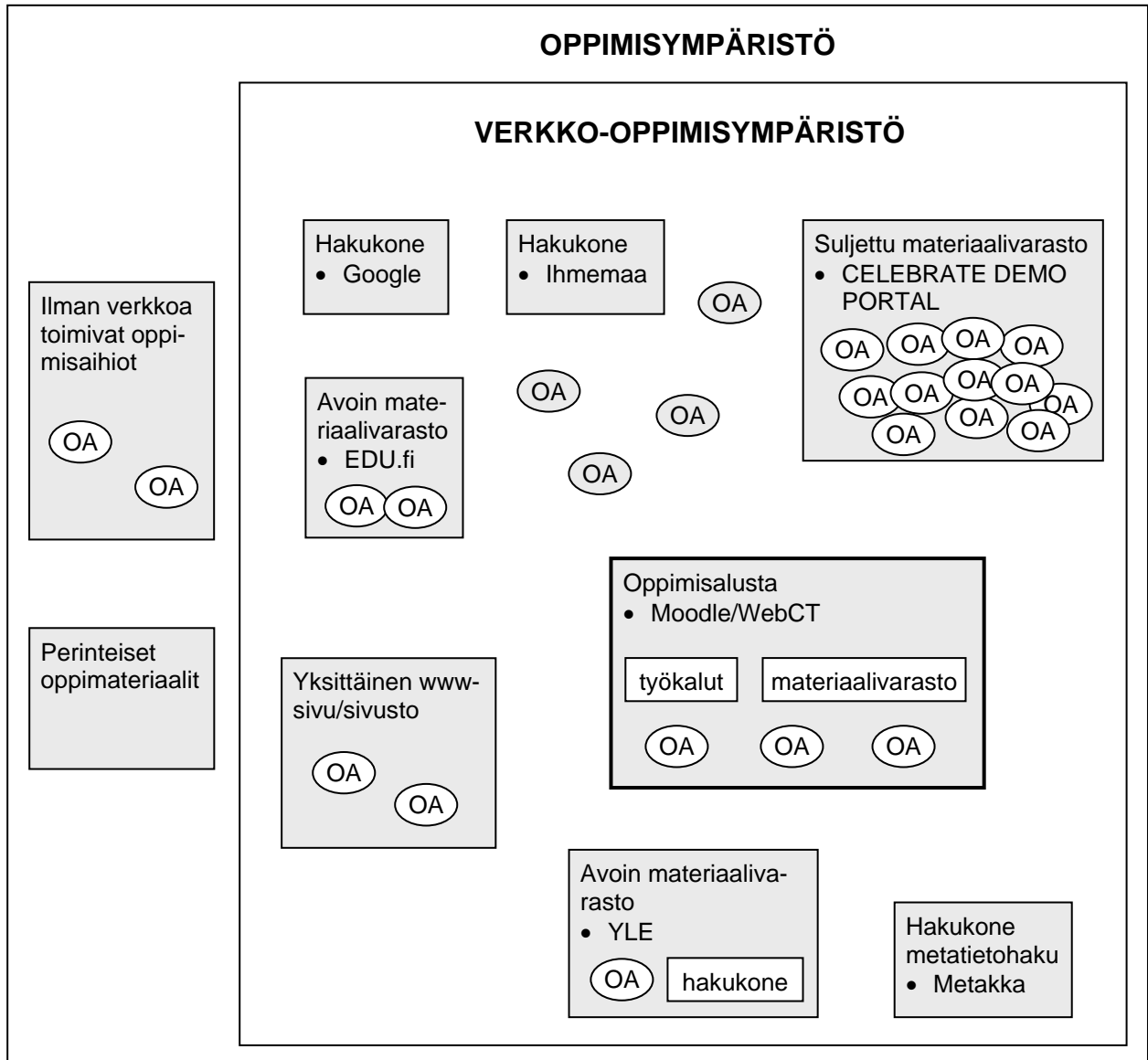
3.2 Oppimisalustat

Tämän työn kannalta keskeinen käsite on oppimisalusta, joka on yksi verkko-oppimisympäristön tärkeimmistä teknisistä apuvälineistä. Usein verkko-oppimisympäristöstä puhuttaessa tarkoitetaan virheellisesti oppimisalustaa, vaikka verkko-oppimisympäristöön kuuluu sen lisäksi sisältö, opiskelun ohjausprosessit, tutorointi ja vuorovaikutusprosessit (Saarinen 2002, 113). Oppimisalusta on internet-palvelimella oleva työryhmäohjelmisto, josta esimerkkeinä mainittakoon WebCT ja Moodle. Englanninkielessä oppimisalustaa kutsutaan tavallisimmin nimillä Learning Management System (LMS) tai Virtual Learning Environment (VLE).

Oppimisalustat sisältävät oppimateriaalin/oppimisasihioiden lisäksi erilaisia *kommunikointi-, muokkaus- ja hallinnointityökaluja* opettajien ja oppijoiden käyttöön (Jaakkola et al. 2004, 38). Kommunikointityökaluja ovat sähköposti, keskustelualueet ja chat, joilla käyttäjät kommunikoiivat keskenään. Muokkaustyökaluilla on mahdollista luoda, muokata ja järjestellä esimerkiksi oppimistehtäviä, testejä, oppimisasihoita tai muita oppimateriaalitiedostoja. Hallinnointityökalut mahdollistavat käyttäjien hallinnoinnin, kuten kurssille lisäämisen, ryhmien luomisen, käyttöaktiivisuuden seurannan sekä tehtävien ja testien tulosten tarkastelun/arvioinnin.

Oman kokemuksen mukaan oppimisalusta on keskeisin paikka, jossa verkko-opiskelun ohjaus ja työskentely tapahtuu. Työskentelyä, esimerkiksi oppimistehtävien tekemistä, tehdään luonnollisesti myös oppimisympäristön ulkopuolella, mutta oppimisympäristö antaa työskentelylle puitteet kuten luokkahuone perinteisessä opiskelussa. Kurssin materiaali voi olla alustan sisällä tai ulkopuolella: oppimisalustalta voidaan tehdä linkkejä erilaisiin alustan ulkopuolisiin materiaali-varastoihin ja oppimisasihioihin, tai oppimisasihoita voidaan viedä alustan omalle tiedostoalueel-

le. Olipa materiaali missä tahansa, niin periaate on, että se on aina käyttäjien saatavilla – käytännössä internetin kautta. Kuva 4 selvittää oppimisympäristöjen, oppimisalustan ja aihoiden suhdetta toisiinsa.



Kuva 4. Oppimisasihot suhteessa oppimisalustaan ja oppimisympäristöihin. Lähteen Jaakkola et al. (2004, 37) kuvaa mukaellen.

Monet oppimisalustat tukevat yleisimpiä määrämuotoisia oppimisasihoita, kuten IMS Content Packaging -spesifikaation ja SCORM-referenssimallin mukaisesti koostettuja materiaalipaketteja, joita käsitellään tarkemmin luvussa 5 *SISÄLLÖNTUOTANTOON LIITTYVIÄ MÄÄRITYKSIÄ*. Periaatteena on, että kurssiympäristön tiedostoalueelle voi tuoda tällaisessa muodossa olevia aihioita, ja vastaavasti kurssilla olevan materiaalin voi tallentaa standardimuotoiseksi paketiksi muualle siirrettäväksi tai muokattavaksi.

3.3 Oppimateriaalin hallintajärjestelmät

Oppimateriaalin hallintajärjestelmät (Learning Content Management Systems, LCMS) sisältävät samoja ominaisuuksia kuin oppimisalustat, mutta painopiste on oppimateriaalin hallinnassa. Sisältömateriaali on oppimisaihiomuotoista, ja sitä voidaan hallita aihoiden alimmalta koostetasolta eli yksittäisistä medialementeistä lähtien, vrt. kuva 1. LCMS:n ydinkomponenttina on oppimisaihiovarasto (learning object repository). Tässä tietokannassa olevista medialementeistä voidaan koostaa eri laajuisia, oppijoiden yksilöllisiä tarpeita vastaavia, oppimisaihioita. Monet LCMS:t sisältävät työkaluja sisällön luomiseen ja koostamiseen. Koostamistyökalut (authoring applications) helpottavat ja automatisoivat koostamisprosessia eli kurssisisältöjen luomista oppimisaihioista esimerkiksi valmiiden suunnittelupohjien (templates) avulla. (Greenberg 2002)

Oppimisalustan (LMS) ja LCMS:n eroa on vaikea määrittää täsmällisesti, joka vaikeuttaa olemassa olevien järjestelmien jakamista näihin luokkiin. Voidaan kuitenkin todeta, että varsinaiset oppimateriaalin hallintajärjestelmät ovat Suomessa perin harvinaisia. Tällaisiksi järjestelmiksi ovat ehkä luettavissa Suomessakin yleisen WebCT:n tietyt versiot: WebCT Vista ja WebCT 6 täydennettynä Vistan Learning Object Manager -moduulilla.

Suomalaisesta Optima-alustastakin löytyy oppimateriaalin hallintajärjestelmille tyypillisiä piirteitä. Optiman ominaisuuksiin kuuluvat tavanomaisten oppimisympäristöominaisuuksien lisäksi sisällöntuotantotyökalut, kuten tekstieditori ja web-editori html-sivujen tekemiseen. Optimassa on myös materiaalikirjasto, jossa olevaa materiaalia voidaan linkittää tai viedä eri kursseille, joita Optimassa kutsutaan työtiloiksi. (Discendum 2005)

3.4 Oppimateriaalipankit

Kun käytettävissä on pelkkä oppimisalusta ilman oppimateriaalin hallintajärjestelmien kaltaisia sisällönhallintaominaisuuksia, eräs käyttökelpoinen ratkaisu on ulkoinen materiaalivarasto eli oppimateriaalipankki (learning content/object repository). Materiaalipankkeja on tehty lukuisia erilaisia, ja niiden toiminnallisuus voi alkaa yksinkertaisesta kuvakirjastosta ulottuen aina oppimateriaalin hallintajärjestelmien kaltaisiin ominaisuuksiin. Eri lähteiden perusteella (mm. Masie 2003) oppimateriaalipankki voidaan määritellä tietokantapohjaiseksi tallennusalueeksi, jota käytetään medialementtien, oppimisaihioiden ja niitä laajempienkin kokonaisuuksien organisoimiseen, ja jossa sisällöt on luokiteltu metatietomerkitöjen avulla löytämisen helpottamiseksi.

Monet oppimateriaalipankit on toteutettu sillä periaatteella, että käyttäjät voivat viedä tekemäänsä oppimateriaalia sinne, ja tässä yhteydessä he itse syöttävät sisältöä luonnehtivat metatietokuvaukset järjestelmään. Käyttäjät voivat vastaavasti etsiä sisältöä materiaalipankista ja tehdä niistä koosteita, jotka on mahdollista tallentaa pankkiin uusiksi objekteiksi esimerkiksi IMS- tai SCORM-paketteina myöhempää käyttöä varten. Halutut yksittäiset tiedostot tai koostepaketit ovat tallennettavissa omalle koneelle edelleen oppimisolustalle vietäväksi.

4 STANDARDOINTI

4.1 Standardoinnin etuja

Kohdassa 2.2.2 *Aihioille asetettavia vaatimuksia s. 6* kuvattiin edellytyksiä opetuksellisesti hyvälle oppimisaihioille. Asetettuihin vaatimuksiin pyritään vastaamaan oppimisjärjestelmien, työkalujen ja sisältöjen standardoinnilla, jolla voidaan saavuttaa seuraavia muun muassa Masien (2003) luettelemaa etuja. *Yhteentoimivuus* (interoperability) mahdollistaa sen, että sisällöt ovat siirrettävissä eri sovellusten ja järjestelmien välillä. Sovellukset voivat olla esimerkiksi sisällöntuotantovälineitä ja testikysymyseditoreja, järjestelmät oppimisalustoja tai oppimateriaalipankkeja. Sisältöobjektien on myös oltava sillä tavalla keskenään yhteensopivia, että eri työkaluilla luotuja sisältöjä on mahdollista yhdistää uusiksi oppimisaihioiksi (Duval & Hodgins 2003). *Uudelleenkäytettävyyden* (re-usability) ansiosta sisältömateriaalista voidaan koostaa helposti uusia kokonaisuuksia, ja sisältöjä voidaan käyttää eri asiayhteyksissä. Aihion *saatavuudella* (accessibility) puolestaan tarkoitetaan sitä, että sisältö on helposti löydettävissä ja edelleen käyttäjille jaettavissa. Tähän päästään metatietostandardeja soveltamalla.

Edelleen, standardoinnilla voidaan pienentää kustannuksia (*edullisuus*, affordability), koska standardointi varmistaa sen, että investointi kyseiseen teknologiaan on mahdollisimman riskitöntä. *Hallittavuuden* (manageability) ansiosta oppimisalusta tms. järjestelmä saa tietoa muun muassa siitä, missä kohtaa oppija etenee oppimateriaalissa ja mitä tuloksia oppija on saanut testeistä. Hallittavuuteen liittyy myös mahdollisuus hallita laajaa materiaalivalikoimaa ja koostaa sieltä kuhunkin tarpeeseen sopivia kokonaisuuksia. Standardointi varmistaa *pysyvyyden / jatkuvuuden* (durability), sillä järjestelmän hankkijan ei tarvitse olla jonkun tietyn toimittajan oman teknologian varassa. Taustalla oleva teknologia voi muuttua, mutta standardoinnin ansiosta sisältöjä ei tarvitse rakentaa uudelleen. *Skaalautuvuus / laajennettavuus* (scalability) puolestaan antaa mahdollisuuden laajentaa toimintaa ja palvella entistä suurempia käyttäjämääriä. (Masie 2003)

Standardoinnista on siis hyötyä kaikille verkko-opetukseen liittyville toimijoille: työkalujen ja järjestelmien valmistajille, sisällöntuottajille ja opettajille. Oppimisaihioita kohtaan esitetystä kritiikistä huolimatta standardoinnista toivon mukaan olisi hyötyä myös oppijoille esimerkiksi käytettävyydeltään yhdenmukaisina sisältöinä.

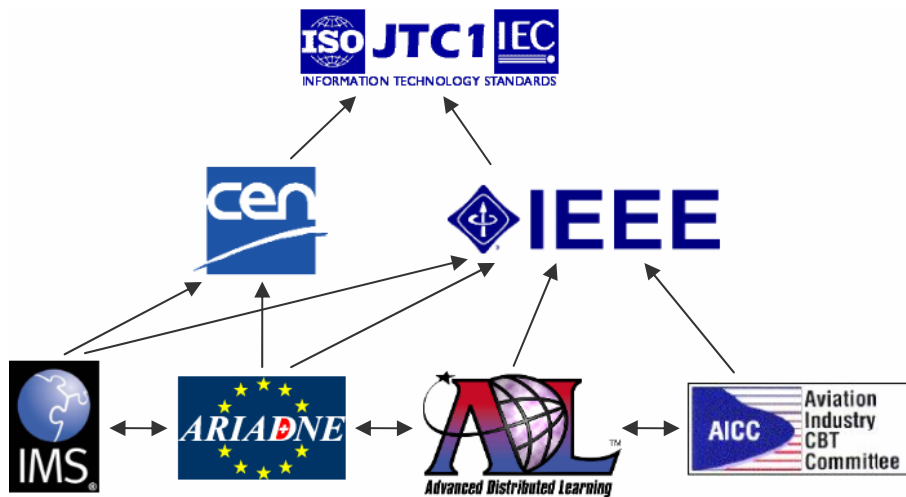
4.2 Spesifikaatioiden ja standardien tuottajat

Opetukseen liittyvän sisällöntuotannon standardointi alkoi vuonna 1988, kun Aviation Industry CBT Committee eli AICC perustettiin yhdenmukaistamaan lentokoneollisuuden tietokoneavusteista opetusta. Myöhemmin standardointityö siirtyi internet-pohjaiseen opetukseen, ja tällä hetkellä aiheen parissa työskentelee useita toisistaan erillisiä ryhmiä. (Fallon & Brown 2003).

Tie varsinaiseksi standardiksi on pitkä ja monivaiheinen, joten tällä hetkellä varsinaisia aihepiiriin liittyviä standardeja on vain IEEE:n LOM (Learning Object Metadata). Sen lisäksi on kuitenkin olemassa joukko niin sanottuja spesifikaatioita eli määrityksiä, joista voi jossain vaiheessa muodostua standardeja. Seuraavassa on kuvattu idean vaiheet matkalla standardiksi (Masie 2003, 12-13). Standardointiprosessi on esitetty myös kuvassa 5.

1. *T&K*: Tehdään tutkimus- ja kehitystyötä ratkaisujen löytämiseksi, esimerkkinä tutkimustyö yliopistoissa ja yrityksissä.
2. *Spesifikaation kehittäminen*: Kun alustavalla ratkaisumallilla nähdään olevan mahdollisuuksia, kirjoitetaan yksityiskohtainen määrittelydokumentti eli spesifikaatio, jonka pohjalta on mahdollista tehdä toteutus. Spesifikaatioiden laatimista varten on perustettu erinäisiä yhteistyöelimiä, kuten AICC, IMS ja ARIADNE.
3. *Kokeilu ja testaus*: Spesifikaatioista laaditaan referenssimalli ja toteutukset. Toteutukset laitetaan testaukseen, jotta nähdään, mikä toimii, mikä ei, mitä puuttuu, mitkä ovat käyttäjien reaktiot jne. Esimerkiksi ADL SCORMin Plugfest-testaustapahtumissa eri toimijat voivat testata välineidensä ja sisältöjensä yhteensopivuutta SCORM-referenssimalliin sekä muihin tuotteisiin nähden (Fallon & Brown 2003, 34).
4. *Virallinen kansainvälisen standardin asema*: testatut ja jotakuinkin valmiit spesifikaatiot arvioidaan akkreditoitun standardointielimen toimesta. Jos spesifikaatio kelpuutetaan, se saa virallisen standardin aseman, ja on saatavilla kyseisen standardointielimen kautta. Opetusteknologia-alan standardointia suorittavia tahoja ovat IEEE LTSC, ISO/IEC JTC1/SC36 ja CEN/ISSS/WS-LT.

Kaikki standardointiprojektit, komiteat ja työryhmät ovat tavalla tai toisella yhteydessä toisiinsa, joko virallisten tai epävirallisten suhteiden kautta. Esimerkkinä jälkimmäisistä suhteista on se, että monissa tapauksissa samat henkilöt työskentelevät useammassa työryhmässä, joten tiedonvaihto niiden välillä on merkittävää. Yhdellä taholla hyväksytyt mallit ja spesifikaatiot tulevat helposti käyttöön otetuksi myös muillakin tahoilla. (Leeuwe 2001). Eri organisaatioiden välillä on myös selvät ”työnjaot”, joita kuva 5 standardointiprosessista selventää.



Kuva 5. Standardointiprosessi ja siihen liittyvät tahot (Duval 2004; Leeuwe 2001).

Mainittakoon tässä yhteydessä, että useimpien oppisisältöihin liittyvien standardien ja spesifikaatioiden taustateknologia on XML-pohjaista.

Sovellusprofiilit

Standardien ja spesifikaatioiden ongelmana on usein se, että ne ovat yleisellä tasolla, eivätkä siten suoraan sovellu kaikille käyttäjätahoille. Esimerkiksi metatietoa käytetään niinkin erilaisten toimijoiden kuten kirjastojen, julkaisutoiminnan, koulutuksen ja yritysten sisällönhallinnassa. LOM-standardiin onkin pyritty lisäämään kenttiä kaikkien tahojen tarpeita varten, mutta standardi ja sen sanasto on silti liian yleisellä tasolla erityissovelluksiin. Kun lisäksi kaikki kentät ovat valinnaisia, niitä ei välttämättä käytetä ollenkaan. Tämä on johtanut jotkut yhteisöt tai käyttäjätahot, kuten ADL, luomaan omat LOM-profiilinsa. Profiileilla pyritään lisäämään spesifisyyttä ja parantamaan yhteensopivuutta (interoperability) tietyn käyttäjätahon sisällä, mutta samalla yleinen yhteensopivuus säilyttäen. (Reusable Learning 2004a). Esimerkiksi metatietoprofiili voi sisältää suosituksia siitä, mitkä kentät vaaditaan, mitkä ovat valinnaisia, mitä ei käytetä ja mitä käsitteitä sanastoissa käytetään (Peace River 2004).

Referenssimallit

Standardit ja spesifikaatiot eivät suoraan kerro, miten joku asia pitäisi toteuttaa. Käytäntöön soveltamista varten onkin tehty referenssimalleja, jotka helpottavat ymmärtämään sitä, miten eri standardit ja spesifikaatiot sovitetaan yhteen ja kuinka niitä on tarkoitettu käytettäväksi. Opetusteknologiaan liittyvistä referenssimalleista tunnetuin on Sharable Content Object Reference Model eli SCORM. (Reusable Learning 2004a)

Seuraavissa kohdissa on kerrottu tarkemmin keskeisimmistä opetusteknologia-alan standardointityötä tekevästä ryhmittymisestä ja toimielimistä sekä niiden tärkeimmistä tuotoksista. Kohtien tarkoituksena on tuoda esille sitä, mistä keskeisimmät määritykset juontavat juurensa ja mikä on niiden suhde toisiinsa. Tämä on johdantoa lukuun 5 *SISÄLLÖNTUOTANTOON LIITTYVIÄ MÄÄRITYKSIÄ*, jossa kuvataan tärkeimpiä tämän työn aihepiiriin liittyviä spesifikaatioita ja standardeja.

4.2.1 AICC - Aviation Industry CBT Committee

Aviation Industry CBT (Computer-Based Training) Committee eli AICC on kansainvälinen teknologiapohjaisen koulutuksen ammattilaisista koostuva yhteisö. AICC tekee ohjeita ja suosituksia ilmailualan teollisuudelle tietokoneavusteisen opetuksen (TAO, CBT) ja siihen liittyvän teknologian kehittämistä, jakelusta ja arvioinnista. Tavoitteena on edistää TAO:n taloudellista ja tehokasta toteutusta yhteensopivuus huomioiden. (AICC 2004)

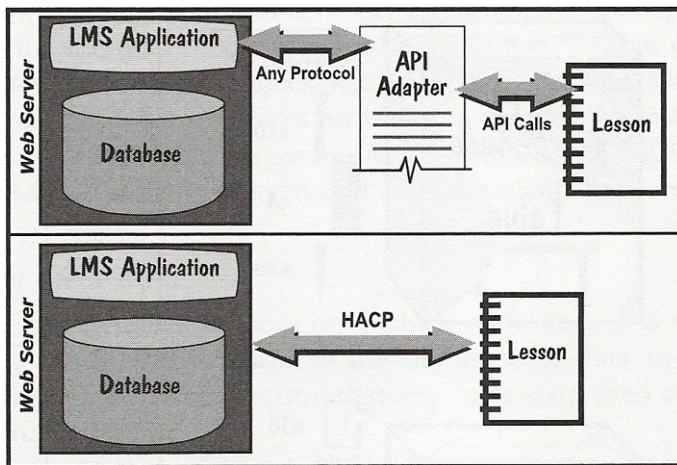
AICC on tehnyt yhdeksän teknistä AGR-suositusta (AICC Guidelines & Recommendations). Kukin AGR on lyhyt dokumentti, joka viittaa tarvittaessa yksityiskohtaiseen tekniseen spesifikaatioon. Monet AICC:n suosituksista on melko vanhoja, jopa 1990-luvun alkupuolelta. Käytetty terminologia ja opetusteknologiat (esim. CBT ja CMI) ovat 1990-luvulta, eivätkä parhaalla tavalla edusta tämän hetken suuntauksia. Tämä on ymmärrettävissä, kun muistetaan, että ilmailuteollisuus on tuottanut koulutusmateriaalia jo 80-luvulta lähtien; mukana on siis jopa DOS-sovelluksia. Silti AGR-suositukset ja niiden spesifikaatiot kattavat sekä perinteisen tietokoneavusteisen opetuksen (CD-ROM- ja lähiverkkokäyttö) että nykyisen internet-pohjaisen opetusteknologian hyödyntämisen. Pitkän kehitystyönsä tuloksena AICC onkin saanut aikaan spesifikaatioita, joita monet muut opetusteknologia-alan ryhmittymät ja standardointielimet ovat nyt alkaneet omaksua ja ovat sovittamassa käyttöönsä (Hodgins & Conner 2000).

AICC:n käsitteistössä oppimisalusta on CMI-järjestelmä (Computer Managed Instruction). Yksi tärkeimmistä spesifikaatioista on *CMI Guidelines For Interoperability*, joka määrittää kolme oppimisaihion (Assignable Unit) ja oppimisalustan (CMI) välistä tiedonvaihtotapaa (CMI001 2004):

- Paikallinen tiedostopohjainen tiedonvaihto, jossa aihiot ja niiden hallintajärjestelmä sijaitsevat joko kiintolevyllä tai lähiverkossa.
- HTTP-pohjainen ns. HACP-tiedonvaihto (HTTP-based AICC CMI Protocol), joka perustuu HTTP-viestien (request/response) vaihtamiseen.

- Tiedonvaihto API-rajapinnan (Application Programming Interface) kautta.

Tiedonvaihtotavoista kiinnostavimmat ovat kaksi jälkimmäistä, jotka hyödyntävät internetiä, kuva 6. Kaikki AICC-yhteensopivat oppimisaihiot sisältävät ohjelmakoodia, jotta aihion ja oppimisalustan välinen kommunikointi olisi mahdollista. HTTP-pohjaisen kommunikoinnin ohjelmointi on sisällöntuottajalle hankalahkoa, joten AICC kehitti yhdessä ADL:n kanssa API-rajapinnan, joka käyttää JavaScript-kutsuja. Huomattakoon, että ADL:n oma SCORM-referenssimalli käyttää myös tätä samaa API-rajapinnan kautta tapahtuvaa tiedonvaihtotapaa. (Fallon & Brown 2003). SCORM ja sen tiedonvaihto esitellään tarkemmin kohdassa 5.4 *SCORM-referenssimalli*.



Kuva 6. AICC-spesifikaation mukaiset API- ja HACP-tiedonvaihtotavat (Fallon & Brown 2003).

4.2.2 ARIADNE - Alliance of Remote Instructional Authoring & Distribution Network for Europe

ARIADNE (Alliance of Remote Instructional Authoring & Distribution Network for Europe) ja ARIADNE II ovat Euroopan Unionin projekteja vuosilta 1996-2000. Niiden tavoitteena oli kehittää työkaluja ja menetelmiä tietokonepohjaisten oppikurssien tuottamiseen, hallintaan ja uudelleenkäyttämiseen. Projektit ovat päättyneet, mutta kehitettyjen työkalujen ylläpitoa, koulutusta ja edelleen kehitystä jatkaa ARIADNE Foundation. (ARIADNE 2004)

ARIADNE on tehnyt merkittävää työtä verkko-opetuksen metatietomäärittysten eteen. ARIADNE ja IMS Global Learning Consortium tekivät tahoillaan metatietoesifikaatiota, kunnes vuonna 1998 antoivat IEEE:lle yhteisen spesifikaation, joka muodosti merkittävän pohjan LOM-standardille (IMS 2005). Työkalupuolelta ARIADNEn aikaansaannoksista mainittakoon Knowledge Pool System, KPM, joka on oppimisaihioita ja niitä kuvaavaa metatietoa sisältävä oppima-

terialipankki. Metatieto KPM:n sisältämistä resursseista on kenen tahansa nähtävissä, mutta varsinainen sisältö on vain jäsenten käytössä. (ARIADNE 2004)

4.2.3 IMS Global Learning Consortium

IMS (Instructional Management System) Global Learning Consortium on merkittävä toimija, joka kehittää spesifikaatioita ja edistää niiden käyttöönottoa etäopetuksessa/-opiskelussa. Spesifikaatiot kattavat sekä internet-pohjaisen että ns. off-line-opiskelun (esim. CD-ROMit), ja ovat organisaation www-sivuilla julkisesti kaikkien saatavilla. Spesifikaatioita on tehty seuraavia toimintoja varten: oppisisältöjen etsiminen ja käyttäminen, oppijan edistymisen seuranta ja suoritusten raportointi sekä suoritustietojen tiedonvaihto hallintajärjestelmien välillä. (IMS 2005; Hodgins & Conner 2000)

IMS Global Learning Consortiumilla on kaksi päätavoitetta. Ensimmäkin tavoitteena on määrittellä tekniset standardit sovellusten ja palvelujen yhteentoimivuudelle etäopetuksessa. Toiseksi halutaan tukea IMS-spesifikaatioiden laajaa käyttöönottoa, jotta eri valmistajien etäopetusympäristöt ja sisällöt saataisiin yhteensopiviksi keskenään. (Hodgins & Conner 2000)

IMS on julkaissut kymmenen spesifikaatiota, ja viisi on lisäksi työn alla. Oppimisaihioihin ja niiden uudelleenkäytettävyyteen liittyvät tärkeimmät spesifikaatiot ovat Meta-Data, Content Packaging ja Question and Test Interoperability, jotka kuvataan tarkemmin luvussa 5 *SISÄL- LÖNTUOTANTOON LIITTYVIÄ MÄÄRITYKSIÄ*. Näiden lisäksi mainittakoon Simple Sequencing, jonka mukaisesti tehdyssä sovelluksessa eteneminen voi haarautua oppijan ja sisällön vuorovaikutuksessa tiettyjen sääntöjen perusteella. Spesifikaatio määrittää vaaditut käyttäytymistavat ja toiminnallisuuden, jotka yhteensopivassa järjestelmässä täytyy olla oppimisaihioiden järjestämiseksi esitettävään muotoon.

4.2.4 ADL - Advanced Distributed Learning Initiative

Advanced Distributed Learning (ADL) Initiative on USA:n puolustusministeriön ja sen yksityis- ja julkissektoreiden kumppanien aloite, jonka tavoitteena on saada aikaiseksi yleinen tekninen kehys (framework), joka mahdollistaa uudelleenkäytettävien oppimisaihioiden ja tietokoneiden välisen yhteentoimivuuden internetin kautta (Masie 2003).

ADL Initiative kehittää spesifikaatioita ja ohjeistuksia, ja soveltaa niitä käytäntöön. Tunnetuin ADL:n spesifikaatio on *Sharable Content Object Reference Model* (SCORM). SCORM-

referenssimalli määrittää kurssikomponenttien väliset riippuvuudet, tietomallit ja protokollat; tämä mahdollistaa oppimisaihoiden siirrettävyyden kaikkien samaa mallia tukevien järjestelmien välillä. (ADL 2005)

SCORM-malli on koostettu useasta toisiinsa liittyvästä spesifikaatiosta, joista monet ovat peräisin aikaisemmin kuvattujen työryhmien tekemästä työstä. SCORM-metatietosovellusprofiili pohjautuu IEEE LOM -standardiin ja IMS Learning Resource Meta-Data -spesifikaatioon. SCORM Content Packaging -sovellusprofiili puolestaan on peräisin IMS Content Packaging -spesifikaatiosta laajennettuna joillakin SCORM-spesifisillä elementeillä. SCORM-runtime-ympäristön komennot ja API-sovitin taas pohjautuvat suoraan AICC:n CMI Guidelines for Interoperability -spesifikaatioon. (ADL 2005; Bailey & Currier 2003). SCORM-mallia käsitellään tarkemmin kohdassa *5.4 SCORM-referenssimalli*.

4.2.5 Standardointiorganisaatiot

Useimmat spesifikaatiot jäävät elämään spesifikaationa. Jotkut niistä voidaan kuitenkin viedä standardiksi asti, jolloin ne saavat virallisen ja kansainvälisen hyväksynnän. Tämä tapahtuu jonkin valtuutetun standardointielimen toimesta, joka arvioi spesifikaation kelpoisuutta standardiksi. Spesifikaatio on standardointikäsitteeseen tullessaan jo varsin valmis ja testattu, ja standardointiprosessissa siitä tehdään yleiskäyttöinen poistamalla kaikki mahdollinen johonkin tiettyyn taho-oon viittaava tieto (Masie 2003, 13).

Eräs näistä organisaatioista on IEEE (Institute of Electrical and Electronics Engineers) Learning Technology Standards Committee, joka koostuu erillisistä opetuksen ja koulutuksen alueelle tietoteknisiä standardeja kehittävästä työryhmistä. Nämä ryhmät tuottavat teknisiä standardeja, suositeltavia toimintatapoja ja ohjeita sovellettaviksi sähköiseen oppimiseen liittyvissä ohjelmissa, työkaluissa, teknologioissa ja suunnittelumetodeissa. (Fallon & Brown 2003)

IEEE LTSC:n Learning Object Metadata (LOM) -standardi on ensimmäinen akkreditoitu standardi opetusteknologia-alalla. Se pohjautuu IMS- ja ARIADNE-projektien omilla tahoillaan tekemälle metatietomallin määrittämiselle. IEEE LTSC -työryhmän 11 (WG11) työstämät Computer Managed Instruction -standardit puolestaan perustuvat AICC:n CMI Guidelines For Interoperability -spesifikaatiolle. IEEE LTSC:n mukaan on laaja yksimielisyys siitä, että kyseisen spesifikaation tietomallissa määritetty sisältöobjektien tiedonvälitys on laajasti sovellettavissa oppimisalustoissa tms. oppimisen hallintajärjestelmissä. (IEEE LTSC 2004). Kyseessä on aikai-

semmin mainittu API-rajapinta, jota käytetään AICC- ja SCORM-yhteensopivissa oppimisaihoissa ja -alustoissa.

Toisena mainittakoon CEN/ISSS-standardointiorganisaatio (European Committee for Standardization – Information Society Standardization System), jossa toimii WS-LT-oppimisteknologia-työryhmä (Work Shop on Learning Technology). WS-LT osallistuu kansainvälisiin oppimisteknologian standardointihankkeisiin tavoitteenaan varmistaa eurooppalaisen näkökulman huomiointi niissä. WS-LT tekee spesifikaatioita, ohjeita ja suosituksia ainoastaan silloin, kun tiettyyn tarpeeseen ei sellaista vielä ole olemassa, tai muualla tehty ratkaisumalli olisi lokalisoitava Euroopan vaatimuksia vastaavaksi. Pyrkimyksenä on siis välttää päällekkäistä työtä muiden kanssa. (CEN/ISSS 2004)

IEEE LTSC:n kehittämät standardit etenevät kansainvälisiksi standardeiksi ISO:n (International Organization for Standardization) toimesta. ISO on luonut yhdessä sähkö- ja elektroniikka-alan standardeja tekevän IEC:n (International Electrotechnical Commission) kanssa Joint Technical Committeeen (JTC). Tämä tekninen komitea, joka tunnetaan nimellä JTC1, sisältää alakomitean SC36, joka vastaa opetukseen ja koulutukseen liittyvän tietotekniikan standardointityöstä. (ISO/IEC 2005; Fallon & Brown 2003)

5 SISÄLLÖNTUOTANTOON LIITTYVIÄ MÄÄRITYKSIÄ

Tässä luvussa kuvataan sellaisia tärkeimpiä sisällöntuotannon standardeja ja spesifikaatioita, jotka liittyvät metatiedon, kysymysten ja testien kuvaamiseen sekä sisällön paketoimiseen.

5.1 Oppimisaihioiden metatiedon kuvaaminen

Yleisen määritelmän mukaan metatieto (metadata) on tietoa tiedosta. Esimerkiksi tekstiasiakirja on tietoa, ja kaikki sitä kuvaava tieto on metatietoa, kuten tekijä, luontipäivämäärä, aihe, sisältöä luonnehtivat avainsanat jne. Metatiedolla on tarkoitus kuvata kohde niin hyvin, että kuvauksen perusteella sisältö on ymmärrettävissä ilman siihen tutustumista (Nirhamo 2004). Opetuskäytössä metatiedon avulla voidaan kuvata oppimisaihoita, jolloin materiaalipankissa olevia aihioita on mahdollista etsiä halutuilla hakuehdoilla.

Jotta metatietokuvauksista olisi käytännön hyötyä, tarvitaan yhteinen sovittu kuvaustapa. Tätä varten on tuotettu useita metatietomäärittämiä, joista yksi on opetusteknologia-alalle tarkoitettu standardi IEEE 1484.12.1-2002. Kyseinen standardi tunnetaan paremmin nimellä Learning Object Metadata Standard eli LOM. Sen kehitystyö lähti käyntiin ARIADNE- ja IMS-organisaatioiden yhteistyönä, mutta pian työ annettiin kokeneen standardointiorganisaation IEEE LTSC:n loppuunsaattavaksi (Nirhamo 2002). LOM:n perusrakenne on puumainen hierarkia koostuen yhdeksästä pääryhmästä eli luokasta. Luokat toimivat ryhmittelevinä elementteinä sisältäen lisää ryhmitteleviä alielementtejä sekä varsinaiset tietoalkiot. Tietoalkioita eli täytettäviä kenttiä on noin 60. LOM:n perusrakenne on esitetty kuvassa 7 standardin suomenkielisen version sanaston mukaisesti. Standardissa on tarkat kuvaukset kunkin elementin käyttötarkoitukselle sekä englanninkieliset nimet elementeille.

<ul style="list-style-type: none"> 1 Yleistä <ul style="list-style-type: none"> 1.1 Tunniste <ul style="list-style-type: none"> 1.1.1 Luettelo 1.1.2 Merkintä 1.2 Nimeke 1.3 Kieli 1.4 Kuvaus 1.5 Avainsana 1.6 Kattavuus 1.7 Rakenne 1.8 Koosteisuuden taso 2 Elinkaari <ul style="list-style-type: none"> 2.1 Versio 2.2 Tila 2.3 Osallistuja <ul style="list-style-type: none"> 2.3.1 Rooli 2.3.2 Taho 2.3.3 Päiväys 3 Tietoa metatiedosta <ul style="list-style-type: none"> 3.1 Tunniste <ul style="list-style-type: none"> 3.1.1 Luettelo 3.1.2 Merkintä 3.2 Osallistuja <ul style="list-style-type: none"> 3.2.1 Rooli 3.2.2 Taho 3.2.3 Päiväys 3.3 Metatietorakenne 3.4 Kieli 	<ul style="list-style-type: none"> 4 Tekniset ominaisuudet <ul style="list-style-type: none"> 4.1 Tallennusmuoto 4.2 Koko 4.3 Sijainti 4.4 Vaatimukset <ul style="list-style-type: none"> 4.4.1 Vaatimusvaihtoehdot <ul style="list-style-type: none"> 4.4.1.1 Tyyppi 4.4.1.2 Nimi 4.4.1.3 Minimiversio 4.4.1.4 Maksimiversio 4.5 Asennusohjeet 4.6 Muut alustavaatimukset 4.7 Kesto 5 Opetuksellisia ominaisuuksia <ul style="list-style-type: none"> 5.1 Vuorovaikutustyyppi 5.2 Oppimateriaalin laji 5.3 Vuorovaikutteisuuden aste 5.4 Asiasisällön tiiviyys 5.5 Kohderyhmä 5.6 Opintojen luonne 5.7 Kohderyhmän ikä 5.8 Vaikeustaso 5.9 Opiskeluun kuluva aika 5.10 Käyttötapakuvaus 5.11 Kohderyhmän kieli 	<ul style="list-style-type: none"> 6 Oikeudet <ul style="list-style-type: none"> 6.1 Maksut 6.2 Tekijänoikeudet ja muut rajoitukset 6.3 Kuvaus 7 Suhde muihin oppisisältöihin <ul style="list-style-type: none"> 7.1 Luonne 7.2 Kohde <ul style="list-style-type: none"> 7.2.1 Tunniste <ul style="list-style-type: none"> 7.2.1.1 Luettelo 7.2.1.2 Merkintä 7.2.2 Sisältö 8 Muistiinpanot <ul style="list-style-type: none"> 8.1 Taho 8.2 Päiväys 8.3 Sisältö 9 Luokitus <ul style="list-style-type: none"> 9.1 Tarkoitus 9.2 Luokituspolku <ul style="list-style-type: none"> 9.2.1 Lähde 9.2.2 Luokitusmerkintä <ul style="list-style-type: none"> 9.2.2.1 Identifiointitunnus 9.2.2.2 Tietue 9.3 Kuvaus 9.4 Avainsana
--	--	---

Kuva 7. LOM-standardin metatiedon perusrakenne (IEEE LOM 2002b).

Mikään LOM:n elementti ei ole pakollinen ja niiden järjestyksen voi valita vapaasti. Tietokenttiä voi lisätä tarpeen mukaan, mutta ne eivät saisi kuitenkaan korvata alkuperäisiä kenttiä. (Nirhamo 2002). Tämä mahdollistaa omien metatietomallien eli sovellusprofiilien luomisen, sillä käytännön työskentelyä ajatellen LOM-tietokenttiä on turhan paljon, eikä standardin käyttämä terminologia välttämättä sovi kaikkiin sovelluskohteisiin. Sovellusprofiileissa voidaan käyttää haluttomia elementtejä ja kohteeseen sopivaa sanastoa eli elementtien nimiä. Eri sovellusprofiilien käytössä saattaa kuitenkin tulla vastaan ongelmia niiden yhteensopivuudessa siirrettäessä tietoja järjestelmästä toiseen. Joku järjestelmä ei välttämättä tunnista toisessa järjestelmässä luotuja tietokenttiä ja jättää ne lukematta, tai tulkitsee ne väärin. Tätä on pyritty estämään erilaisilla tietoteknisillä ratkaisuilla, kuten metatietorekisterit, jotka kykenevät tunnistamaan erilaiset metatietomallit ja sovellusprofiilit ja luomaan yhteyden niiden välille (Duval et al. 2002). Jos sovellusprofiilissa käytetään sellaisia uusia elementtejä joita ei esiinny muualla, niitä varten on luotava oma nimiavaruutensa (Heery & Patel 2000). Seuraavassa esimerkissä on yhdistetty samaan metatietoresurssiin IMS Metadata - ja Dublin Core -sovellusprofiileja, joiden elementit erotetaan toisistaan ims:- ja dc:-nimiavaruus-tuliitteillä.

```

<?xml version="1.0"?>
<record
  xmlns="http://example.org/learningapp/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://example.org/learningapp/ http://example.org/learningapp/schema.xsd"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:ims="http://www.imsglobal.org/xsd/imsmd_v1p2">

  <dc:title> Frog maths</dc:title>
  <dc:identifier>http://somewhere.com/frogmaths/</dc:identifier>
  <dc:description>Simple maths games for 5-7 year olds.</dc:description>
  <ims:typicallearningtime>
    <ims:datetime>0000-00-00T00:15</ims:datetime>
  </ims:typicallearningtime>
</record>

```

Yksi esimerkki sovellusprofiileista on IMS LRM (Learning Resource Meta-data), joka on hyvin lähellä IEEE LOM -standardia. Muita profiileja ovat muun muassa UKCMF (UK Common Metadata Framework) ja CanCore (Canadian Core Learning Object Metadata Application Profile). LOM:n suomalaisen sovellusprofiilin luonnoksessa on määritetty vain 14 pakollista kenttää: Oppisisällön nimi, Kieli, Kuvaus, Tila, Tekijä, Päiväys, Metatietokuvaus, Kuvauksen kieli, Tallennusmuoto, Koko, Sijainti, Oppimateriaalin laji, Tekijänoikeudet ja Avainsanat/Luokitus (Nirhamo 2004).

Metatiedon kytkentä resurssitiedostoihin voidaan toteuttaa kolmella tavalla: upotettuna (embedded), liitettynä (associated) ja tietokantaan tallennettuna metatietona (Duval et al. 2002). Upotettu metatieto on tallennettu itse tiedostoon; tämä on mahdollista joissakin tiedostoformaateissa kuten Word- ja PDF-dokumentit, html-sivut ja jotkut kuvatiedostoformaattit. Ongelmana tässä tallennustavassa on se, että käytettävissä olevat metatietokentät voivat olla hyvinkin erilaisia tiedostotyypistä riippuen. Liitetty metatieto puolestaan tarkoittaa sitä, että kuvaustiedot tallennetaan erilliseen tiedostoon, joka on tyypillisesti xml-rakenteinen. Hankaluutena tässä on se, että miten erilliset resurssi- ja metatietotiedostot saadaan pysymään yhteydessä toisiinsa. Kolmas vaihtoehto on tallentaa metatietokuvaus tietokantaan samalla kun sisältöresurssi viedään järjestelmään; tietokantasovelluksessa hakujen tekeminen metatietoihin, eli oppimateriaalien etsiminen, on helppo toteuttaa. Useimmiten tämäntyyppinen ratkaisu toteutuu LCMS- tai vastaavissa järjestelmissä, joita Suomen mittakaavassa ovat oppimateriaalipankit.

5.2 IMS Question and Test Interoperability – kysymykset ja testit

5.2.1 Yleistä

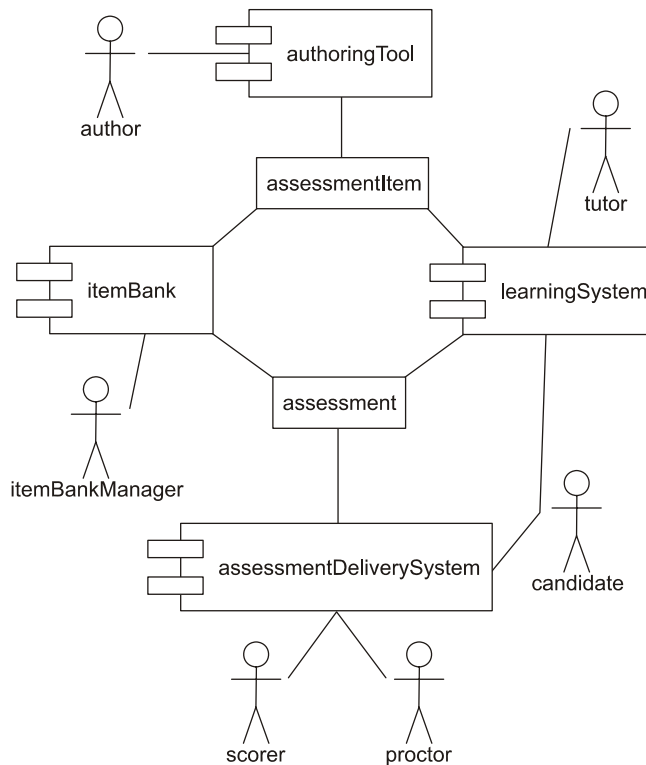
IMS Question and Test Interoperability -spesifikaatio (QTI) kuvaa tietomallin yksittäisten kysymysten ja niistä laadittujen testien esittämiseen ja tulosten raportointiin. QTI:n suunnittelussa on erityisesti pyritty siihen, että se tarjoaisi kysymysten tallentamiseen hyvin dokumentoidun sisällytöformaatin, joka on riippumaton kysymysten luomisessa käytettävistä työkaluista. Tavoitteena on tukea sellaista toimintatapaa, jossa erilaisista lähteistä peräisin olevat kysymykset tai kokonaiset kysymyspankit ovat hyödynnettävissä monenlaisten oppimisolustojen ja testienjakelujärjestelmien kanssa. Lisäksi QTI mahdollistaa testien tulosten raportoinnin yhdenmukaisella tavalla. (IMS QTI 2005)

Tällä hetkellä kysymystyökalut ja -järjestelmät tukevat yleisimmin IMS QTI -spesifikaation versioita 1.2 ja 1.2.1. Uusin versio spesifikaatiosta on kuitenkin 2.0, joka on julkaistu vuonna 2004. Version 2.0 pääasiallinen ero edelliseen verrattuna on, että item-kysymyskomponentin rakenne on suunniteltu uudelleen.

QTI-spesifikaatiosta on olemassa myös kevennetty QTI-Lite-versio, jota suositellaan ensimmäistä kertaa tähän aihepiiriin sovelluksia tekeville. QTI-Lite tukee vain monivalintatehtäviä, mutta tehdyt kysymykset ja testit ovat yhteensopivia varsinaisen QTI-spesifikaation kanssa.

5.2.2 Spesifikaation käyttöalue

Spesifikaation käyttöalue voidaan esittää kuvan 8 mukaisella kaaviolla, joka esittää kysymysten (assessmentItem), niistä koostettujen testien (assessment) sekä muiden komponenttien ja toimijoiden roolit kokonaisuudessa.



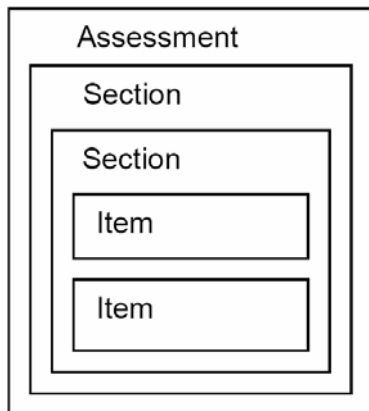
Kuva 8. IMS QTI 2.0:n mukaiset testeihin liittyvät komponentit ja toimijat (IMS QTI 2005).

AssessmentDeliverySystem on järjestelmä, joka hallinnoi testien esittämisen oppijoille eli käyttäjille (**candidates**) sisältäen ”esitysmoottorin” (delivery engine), joka muodostaa kysymykset ja jakaa ne käyttäjille, sekä pisteyttää vastaukset automaattisesti tai lähettää ne arvioijille (**scorer**).

AuthoringTool on työväline, jolla tekijä (**author**) luo ja muokkaa kysymyksiä (**assessment item**). Kysymys (item) sisältää kysymyksen (question), käyttäjälle esitettävät ohjeet, vastauksen käsittelyn (pisteytyksen) ja mahdollisen palautteen. **ItemBank** on kysymysten kokoamiseen ja hallitsemiseen liittyvä järjestelmä, kysymyspankki. Toimijaa, jonka vastuulla on kysymyspankin kysymyskokoelmien hallitseminen, kutsutaan **itemBankManageriksi**.

LearningSystem on oppimisalustajärjestelmä käyttäjien (oppijoiden) opiskelutoimintoja varten, jota mahdollisesti koordinoi ohjaaja (**tutor**). Testien jakelun valvoja puolestaan on **proctor**. Spesifikaatiossa valvoja voi olla kuka tahansa prosessissa mukana oleva henkilö, paitsi käyttäjä ja käyttäjän vastauksia arvioivat henkilöt. **Scorer** eli arvioija on henkilö tai ulkoinen järjestelmä, joka on vastuussa käyttäjän vastausten arvioinnista. Arvioija on valinnainen, koska monet kysymykset voidaan arvioida automaattisesti käyttämällä kysymyksiin liitettyjä vastausten käsittelysääntöjä.

Kysymyksiä (item) voidaan ryhmitellä kysymysryhmiksi (section) esimerkiksi aihealueittain. Näin ollen testi (assessment) voi sisältää ryhmiä ja/tai kysymyksiä, ja ryhmä voi sisältää kysymyksiä ja/tai muita ryhmiä. Tämä on ns. ASI-rakenne (Assessment-Section-Item), kuva 9. Yhteenniputettuja kysymyksiä ja/tai kysymysryhmiä kutsutaan kysymyspankiksi (object bank, item bank). (IMS QTI 2005). Kysymyspankit on suunniteltu sisällön siirtämiseksi järjestelmästä toiseen, ja niiden täytyy sisältää metatietoa hakujen mahdollistamiseksi. Kysymyspankit voivat myös toimia sellaisina tietokantoina, joista muodostetaan testejä erilaisiin tarpeisiin. (Fallon & Brown, 102)



Kuva 9. ASI-rakenne (IMS QTI 2005).

5.2.3 Kysymystyyppit

QTI-spesifikaatioissa kysymystyyppit jaetaan neljään ryhmään: looginen tunniste, xy-koordinaatti, merkkijono, numeerinen ja looginen ryhmä. Sama kysymystyyppi voi esiintyä pienin variaatioin eri ryhmissä, joten taulukossa 3 on lyhyet kuvaukset eri kysymystyypeistä ilman ryhmäjakoa. Spesifikaatio mahdollistaa kysymysten toteuttamisen monella eri tavalla, esimerkiksi niiden yhdistelmät. Taulukossa 3 on kuvattu tavallisimmat toteutusmuodot.

Taulukko 3. IMS QTI -kysymystyyppit (IMS QTI 2005).

Kysymystyyppi	Kuvaus
True/False	Oikein/väärin-kysymys, jossa valitaan oikea väittämätteksti.
Multiple Choice	Monivalintakysymys, jossa teksti-, kuva- tai äänityyppisistä vastausvaihtoehdoista valitaan yksi oikea asianmukaista valintapainiketta napsauttamalla.
Multiple Response	Monivalintakysymys, jossa tekstityyppisistä vastausvaihtoehdoista valitaan kaikki oikeat vaihtoehdot asianmukaisia valintaruutuja napsauttamalla.

Kysymystyyppi	Kuvaus
Image Hot Spot	Kuvaan on määritetty yksi tai useampia kohdepisteitä, joista käyttäjä valitsee oikean/oikeat vaihtoehdot hiirellä napsauttamalla.
Slider	Liukusäätimellä valitaan oikea annetuista vaihtoehdoista (monivalinta), tai säädintä liu'uttamalla etsitään oikea lukuarvo.
Order Objects	Kuvat tai tekstit on siirrettävä hiirellä oikeaan järjestykseen.
Connect-the-Points	Kuvassa on kohdepisteitä, joista on valittava oikeat esimerkiksi jonkin halutun kuvion muodostamiseksi.
Fill-in-Blank	Tekstissä olevaan aukkoon on kirjoitettava oikea vastaus, joka on kysymystyyppistä riippuen tekstiä, kokonaisluku tai desimaaliluku. Tekstityypissä kysymyksissä aukkoja voi olla yksi tai useampia, numeerisissa yksi. Kirjoittamisen sijaan vastaus on myös mahdollista valita alasvetoluettelosta.
Short Answer	Kysymys, jossa tekstikenttään kirjoitetaan vapaamuotoinen lyhyt vastaus.
Drag-and-Drop	Oikeaa vaihtoehtoa esittävä kuva vedetään sille varattuun kohtaan hiirellä.
Match items	Kysymystyyppi, jossa on yhdistettävä oikeat parit.

IMS QTI -spesifikaatiot sisältävät esimerkkejä eri kysymystyypeistä. Niiden lisäksi CETIS on tehnyt spesifikaation 1.2.1-versioon pohjautuvat toimivat esimerkit (CETIS 2004).

5.2.4 Kysymysten ja testien tallentaminen

Yksinkertaisimmillaan QTI-spesifikaation mukaiset kysymykset tallennetaan XML-tiedostoksi eteenpäin siirrettäviksi. Jos kysymykseen liittyy kuvia tai muita mediatiedostoja, ne ovat luonnollisesti erillisinä tiedostoina. Käytännössä XML-muotoisen kysymystiedoston ja mediatiedostojen lisäksi käytetään myös julistustiedostoa (imsmanifest.xml). Julistustiedostossa kuvataan kysymykseen liittyvä metatieto sekä kysymystiedostot ja kysymykseen liittyvät mediatiedostot. Kaikki tiedostot paketoidaan IMS Content Packaging -metodilla yhdeksi zip-tiedostoksi.

Seuraavassa on esimerkkinä Respondus-ohjelmasta tallennetun testin julistustiedosto (imsmanifest.xml), joka on rakenteeltaan IMS CP:n mukainen.

```
<?xml version="1.0"?>
<manifest identifier="MANIFEST1" xmlns="http://www.imsglobal.org/xsd/imscp_v1p1"
  xmlns:imsmd="http://www.imsglobal.org/xsd/imsmd_v1p2"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.imsglobal.org/xsd/imscp_v1p1 imscp_v1p1.xsd
  http://www.imsglobal.org/xsd/imsmd_v1p2 imsmd_v1p2p2.xsd">
```

```

<metadata>
  <schema>IMS Content</schema>
  <schemaversion>1.1.3</schemaversion>
  <imsmd:lom>
    <imsmd:general>
      <imsmd:identifier>
        <imsmd:langstring xml:lang="en-
          US">B0AECF4105C84C37B1CE11CFD86A3083</imsmd:langstring>
      </imsmd:identifier>
      <imsmd:title>
        <imsmd:langstring xml:lang="en-US">QTITesti (Respondus IMS QTI export)
        </imsmd:langstring>
      </imsmd:title>
    </imsmd:general>
  </imsmd:lom>
</metadata>
<organizations default="EXAM1">
  <organization identifier="EXAM1" structure="hierarchical">
    <title>default</title>
    <item identifier="ITEM1" identifierref="RESOURCE1">
      <title>Exam 1</title>
    </item>
  </organization>
</organizations>
<resources>
  <resource identifier="RESOURCE1" type="imsqti_xmlv1p1" href="QTITesti.xml">
    <file href="QTITesti.xml" />
    <file href="kuva1.jpg" />
    <file href="kuva2.jpg" />
  </resource>
</resources>
</manifest>

```

<resource>-elementeillä kuvataan kaikki testin kysymykset. Tässä esimerkissä on vain yksi <resource>-elementti, joka viittaa varsinaiseen kysymystiedostoon QTITesti.xml; kysymystiedosto voi sisältää yhden tai useampia kysymyksiä. Type-attribuutti imsqti_xmlv1p1 ilmaisee, että kysymyksessä on IMS QTI 1.1 -version mukainen kysymystiedosto. <file>-elementeillä kerrotaan kaikki kyseiseen <resource>-elementtiin sisältyvät mediatiedostot. Kysymykset olisi mahdollista

hajauttaa eri xml-tiedostoihin, jolloin jokaista kysymystä varten olisi oltava oma <resource>-elementtinsä.

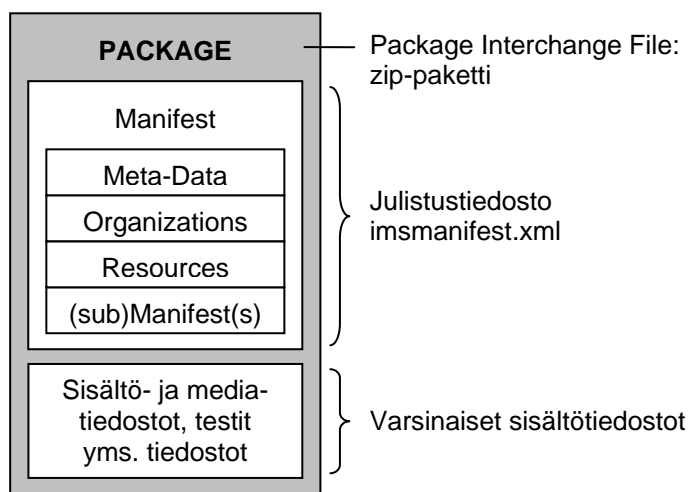
<manifest>-elementtiin liitetty <metadata>-elementti, kuten tässä esimerkissä, kuvaa kyseisen paketin, ei sen sisältöä. Jos halutaan kuvailla kunkin kysymyksen metatiedot, <metadata>-elementin on oltava <resource>-elementin sisällä. Tällöin kukin <resource>-elementti saa viitata vain yhteen kysymykseen (yksi kysymys kussakin XML-kysymystiedostossa). (IMS QTI 2005)

<organization>-elementti on IMS Content Packaging -spesifikaatiosta peräisin, mutta sitä ei käytetä QTI-spesifikaatioissa kysymysten esitysjärjestyksen määrittämisessä. Poikkeuksena on tilanne, jossa IMS QTI 2.0 ja IMS Simple Sequencing on integroitu toisiinsa, jolloin esitysjärjestys voidaan määrittää. (IMS QTI 2005)

5.3 IMS Content Packaging – sisällön paketointi

5.3.1 Sisältöpaketin komponentit

IMS Content Packaging -spesifikaatio (IMS CP) kuvaa määrämuotoisen tietorakenteen, jonka mukaisesti sisällöt paketoidaan. Spesifikaation tavoitteena on mahdollistaa sisällön yhteensopi- vuus eri välineiden ja järjestelmien, kuten sisällönluomistyökalut ja oppimisalustat, välillä. Ku- vassa 10 on esitetty IMS CP -mallin eri komponentit.



Kuva 10. IMS Content Packaging -tietomallin komponentit (mukaellen IMS CP 2005).

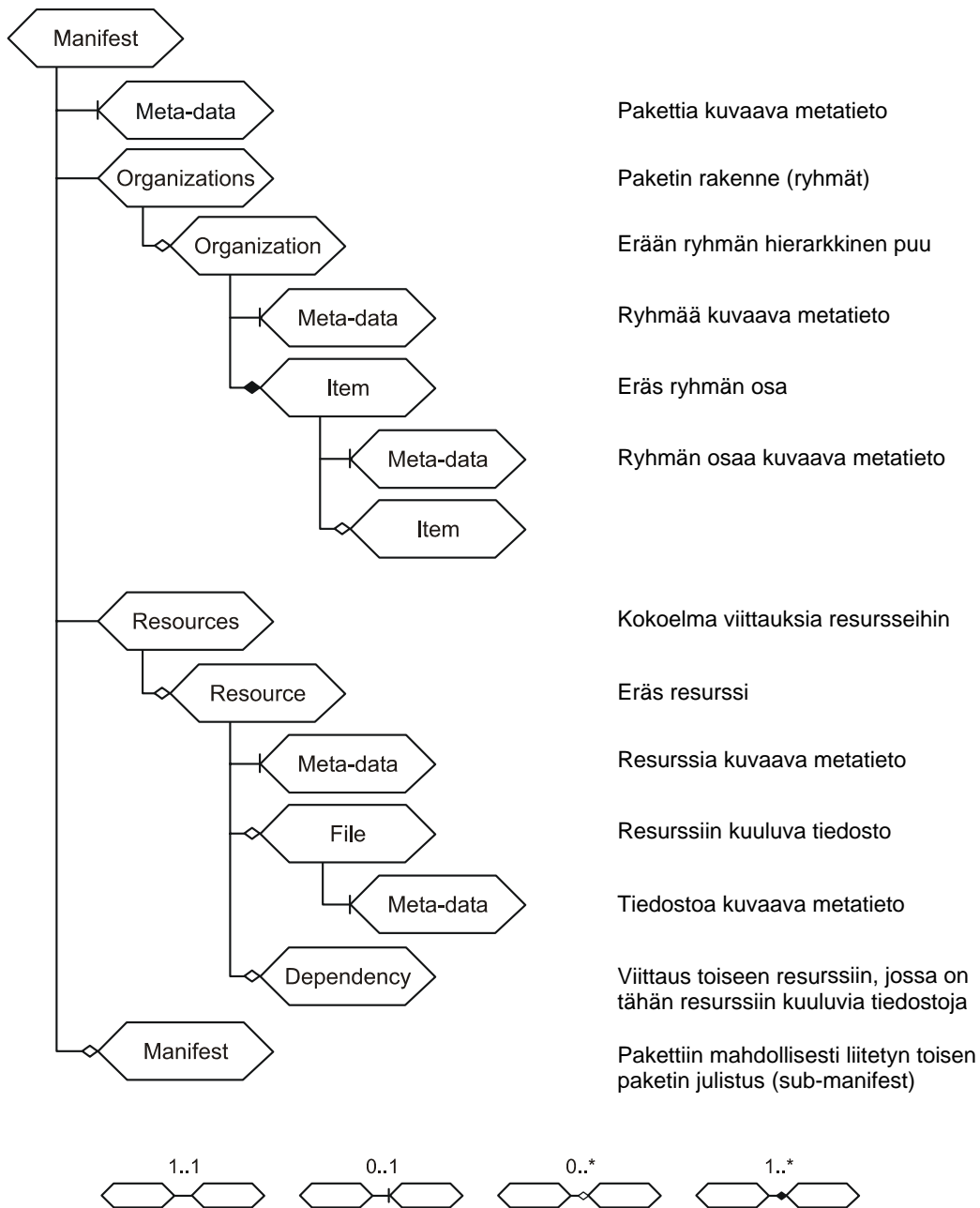
Sisältöpaketti (package) koostuu kahdesta pääelementistä: imsmanifest.xml-julistustiedostosta ja varsinaisista sisältötiedostoista. Julistustiedostossa kuvataan pakettiin kuuluvat sisältötiedostot

(resources), niiden järjestys (organizations) ja metatiedot. Sisältötiedostot voivat olla mitä tahansa tiedostoja, esimerkiksi html-tiedostoja ja niihin liittyviä kuvatiedostoja, video- ja äänitiedostoja jne. Julistus- ja sisältötiedostot pakataan yleensä yhdeksi zip-paketiksi, joka on siirrettävissä työkaluohjelmasta tai järjestelmästä toiseen. Sisältötiedostot on mahdollista sijoittaa alikansioihin, mutta julistustiedoston on aina sijaittava paketin juurikansiossa. On huomattava, että spesifikaatio ei edellytä tiedostojen paketoimista yhdeksi tiedostoksi; paketti (package) voidaan jakaa myös erillisinä tiedostoina.

Avattaessa paketti kohdesovellukseen, se lukee paketista julistusosan, ja muodostaa sen tietojen perusteella käyttäjälle näytettävän rakenteen. IMS CP ei ota varsinaisesti kantaa paketin eli oppimisaihion sisällön laajuuteen; tekijä voi valita haluamansa koostetason. Vaatimuksena on kuitenkin, että paketti on itsenäinen kokonaisuus, kun se tarjotaan oppijalle. Yksi pakettien ominaisuus on, että niitä voidaan yhdistää (aggregation) uusiksi paketeiksi, ja näin koostettuja paketteja voidaan purkaa (disaggregation) takaisin alkuperäisiksi paketeiksi.

5.3.2 Julistusosan elementit

Kuva 11 esittää kaaviona julistusosan elementit. Julistus (manifest) voi sisältää yhden, useampia tai ei yhtään alijulistusta. Tämä mahdollistaa useampien pakettien yhdistämisen yhdeksi laajemmaksi koostepaketiksi.



Kuva 11. Julistusosan elementit (IMS CP 2005).

Kuvan 11 mukaisesti metatieto on mahdollista liittää moniin paketin elementteihin. On kuitenkin huomattava, että metatiedon määrittäminen ei ole pakollista. Metatiedot pohjautuvat IMS Meta-Data -spesifikaation mukaiseen sanastoon. (IMS CP 2005)

Kussakin julistuksessa voi olla vain yksi <organizations>-elementti, joka on pakollinen, ja se sisältää yhden, useampia tai ei yhtään <organization>-alielementtiä. Kukin <organization>-elementti voi sisältää yhden tai useampia <item>-elementtejä, jotka voivat sisältää myös <item>-

alielementtejä. (IMS CP 2005). <item>-elementeillä saadaan sisältö esitettyä käyttäjälle kirjan sisällysluettelon tapaan, kuva 12.

Sisällysluettelo

- ▼ 1. IMS Content Packaging
 - 1.1. Sisältöpakettin komponentit
 - ▼ 1.2. Julistusosan elementit
 - 1.2.1. metadata
 - 1.2.2. organizations
 - 1.2.3. resources
- ▶ 2. IMS QTI



Kuva 12. Saman sisältöpakettin kaksi erilaista sisällysluettelon esitystapaa.

<organization>-elementtien avulla sisältöä voidaan niin haluttaessa ryhmitellä erilaisiksi näkymiksi. Spesifikaation ajatuksena on tarjota mahdollisuus esimerkiksi sellaiseen esitystapaan, jossa käyttäjä voi valita joko pikakatsauksen sisältöön tai yksityiskohtaisen sisällön esittämisen. Sisältöryhmien avulla sisältö voidaan myöskin esittää erilaisena eri käyttäjäryhmille, tai kohdistaa erilaisille oppimistyylyille. (Ostyn 2000). Käyttäjä valitsee haluamansa sisältöryhmän esimerkiksi alasvetovalikosta. Jos käytetään useampaa kuin yhtä <organization>-elementtiä, jokin niistä on määritettävä oletuselementiksi; kyseinen sisältöryhmä siis näkyy alasvetovalikossa oletusvalintana.

Pakollisen <resources>-elementin <resource>-alielementit sisältävät viittaukset kaikkiin niihin tiedostoihin, jotka tarvitaan <organizations>-elementissä määritetyn rakenteen toteuttamiseksi. Yhteenkytkentä <organization>-elementin <item>-alielementtien ja niitä vastaavien <resource>-elementin välillä saadaan aikaiseksi siten, että kunkin <item>-alielementin identifierref-attribuutilla ja sitä vastaavan <resource>-elementin identifier-attribuutilla on sama arvo, vrt. alla oleva koodiesimerkki.

```

<organization identifier="ORG-09898583" structure="hierarchical">
  <title>Sisältö</title>
  <item identifier="ITEM-5DA994B8" identifierref="RES-BD753966" isvisible="true">
    <title>IMS Content Packaging</title>
  </item>
  ...
  
```

```

<resources>
  <resource identifier="RES-BD753966" type="webcontent" href="IMS_CP.htm">
    <file href="IMS_CP.htm" />
    <file href="_themes/sandston/astonrul.gif" />
  </resource>
  ...

```

On kuitenkin huomattava, että kaikki <item>-elementit eivät välttämättä sisällä viittausta johonkin resurssitiedostoon. Näin ollen osa sisällysluettelon kohdista voi olla pelkkiä sisältöä jäsentäviä tekstejä, kuten esimerkiksi kohta *Julistusosan elementit* kuvassa 12.

Viittaukset resurssitiedostoihin <resource>-elementeissä voivat olla paketin sisäisiä suhteellisia viittauksia, tai absoluuttisia viittauksia paketin ulkopuoliseen osoitteeseen. Resurssin ei siis tarvitse – eikä voikaan olla – olla osa pakettia sellaisissa tapauksissa, kun viittaus on esimerkiksi ulkopuoliseen URL-osoitteeseen. Paketin sisäiset viittaukset ilmoitetaan <file>-elementillä. Julistuksessa voi olla vain yksi <resources>-elementti, ja se voi sisältää yhden, useampia tai ei yhtään <resource>-elementtiä. (IMS CP 2005)

<resource>-elementti voi sisältää <dependency>-alielementin, joka viittaa johonkin toiseen <resource>-elementtiin, joka sisältää yksittäisen resurssitiedoston. Tätä voidaan hyödyntää silloin, kun useammassa <resource>-elementissä käytetään samoja resurssitiedostoja. Sen sijaan, että samat resurssitiedostot lueteltaisiin jokaisessa <resource>-elementissä, viitataan niihin <resource>-elementteihin, joihin resurssitiedostot on sijoitettu. (IMS CP 2005)

5.4 SCORM-referenssimalli

SCORM-referenssimalli (Sharable Content Object Reference Model) on IMS CP:n lailla tapa kuvata sisältöpaketteja. SCORM 1.2 muodostuu kahdesta osasta: The SCORM Content Aggregation Model (CAM) ja The SCORM Run-Time Environment (RTE). Näistä ensin mainittu kuvaa sen rakenteen, jolla sisältömateriaali koostetaan esimerkiksi oppimisalustalle siirrettäväksi sekä miten metatieto liitetään sisältöön. RTE puolestaan kuvaa tavan, jolla koostettu sisältö ja oppimisalusta kommunikoivat keskenään.

SCORM-referenssimallin yleisimmin käytössä oleva versio on tällä hetkellä 1.2. Siitä on julkaistu uudempi versio 1.3, joka tunnetaan paremmin nimellä SCORM 2004. Seuraavissa kohdissa kuvataan version 1.2 mukainen sisällönkoostamismalli (CAM) ja käytönaikainen ympäristö (RTE) sekä lopuksi versiossa 2004 uutena määrittymisenä mukaan tulleet SCORM Sequencing and Navigation, jolla kuvataan sisällön esittämisjärjestys- ja navigointisäännöt.

5.4.1 Sisällönkoostamismalli

SCORM CAM -sisällönkoostamismalli on kokoelma erilaisia spesifikaatioita, joita muut tahot ovat tuottaneet. Sisällön rakenne pohjautuu AICC:n spesifikaatioihin sekä IMS CP -spesifikaatioon. Metatietomalli puolestaan perustuu IMS:n ja IEEE:n metatietomäärittymisille. (SCORM CAM 2001)

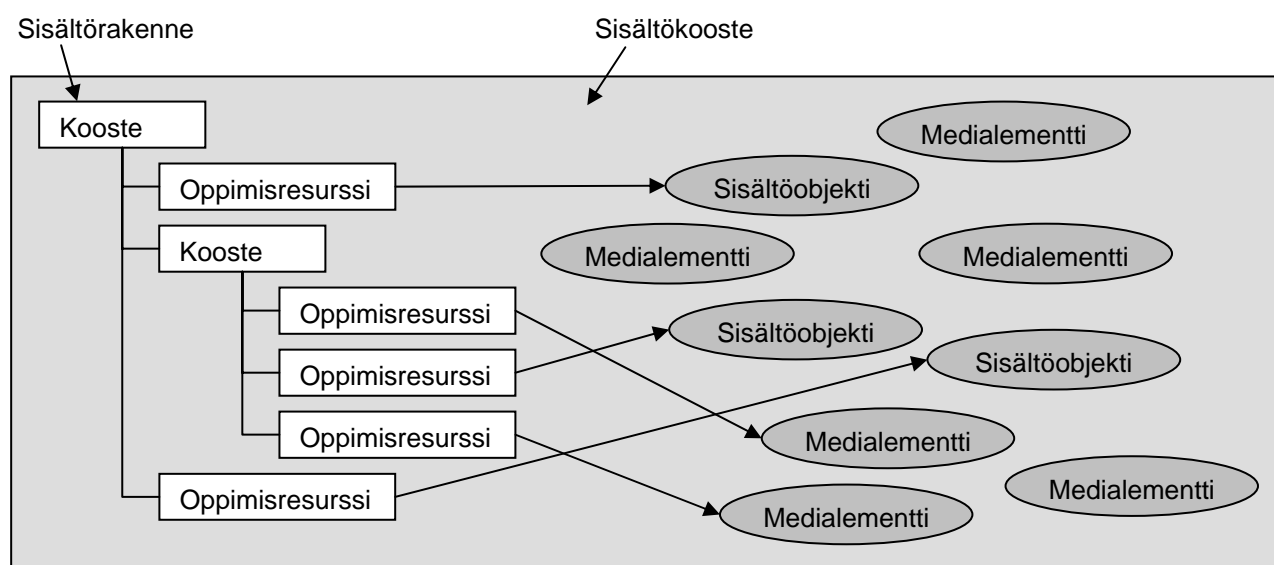
Sisällönkoostamismalli muodostuu sisältömallista, metatiedosta ja sisällön paketoinnista, joita käsitellään seuraavissa kohdissa tarkemmin.

Sisältömalli

Sisältömallilla kuvataan oppimisaihioiden eri koostetasoja. Sisällön perusosanen on *medialementti* (asset), joka voi olla esimerkiksi teksti, kuva, flash-animaatio, www-sivu tms. Seuraavana on SCORM-mallin keskeisin oppimisaihi Sharable Content Object (SCO), *sisältöobjekti*. SCO muodostuu yhdestä tai useammasta mediaelementistä, ja on pienimmän koostetason objekti, joka pystyy kommunikoimaan oppimisalustan kanssa ajonaikaisen ympäristön (RTE) välityksellä. Tämän toteuttamiseksi sisältöobjektit sisältävät JavaScript-koodia, jonka avulla SCO pystyy paikantamaan oppimisalustan API-sovittimen (API Adapter). (SCORM CAM 2001). Tästä kerrotaan tarkemmin kohdassa 5.4.2 *Käytönaikainen ympäristö*.

Sisällönkoostamismallin käsitteistössä laajin oppimisaihi on *sisältökooste* (Content Aggregation). Tällä koostetasolla oppimisaihi voi olla kokonainen kurssi, luku, moduuli tms. Käytännössä sisältökooste määrittää oppimateriaalin rakenteen ja esitysjärjestyksen käyttäjälle. Sisältökoosteen oppimisaihioiden esitysjärjestys määritetään SCORM 1.2 -versiossa aihioille asetettavien ennakkoehtojen (prerequisites) avulla. Tämä tarkoittaa sitä, että kullekin oppimisaihiolle voidaan määrittää, mikä tai mitkä muut aihiot on käytävä läpi ennen kyseiseen aihioon siirtymistä. (SCORM CAM 2001). Kohdassa 5.4.3 *Oppimisaihioiden esittämisjärjestys ja niiden välinen navigointi* kuvataan, miten esitysjärjestys määritetään SCORM 2004 -spesifikaation mukaisissa sisällöissä.

Kuva 13 havainnollistaa sisältökoosteen rakennetta. Merkillepantavaa on, että käyttäjälle tarjottava oppimisresurssi voi olla sekä mediaelementti että sisältöobjekti. ”Irralliset” mediaelementit ovat sellaisia resurssitiedostoja, joita kutsutaan sisäisesti jostakin toisesta mediaelementistä, kuten esimerkiksi html-sivuun linkitetty videotiedosto. On kuitenkin huomattava, että uudelleenkäytävyyttä ajatellen oppimisresurssien välille ei muutoin saisi tehdä linkityksiä; sisällön osasta toiseen siirtyminen tapahtuu käyttäjälle esitettävän sisältörakenteen kautta. Sisältörakennetta kuvassa esittävät kaikki valkoiset nelikulmiot.



Kuva 13. Sisältökoosteen rakenne (lähde SCORM CAM 2001 mukaellen).

Metatieto

Spesifikaation mukaan metatietokuvaus voidaan liittää medialementteihin, sisältöobjekteihin ja sisältökoosteisiin. Näiden lisäksi myös SCORM-pakettia on mahdollista kuvata metatiedoilla. Paras paikka metatiedolle onkin pakettitasolla, koska käytännössä juuri paketteja siirretään käyttäjien, oppimislustojen ja oppimateriaalipankkien välillä eikä yksittäisiä pakettien osia. (Ostyn 2000)

Metatieto voi sijaita joko sisältöpaketin julistustiedostossa (imsmanifest.xml) tai IMS CP -mallista poiketen erillisessä tiedostossa, johon on viittaus julistustiedostosta. Viittaus ulkopuoliseen tiedostoon tehdään <adlcp:location>-elementin avulla seuraavan esimerkin tapaan.

```
<metadata>
  <schema>ADL SCORM</schema>
  <schemaversion>1.2</schemaversion>
  <adlcp:location>course/metadata1.xml</adlcp:location>
</metadata>
```

Sisällön paketointi

Sisällön paketoinnissa käytetään kohdassa 5.3 *IMS Content Packaging – sisällön paketointi* kuvattua tapaa SCORM-erityispiirtein lisättyinä. Paketoimiseen käytetään kahta vaihtoehtoista sovellusprofiilia: *resurssipaketit* (resource packages) ja *sisältökoostepaketit* (content aggregation packages). Resurssipaketit sisältävät mediaelementtejä ja/tai sisältöobjekteja, mutta paketissa ei ole sisältörakennetta, koska paketit on tarkoitettu pelkkään sisällön siirtämiseen paikasta toiseen. Kun paketoidaan sisältökoosteita, kyse on sisältökoostepaketeista. Tällöin käyttäjälle näytettävä sisältörakenne esitetään pakettia kuvaavassa imsmanifest.xml-tiedostossa <organization>- ja <item>-elementeillä IMS CP:n tapaan. (SCORM CAM 2001). Sisältökoostepaketit ovat IMS CP:n mukaisia paketteja, mutta resurssipaketit eivät.

IMS CP -tietomalliin verrattuna SCORM sisältää viisi valinnaista (ei pakollista) laajennusta sisältöä kuvaaville <item>-elementeille: <adlcp:prerequisites>, <adlcp:maxtimeallowed>, <adlcp:timelimitaction>, <adlcp:datafromlms> ja <adlcp:masteryscore>. Laajennusten tehtävät ovat seuraavat (SCORM CAM 2001):

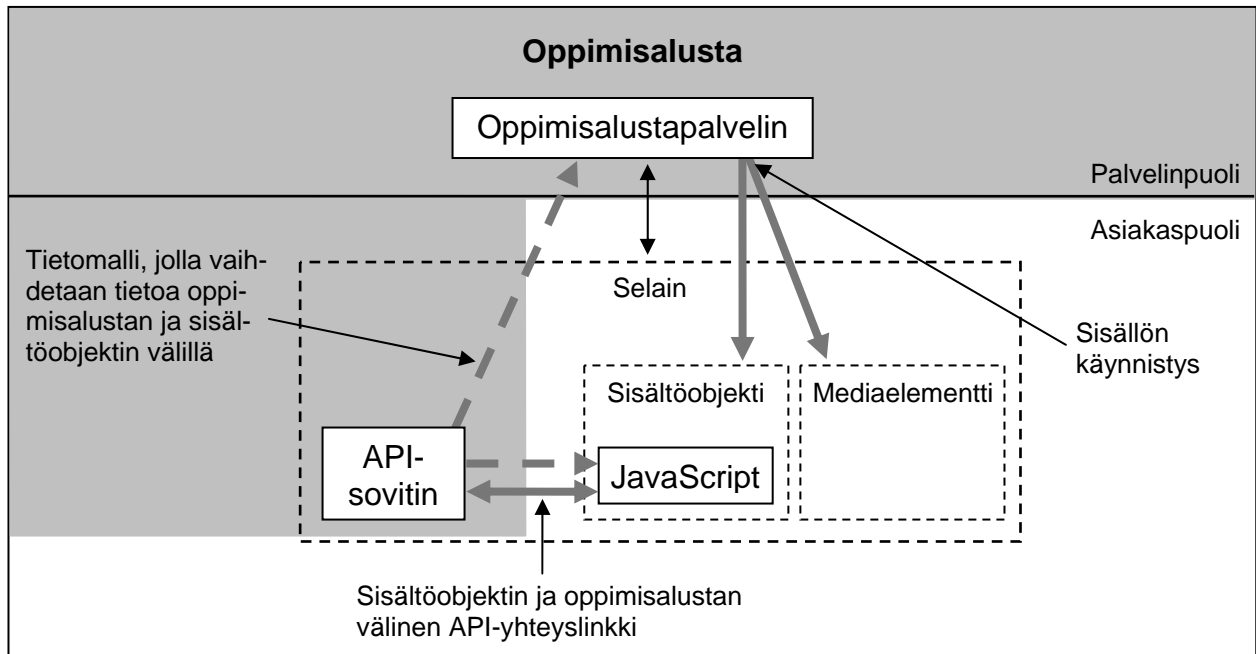
<adlcp:prerequisites>	Tässä elementissä määritetään, mitkä muut sisällön osat on käytävä läpi ennen kuin kyseinen <item>-osa voidaan aloittaa. Tämä mahdollistaa erilaisten oppimispolkujen määrittämisen sisältöön.
<adlcp:maxtimeallowed>	Tällä elementillä määritetään, kuinka pitkään kyseisen <item>-sisältökohdan tarkastelu on sallittu kyseisellä yrityserralla.
<adlcp:timelimitaction>	Tämä elementti määrittää, mitä tehdään sen jälkeen, kun määritetty aika on kulunut umpeen. Mahdollisia vaihtoehtoja ovat 'exit,message', 'exit,no message', 'continue,message' ja 'continue,no message'.
<adlcp:datafromlms>	Elementti tarjoaa <item>-sisältöobjektin odottaman alustustiedon.

<adlcp:masteryscore> Tässä elementissä määritetään se pistemäärä, joka on saavutettava, jotta kyseinen <item>-osa merkittäisiin suoritetuksi. Pistemäärä voi olla väliltä 0-100.

5.4.2 Käytönaikainen ympäristö

Käytönaikaisella ympäristöllä (run-time environment) tarkoitetaan sitä toteutusta, joka mahdollistaa oppimisalustan ja sisällön välisen kommunikoinnin. Oppimisalusta voi käynnistää (launch) kahdentyyppisiä sisältökomponentteja: medialementtejä ja SCO-sisältöobjekteja. Näistä vain sisältöobjektit voivat vaihtaa tietoa oppimisalustan kanssa, joten jatkossa esitellään vain tämän tyyppisten objektien toimintaa. (SCORM RTE 2001). Oppimisalustan ja sisältöobjektin välinen tiedonvaihto voi sisältää tiedon käyttäjästä, SCO:n käyttämiseen kuluneen ajan, testien ja kysymysten tulokset, tiedon SCO:n suorittamisen tilasta (aloittamatta, kesken, hylätty, hyväksytty) sekä kirjanmerkit, joiden perusteella oppija voi myöhemmin palata suorittamaan kesken jääneet sisältöobjektit (Reusable Learning 2004b).

Käytönaikaisen ympäristön periaatekaavio on esitetty kuvassa 14. Oppimisalusta tarjoaa API-sovittimen (API Adapter), jonka kautta sisältöobjektit kommunikoivat oppimisalustan kanssa. API-sovitin piilottaa sisältöobjekteilta kommunikointirakenteen monimutkaisuuden, joka on mahdollista toteuttaa monella eri tavalla alustasta riippuen. Sen sijaan oppimisaihion ja API-sovittimen välinen kommunikointi on standardoitu: API-sovitin sisältää kaiken kaikkiaan kahdeksan erilaista JavaScript-funktiota, joita sisältöobjekti voi kutsua. Jokaisessa SCO-sisältöobjekteissa on kommunikointia varten JavaScript-koodia, jonka on sisällettävä vähintään API-kutsut LMSInitialize() ja LMSFinish(). (SCORM RTE 2001)



Kuva 14. SCORM:n käytönaikainen ympäristö (mukaellen SCORM RTE 2001).

Oppimisolustan ja sisältöobjektin välinen toimintatapa on pääpiirteissään seuraavan kaavan mukainen. (SCORM RTE 2001)

1. Käyttäjä valitsee sisältökoosteesta haluamansa sisältöobjektin.
2. Oppimisolusta käynnistää sisältöobjektin.
3. Sisältöobjekti etsii API-sovittimen ja kutsuu `LMSInitialize()`-funktiota, jolla SCO perustaa kommunikointiyhteyden oppimisolustan kanssa.
4. Käyttäjä käyttää sisältöobjektia. Riippuen objektin toteutuksesta, se voi lähettää tietoa oppimisolustalle tai pyytää sitä oppimisolustalta tiettyjen funktiokutsujen avulla.
5. Kun käyttäjä lopettaa sisältöobjektin käytön, se kutsuu API-sovittimen `LMSFinish()`-funktiota, jonka perusteella oppimisolusta voi päivittää kirjanpitonsa ja merkitä esimerkiksi sisällön suoritetuksi.

Tässä toimintamallissa on joitakin sääntöjä, joista ensimmäinen on, että sisältöobjektin saa käynnistää vain oppimisolusta, ei toinen objekti. Kerrallaan käytössä voi olla vain yksi sisältöobjekti. Lisäksi kaikki API-sovittimen ja sisältöobjektin välinen kommunikointi aloitetaan aina sisältöobjektin toimesta. (SCORM RTE 2001)

Aikaisemmin mainittujen `LMSInitialize()`- ja `LMSFinish()`-kutsujen lisäksi objekti voi sisältää funktiokutsut tiedonsiirtoa ja virheenkäsittelyä varten. Tiedonsiirtofunktioista tärkeimmät ovat `LMSGetValue()` ja `LMSSetValue()`, joilla sisältöobjekti voi pyytää tietoa oppimisolustalta ja

lähettää sitä oppimisalustalle. Jos sisältöobjekti on esimerkiksi arvioitava tehtävä, suorituspisteet voidaan lähettää tallennettavaksi oppimisalustalle. (SCORM RTE 2001)

5.4.3 Oppimisaihoiden esittämisjärjestys ja niiden välinen navigointi

Suurin ero SCORM 1.2 - ja 2004 -versioiden välillä liittyy sisältökoosteen oppimisaihoiden väliseen navigointiin, erityisesti navigointipolkujen määrittämiseen. SCORM 1.2:ssa ei ole varsinaista tapaa polkujen luomiseen, vaan sisällön osille määritetään prerequisites-ennakkoehdoilla, mitkä muut osat on ensin käytävä läpi. SCORM 2004:ssä on sen sijaan otettu käyttöön uudet esittämisjärjestys- ja navigointisäännöt, jotka on kuvattu SCORM Sequencing and Navigation (SN) -dokumentissa. SCORM SN pohjautuu IMS Simple Sequencing -spesifikaatioon.

SCORM SN:n perustana on hierarkkinen toimintopuu (Activity Tree), joka kuvaa oppimistapahtumien rakenteen. Oppimisalusta luo toimintopuun imsmanifest.xml-julistustiedoston tietojen pohjalta; <organization>-elementti on toimintopuun juuri ja kaikki sen <item>-elementit esittävät oppimistapahtumia (learning activities). Oppimistapahtumasta käytetään myös lyhyempää aktiviteetti-nimitystä (activity). Aktiviteetti voi olla viittaus suoraan johonkin oppimisresurssiin eli SCO-sisältöobjektiin tai mediaelementtiin, tai se voi olla kokoelma lehti- eli aliaktiviteetteja (leaf / sub-activities). Aliaktiviteettien kokoelmaa kutsutaan aktiviteettiryhmäksi (cluster) silloin, kun siinä on yksi vanhempi-aktiviteetti (parent activity) ja sen välittömät lapsiaktiviteetit, mutta ei niiden perillisiä. (SCORM SN 2004). Seuraava esimerkkikoodi selventää edellä mainittujen käsitteiden sijoittumista sisällön rakennetta kuvaavan <organizations>-elementin sisällä.


```

<organizations default="ED59">
  <organization identifier="ED59" structure="hierarchical">
    <title>Kurssi ABC</title>          Toimintopuun otsikko
    <item identifier="ITEM-5C7A" isvisible="true">
      <title>Oppitunti 1</title>      Aktiviteetin otsikko
      <item identifier="ITEM-F6CE" isvisible="true" identifierref="RES-A418">
        <title>Esitesti</title>      Aliaktivitetti, viittaus oppimisresurssiin
      </item>
      <item identifier="ITEM-DEFE" isvisible="true" identifierref="RES-74DB">
        <title>Sisältö</title>       Aliaktivitetti, viittaus oppimisresurssiin
      </item>
      <item identifier="ITEM-0C3C" isvisible="true" identifierref="RES-59FF">
        <title>Jälkitestit</title>   Aliaktivitetti, viittaus oppimisresurssiin
      </item>
    </item>
    <item identifier="ITEM-010E" isvisible="true" identifierref="RES-B320">
      <title>Oppitunti 2</title>     Aktiviteetin otsikko, viittaus oppimisresurssiin
    </item>
  </organization>
</organizations>

```

} Aktiviteetti 1
 (aktiviteetti-ryhmä)
 } Aktiviteetti 2

Spesifikaation mukaan toimintopuun ei välttämättä tarvitse olla staattinen rakenne; on sallittua tehdä toteutus, jossa oppimisalusta voi dynaamisesti muuttaa puun rakennetta ja aktiviteetteihin liitettyä esitysjärjestysinformaatiota. Säännöt esitysjärjestyksen toteuttamiseksi määritetään sisältöpaketin julistustiedoston <organizations>- tai <item>-elementteihin lisätyillä <sequencing>- tai <sequencingCollection>-elementeillä.

Jokaiseen aktiviteettiryhmään voidaan määrittää yhdestä neljään hallintamoodia <controlMode>-elementillä. Hallintamoodien tarkoituksena on säätää oppijan pääsyä aktiviteettiryhmän aliaktiviteetteihin. Sen lisäksi jokaiselle aktiviteetille voidaan määrittää rajoittamaton määrä esitysjärjestyssääntöjä <sequencingRules>-elementillä. Niiden avulla säädetään erilaisia toimintatapoja aktiviteettiin saavuttaessa ja siitä poistuttaessa. (Mackenzie 2004)

Esitysjärjestyksen lisäksi SCORM SN mahdollistaa SCO-sisältöobjektien välisen navigoinnin toteuttamisen halutulla tavalla. Tyypillisestihän oppimisalusta tarjoaa sisällön navigointitoiminnot, mutta joissakin tapauksissa sisällön kehittäjä saattaa haluta tarjota sen sijaan oman navigointitoteutuksensa. SCO-objektien sisäinen navigointi on ollut mahdollista aikaisemminkin, mutta

SCORM-navigointimalli mahdollistaa myös niiden välisen navigoinnin. (SCORM SN 2004).
Sisältoobjekteissa voidaan näyttää esimerkiksi Jatka-painike, jolla siirrytään seuraavaan objektiin. Sisältoobjekti kommunikoi käytönaikaisen ympäristön kanssa, joka päättää esitysjärjestysääntöjen perusteella, onko toiseen objektiin siirtyminen sallittua. (Ostyn 2004)

6 SISÄLLÖN TUOTTAMINEN OPPIMISALUSTOILLE

Tässä luvussa esitellään WebCT- ja Moodle-oppimisalustat sisällöntuotannon näkökulmasta sekä joitakin työkaluja, jotka tukevat luvussa 5 kuvattuja yleisiä sisällöntuotantostandardeja ja -spesifikaatioita. Tarkastelun kohteena ovat sekä kysymysten ja testien tekemiseen että sisältömateriaalin paketointiin liittyvät sovellukset. Tarkoituksena on selvittää sitä, miten ne soveltuvat järjestelmästä toiseen siirrettävien aineistojen tekemiseen, ja tuoda esille mahdollisia ongelmakohtia.

6.1 WebCT

WebCT-kurssin käyttöliittymä koostuu tyypillisesti kuvakelinkeistä, joista siirrytään kurssin eri toimintoihin, kuten keskustelut, tehtävät ja testit. Kuvake voi myös olla linkki www-sivuille, tai se voi avata jonkin tiedoston tai tuoda näkyville sisältömateriaalin sisällysluettelon.

Tiedostoalueella olevaa tekstieditoria lukuun ottamatta WebCT:ssä ei ole työkaluja, joilla sisältömateriaalia voisi oppimisalustalla luoda. Käytännössä sisältö luodaankin muilla ohjelmilla ja tuodaan sitten WebCT:lle. Varsinainen oppimateriaali koostetaan tavallisesti WebCT:n Sisältökokonaisuus-työkalulla, jolla luodaan haluttu sisällysluettelo materiaaliin. Sisällysluettelosta tehdään sitten linkit WebCT:n tiedostoalueelle tuotuihin tiedostoihin, jotka voivat olla tyypiltään mitä tahansa html-tiedostoista ja Word-dokumenteista videotiedostoihin.

WebCT 4.0:aan ja joihinkin aikaisempiin alustaversioihin on alustan valmistajan www-sivuilta noudettavissa ja asennettavissa IMS Content Migration Utility (CMU), jonka avulla on mahdollista tuoda kurssille ja vastaavasti tallentaa kurssilta tiedostoksi IMS CP -spesifikaation mukaisia sisältöpaketteja. CMU sisältyy WebCT 4.1:een ja uudempiin alustaversioihin. Tuodut sisältöpaketit näkyvät WebCT:llä Sisältökokonaisuus-työkalun kautta. Jos kurssin tiedostoalueelle tuoduista tiedostoista on koostettu sisältöä Sisältökokonaisuus-työkalun avulla, myös tämä sisältö on tallennettavissa IMS CP -sisältöpakettiksi muille WebCT-kursseille tai muille oppimisalustoille siirrettäväksi.

Testeissä ja itsearviointitehtävissä tarvittavia kysymyksiä varten WebCT:n kysymystietokannassa on varsin toimivat kysymystenluontitoiminnot. Ongelmana on kuitenkin se, että kysymystietokanta on kurssikohtainen. Eli jos käyttäjällä on useita kursseja, luotuja kysymyksiä ei saa näkyään yhteisesti kaikilla kursseilla.

6.2 Moodle

Moodle poikkeaa jonkin verran WebCT:stä käyttötavaltaan. Kun WebCT on jossain määrin työkalukeskeinen, Moodlessa paino on enemmän oppimistapahtumilla; kurssin etusivulle lisätään linkkejä työkaluihin ja tiedostoihin aina tarpeen mukaan kurssin edetessä.

Sisältöjen luomiseen Moodlessa on monia työkaluja, joista helppokäyttöisellä web-sivueditorilla kurssille pystyy tekemään tekstisivuja kuvineen ja linkkeineen. Tehtyjä sisältöjä ei kuitenkaan ole mahdollista tallentaa tiedostoksi esimerkiksi muilla kurseilla käytettäväksi. Näin ollen käytökelpoisempi tapa on tehdä sisällöt muilla ohjelmilla ja tuoda sitten Moodleen. Omalta koneelta tuotujen tiedostojen linkittäminen Moodlessa on jonkin verran helpompaa kuin WebCT:ssä.

Moodle sisältää SCORM-moduulin, joka mahdollistaa SCORM- ja IMS CP -spesifikaatioiden mukaisten sisältöpakettien tuomisen kurssille. Moduulin luvataan olevan täysin SCORM 1.2 -yhteensopiva (conformant), ja myös AICC-muotoisten pakettien tuomisen pitäisi olla mahdollista.

Testikysymysten luomista varten Moodlessa on hyvät työkalut WebCT:n tapaan. Sen lisäksi kysymyskategoriat on mahdollista määrittää näkymään globaalisti, jolloin samat kysymykset ovat käytettävissä muillakin kurseilla. Seurauksena tosin on, että kaikki muutkin käyttäjät näkevät kysymykset. Globaalisti julkaistujen kysymyskategorioiden käyttö on kuitenkin toteutettu huonosti, sillä omien kategorioiden löytäminen kaikkien muiden joukosta on hankalaa, lisäksi kysymysten poistamista muiden luomista kategorioista ei ole mitenkään estetty.

6.3 Kysymykset ja testit

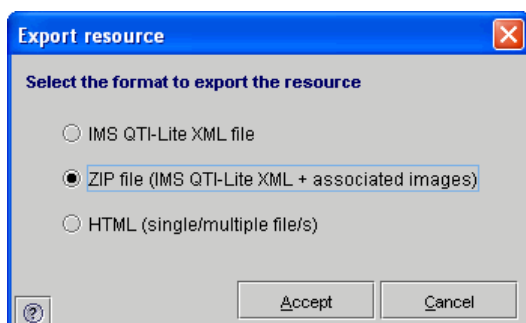
Markkinoilla on joitakin kysymysten ja testien tekemiseen ja hallintaan liittyviä kaupallisia sovelluksia, jotka tukevat IMS QTI -spesifikaatiota. Tällaisia tuotteita ovat esimerkiksi Questionmark Perception ja Respondus. Saatavilla on myös ainakin yksi ilmainen editori, QAed. Näiden lisäksi on meneillään erinäisiä hankkeita, joissa tehdään QTI-pohjaisia sovelluksia, kuten kysymyseditorit, testien ja kysymysten hallintajärjestelmät sekä QTI-yhteensopivat oppimisalustat. Avoimen lähdekoodin pohjalta ovat tekeillä esimerkiksi IMS QTI Editor ja Alfanet QTI Tools. Seuraavissa kohdissa esitellään joitakin yleisimpiä QTI-yhteensopivia editoreja. Niiden lisäksi esitellään yleisesti käytössä oleva Hot Potatoes -ohjelmisto, koska sillä luotuja kysymyksiä on mahdollista siirtää WebCT:lle ja Moodleen. Editoreilla luotuja kysymyksiä on työn puitteissa siirretty toisiin työvälineisiin, jotta saataisiin tuntumaa kysymysten siirrettävyydestä ja mahdolli-

sista ongelmista. Yhtenä testivälineenä on käytetty ilmaista SARAS QTI Vieweriä, jossa kysymyksiä voi katsella ja testata. SARAS sisältää myös QTI XML -validaattorin koodin oikeellisuuden tarkistamiseen.

Terminologiassa on jonkin verran vakiintumattomuutta ainakin suomenkielen osalta, sillä joidenkin ohjelmistojen kohdalla, esimerkkinä Hot Potatoes, kysymysten sijaan puhutaan tehtävistä.

6.3.1 QAed-editori

QAed (Questions & Assessments Editor) on ilmainen kysymyseditori, jonka toiminta pohjautuu IMS QTI-Lite -spesifikaatiolle. QAed mahdollistaa testien ja kysymysten tallentamisen (export) ja tuomisen (import) QTI-Lite-spesifikaation mukaisessa XML-muodossa, kuva 15. Tallennusvalinta *IMS QTI-Lite XML file* luo XML-tiedoston, jossa kysymykset on kuvattu, mutta ei kopioi niihin mahdollisesti liittyviä kuvia samaan kansioon. Kuvia sisältävissä tehtävissä onkin käytettävä toista tallennusvaihtoehtoa, joka luo zip-tiedoston kaikista tehtäviin liittyvistä resurssitiedoista. Zip-tiedosto ei sisällä *imsmanifest.xml*-julistustiedostoa, mutta se ei estä kysymysten avaamista muihin ohjelmiin.



Kuva 15. QAed:n Export-toiminnon tallennusvaihtoehdot.

QAed:n luoma XML-kysymystiedosto ei avaudu sellaisenaan esimerkiksi Respondus-kysymyseditoriin tai SARAS QTI Vieweriin ennen kuin XML-tiedostoa on muokattu poistamalla *imsqti-nimiavaruus*etuliitteet kaikista elementeistä. Muokkauksen jälkeen kysymykset avautuvat molempiin ohjelmiin. Seuraavassa on lyhyt listaus alkuperäistä koodia, johon on lihavoinnilla merkitty poistettavat etuliitteet.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<imsqti:questestinterop xmlns:imsqti="http://www.imsglobal.org/xsd/ims_qtilitev1p2"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.imsglobal.org/xsd/ims_qtilitev1p2_ims_qtilitev1p2.xsd">
  <imsqti:item ident="J200542505743" title="Presidentit">
    <imsqti:presentation>
      <imsqti:material>
        <imsqti:mattext xml:lang="FI">Kuka on Suomen presidentti?</imsqti:mattext>
      </imsqti:material>
    </imsqti:presentation>
  </imsqti:item>
</imsqti:questestinterop>
```

6.3.2 Questionmark Perception

Questionmark Perception on työkalu, jolla on mahdollista luoda kysymyksiä ja erityyppisiä testejä. Perception on alun perin tehty työkaluksi, jolla testit julkaistaan Perception-palvelimelle. Käyttäjät avaavat testit palvelimelta, jonne suoritusten arvioinnit lopuksi tallentuvat. Ohjelmisto koostuu Question Manager -kysymyseditorista ja testien hallintaan tarkoitettusta Assessment Managerista. Näistä vain Question Manager tukee IMS QTI -spesifikaatiota mahdollistaen kysymysten tallentamisen ja sisään tuomisen tässä muodossa.

Kysymyseditori on varsin helppokäyttöinen sisältäen 18 erilaista kysymystyyppiä. Osa kysymystyypeistä on itse asiassa samoja vain pienin muutoksin, kuten esimerkiksi *Fill in Blanks* ja *Numeric*, jotka molemmat ovat ns. aukkotehtäviä. Perception sisältää IMS QTI -spesifikaation mukaisista kysymystyypeistä yleisimmät.

Question Managerin QTI XML -tallennustoiminnolla (export) kysymykset on mahdollista tallentaa yksitellen, tai jos useampi kysymys on valittu ennen tallentamista, kaikki ne tallennetaan samaan XML-tiedostoon. Ohjelma ei tue sisällön paketoitua, eivätkä kysymyksiin liitetyt kuvat tallennu samaan kansioon mihin kysymystiedosto, joten kuvien yms. mediatiedostojen suhteen joudutaan tekemään melkoisesti käsityötä. Vastaavasti import-toiminnolla on mahdollista avata vain yksittäisiä XML-kysymystiedostoja, ei zip-kysymyspaketteja. Muilla ohjelmilla tehdyissä kysymyksissä olevat kuvat eivät siirry Question Manageriin.

Tallennus ei luo XML-tiedostoon encoding-attribuuttia, jonka takia kaikki ohjelmat, esim. SARAS QTI Viewer, eivät pysty avaamaan sellaisia tiedostoja, joissa on käytetty skandimerkkejä (ääö). Kysymykset saa avattua lisäämällä XML-tiedoston ensimmäiselle riville attribuutin encoding="ISO-8859-1".

6.3.3 Respondus

Respondus on monipuolinen kysymyseditori, jossa on helppo laatia monentyyppisiä kysymyksiä, määrittää niiden pisteytys ja halutut palautetekstit sekä lisätä tarvittavat kuvat kysymyksiin. Käytön aluksi valitaan jokin yhdeksästä ns. ”persoonallisuudesta”, esimerkiksi *WebCT 3.x-4.x* tai *IMS QTI 1.1+*. Ohjelman toiminnallisuus ja käytettävissä olevat kysymystyypit vaihtuvat valitun persoonallisuuden mukaan.

Respondus toimii hyvin etenkin WebCT:n kanssa; tehdyt kysymykset on mahdollista viedä suoraan ohjelmasta WebCT-palvelimen niille kursseille, joille käyttäjällä on oikeudet. Kysymykset lisätään WebCT-kurssin tehtävätietokantaan, ja samalla Respondus luo kurssille joko uuden testin tai liittää kysymykset johonkin valittuun jo olemassa olevaan testiin.

Valittaessa persoonallisuudeksi IMS QTI 1.1+, käytettävissä on seitsemän yleisintä IMS QTI -spesifikaation mukaista kysymystyyppiä. QTI-muotoiset kysymykset voi tallentaa yksittäiseksi XML-tiedostoksi, jolloin kysymyksiin liittyvät resurssitiedostot tallentuvat samaan kansioon. Respondus tukee IMS Content Packaging -spesifikaatiota, joten kysymykset voi tallentaa myös yhdeksi zip-tiedostoksi, joka sisältää kysymystiedoston, kaikki niihin liittyvät resurssitiedostot ja imsmmanifest.xml-julistustiedoston. Vastaavasti Respondus osaa avata sekä yksittäiset kysymystiedostot että zip-paketit.

Muilla ohjelmilla tehtyjä kysymyksiä avattaessa Respondus ei osaa näyttää skandimerkkejä (ääö) oikein, jos tiedostomuodoksi on tuontivaiheessa valittu oletuksena oleva *IMS QTI Question*. Avattavan tiedoston muodoksi onkin huomattava valita *IMS QTI ISO-8859 encoding*, jolloin merkistö koodataan tuontivaiheessa skandien kannalta oikeaksi.

Kysymyksiä Responduksesta tallennettaessa ohjelma asettaa merkistöksi eurooppalaiset merkit sisältävän ISO-8859-1:n ja koodaa skandimerkit tapauksesta riippuen desimaali- tai heksadesimaalikoodilla tai symbolisella notaatiolla. Esimerkiksi *ä* on desimaalikoodina `ä`, heksadesimaaleina `�E4;` ja symbolisella notaatiolla merkittynä `ä`. Tällä varmistetaan skandimerkkien näkyminen muissa ohjelmissa tai järjestelmissä, mutta kysymysten muokkaaminen niissä hankaloituu tekstin hankalan luettavuuden takia, vertaa kuva 16.

Kysymys	
Kategoria:	Työmenetelmät
*Otsikko:	Työmenetelmät
*Kysymys:	Mikä seuraavista työmenetelmistä on uusin?

Kuva 16. Skandien koodaus WebCT:lle vietyssä kysymyksessä.

6.3.4 Hot Potatoes

Hot Potatoes -ohjelmisto sisältää joukon osaohjelmia, joilla tehdään erityyppisiä kysymyksiä. Kysymyksistä luodaan www-sivu, jonka voi linkittää oppimateriaaliin. Tehtävien luonne on tällöin itsearviointia tukeva, sillä vastaukset eivät tallennu minnekään. Osa kysymystyypeistä on mahdollista tallentaa myös WebCT-muodossa joko WebCT:n itsearviointitehtäviin tai testeihin liitettäväksi. WebCT:n itsearviointitehtävissä vain sellaiset kysymystyypit ovat toimivia, joissa ainoastaan yksi vastausvaihtoehto on valittavissa.

Hot Potatoesilla luotuja kysymyksiä voidaan hyödyntää Moodlessa kolmella tavalla. Ensimmäinen osa kysymystyypeistä on tallennettavissa WebCT-muodossa, jonka jälkeen kysymykset ovat liitettävissä Moodlen tenttikysymyskategorioihin. Toinen vaihtoehto on tuoda tenttikysymyskategorioihin alkuperäisiä Hot Potatoes -tiedostoja. Tämä tuontitapa näyttää kuitenkin vielä varsin rajoittuneelta ja keskeneräiseltä, sillä vain JCross-ristisanatehtävät siirtyvät oikein Moodlen Lyhytvastaus-kysymyksiksi. Kolmantena vaihtoehtona Moodlessa on *Hot Potatoes kysely* -työkalu, joka mahdollistaa kaikkien viiden kysymystyypin liittämisen Moodleen Hot Potatoes -tiedostoina. Kysymykset esiintyvät ja käyttäytyvät Moodlessa aitojen Hot Potatoes -kysymysten tavoin ja pisteet suorituksista tallentuvat Moodleen.

Seuraavassa tarkastellaan Hot Potatoesia siitä näkökulmasta, miten sillä luodut kysymykset ovat siirrettävissä WebCT:lle ja Moodleen varsinaisiksi testikysymyksiksi, kun kysymykset on tallennettu WebCT-muodossa. Taulukossa 4 on lueteltu Hot Potatoesin kysymystyypit ja niiden siirrettävyys WebCT:lle ja Moodleen.

Taulukko 4. Hot Potatoes -kysymystyyppi.

Hot Potatoes -kysymystyyppi	WebCT-yhteensopivuus	Moodle-yhteensopivuus
JQuiz-monivalintatehtävä, tyyppi <i>Valinta (1 oikea)</i>	Kyllä	Kyllä
JQuiz-monivalintatehtävä, tyyppi <i>Valinta (monta oikein)</i>	Kyllä	Kyllä, muokkauksen jälkeen
JQuiz-monivalintatehtävä, tyyppi <i>Lyhytvastaus</i>	Kyllä	Kyllä
JQuiz-monivalintatehtävä, tyyppi <i>Hybridi</i>	Ei (siirtyy lyhytvastausteh- tävänä)	Ei (siirtyy lyhytvastausteh- tävänä)
JCloze-aukkotehtävät	Kyllä	Ei
JMatch-yhdistelytehtävät	Kyllä	Ei
JMix-sekoitetut lauseet	Ei	Ei
JCross-ristisanatehtävät	Ei	Ei

Lyhytvastaus-kysymystyyppi vastaa QTI-spesifikaation Fill-in-Blank-kysymystä, jossa vastaus annetaan kirjoittamalla. Hot Potatoesin erikoisuus on hybridi-tyyppinen monivalintatehtävä, joka esiintyy käyttäjälle ensin lyhytvastauksena, mutta muuttuu tiettyjen yrityskertojen jälkeen monivalintakysymykseksi. WebCT-muotoon tallennettaessa kysymys muotoutuu lyhytvastaukseksi.

Valinta (monta oikein) -tyyppiset kysymykset eivät siirry suoraan Moodleen kysymysten pisteytystavan takia. Moodle hyväksyy tehtävän vain jos oikeiden vastausten yhteenlaskettu maksimi on 100 %. Alla on esimerkki yhdestä kysymyksestä ja siihen tarvittavista muutoksista.

:TYPE:MC:N:0:A

:TITLE:Digitaalivalokuvaus (1)

:CAT:Digitaalivalokuvaus

:QUESTION:H

Mitkä seuraavista ohjelmista soveltuvat digivalokuvien käsittelyyn?

:ANSWER1:100:H **(Tälle riville 100:n tilalle 50)**

Adobe Photoshop

:REASON1:H

Oikein!

:ANSWER2:0:H

CorelDRAW

:REASON2:H

Väärin!

:ANSWER3:100:H **(Tälle riville 100:n tilalle 50)**

Paint Shop Pro

:REASON3:H

Oikein!

WebCT:lle vietäessä pisteytys toimii ongelmitta, koska kyseisessä järjestelmässä on Moodlea monipuolisemmat kysymysten asetusvaihtoehdot. Esimerkiksi tässä tapauksessa kukin oikea vaihtoehto voi olla 100 %, tai oikeiden vaihtoehtojen yhteenlaskettu prosenttimäärä voi olla 100; valinta kysymystä luotaessa riippuu siitä, millä tavalla se halutaan pisteyttää. WebCT osaa lukea kysymystiedoston pisteytyksen ja säätää asetukset sen mukaisesti sopiviksi.

Hot Potatoesilla luoduissa ja WebCT-muotoon tallennetuissa kysymyksissä on sama ongelma kuin monissa muissakin kysymyseditoreissa: erikoismerkkien näkyvyyden varmistamiseksi skandimerkit koodataan heksadesimaalikoodina, joka hankaloittaa kysymysten muokkaamista WebCT:llä ja Moodlella.

6.4 Sisällön paketointi

Sisällön paketoinnilla tarkoitetaan materiaalin koostamista SCORM- tai IMS Content Packaging (CP) -spesifikaatioiden mukaiseen muotoon. Monen sisällöntuotantovälineen esittelyteksteissä mainostetaan SCORM- ja/tai AICC-yhteensopivuutta, jotkut harvemmat ovat IMS CP -yhteensopivia. WebCT tukee IMS CP -paketointia, ja Moodle SCORM- sekä AICC-paketointia. Koska Moodlen AICC-pakettien tuontiominaisuus on tällä hetkellä uusi ja hieman keskeneräinen, AICC on rajattu tarkastelun ulkopuolelle.

WebCT:ssä sisältö on perinteisesti koostettu WebCT:n Sisältökokonaisuus-työkalulla kurssin tiedostoalueelle viedyistä tiedostoista. WebCT:n IMS CP -tuki mahdollistaa sisällön koostamisen myös kurssiympäristön ulkopuolella jollakin erillisellä ohjelmalla. Moodlella vastaava toiminnallisuus saadaan aikaan SCORM-työkalulla; sisältö sitä varten on kuitenkin koostettava jollakin muulla ulkopuolisella ohjelmalla.

Paketoitu sisältö eroaa yksittäisistä tiedostoista ensiksikin siinä, että sisältöpaketissa materiaalilla on tietty struktuuri. Tämän avulla sisältö on esitettävissä käyttäjille jäsennehtynä kirjan sisällysluettelon tapaan, kuva 17. Toinen etu on siinä, että sisältö on oppimisalustan kontrolloitavissa. Esimerkiksi WebCT-kurssin opettaja pystyy näkemään, millä sisältösivuilla oppijat ovat käyneet ja millä eivät. Moodlen SCORM mahdollistaa vielä pitemmälle viedyn seurannan, esimerkiksi materiaalissa mahdollisesti olevien tehtävien suoritus pisteet välittyvät oppimisalustaan.

WebCT:n Sisältökokonaisuus

Sisällysluettelo

1. Johdanto
2. Internet-selaimet
3. Kurssin avaaminen
4. Kurssinäkymä
5. Linkit ja kurssin rakenne
6. Käytettävissä olevat kurssityökalut
- ▼ 7. Perustyökalut
 - 7.1. Kalenteri
 - 7.2. Keskustelut
 - 7.3. Sähköposti
- ▼ 8. Sivujen ja työkalujen lisääminen, linkkien muokkaaminen
 - 8.1. Sivujen ja työkalujen lisäämisen työvaiheet
 - 8.2. Hierarkkisen rakenteen luominen
 - 8.3. Linkkien siirtäminen
 - 8.4. Linkkien asetukset
- ▼ 9. Liitteet
 - 9.1. Linkkejä lisätiedon lähteille

Moodlen SCORM



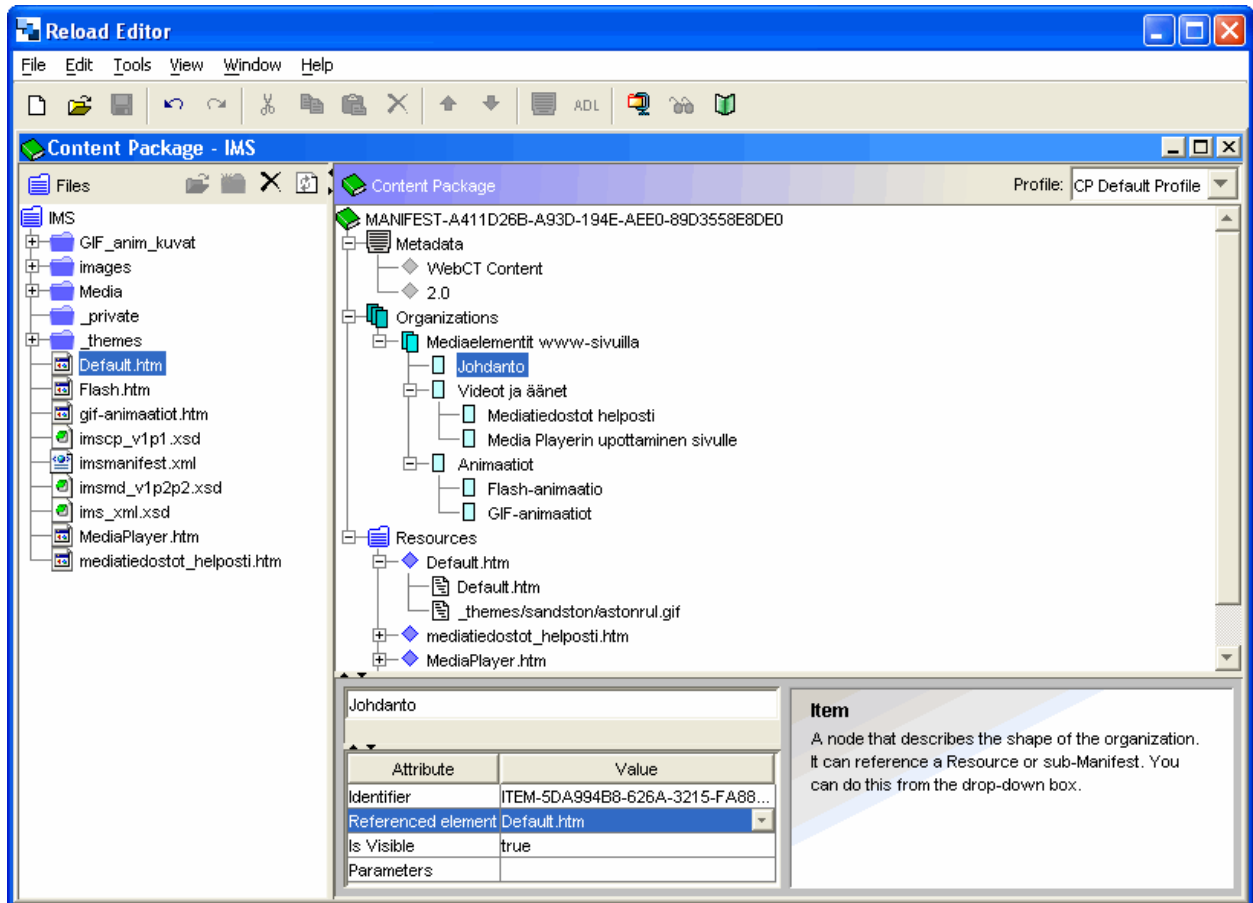
Kuva 17. Sisältömateriaalin sisällysluettelot WebCT:llä ja Moodlessa.

Sisällön paketointiin on saatavana on lukuisia kaupallisia sisällöntuotantovälineitä, kuten Coursegenie, Lectora Publisher, Thesis Professional ja Camtasia Studio, jotka tukevat yhtä tai useampia edellä mainittuja sisällönpaketoitintapoja. Näiden lisäksi on olemassa hyvä ja ilmainen avoimen lähdekoodin periaatteella tehty RELOAD Editor, jolla koostetaan sisältöpaketteja valmiista sisältömateriaaleista. Toinen ilmainen ohjelma on eXe, jolla voi helposti tehdä yksinkertaisia sisältöjä ja tallentaa ne sisältöpaketeiksi. Mainittakoon vielä ConceptTutor, joka niin ikään on ilmainen ja helppokäyttöinen ohjelma, mutta on tarkoitettu pelkästään käsitteiden (concepts) opettamiseen liittyvien oppimateriaalien tekemiseen.

6.4.1 RELOAD Editor

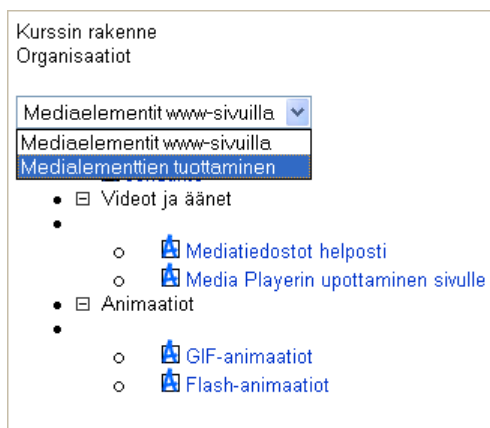
RELOAD Editor on työkalu, johon voi avata valmiita IMS CP - tai SCORM-sisältöpaketteja katseltavaksi ja muokattavaksi. Sen lisäksi editorilla voi koostaa paketteja lähdeaineistosta – myös useampien pakettien koostaminen yhdeksi laajemmaksi paketiksi on mahdollista.

Kuvassa 18 on RELOAD Editorin käyttöliittymä. Vasemmassa reunassa näkyvään tiedostoluetteloon tuodaan tarvittavat sisältötiedostot. Keskialueella olevassa Content Package -kehyksessä luodaan haluttu rakenne sisällölle. Näkyvä rakenne vastaa IMS CP - ja SCORM-spesifikaatioiden mukaisen imsmanifest.xml-tiedoston <organizations>-elementin rakennetta. Ikkunan alaosassa näkyvät XML-tiedoston elementtien attribuuttien arvot, joita voi tarvittaessa muokata.



Kuva 18. RELAOD Editorin käyttöliittymä.

Sisältörakenteen luominen aloitetaan napsauttamalla hiiren kakkospainikkeella Organizations-tekstiä ja valitsemalla Add Organization, joka lisää imsmanifest.xml-tiedostoon uuden <organization>-alielementin eli sisältöryhmän. Tämä näkyy kuvassa kuvassa 16 *Mediaelementit www-sivuilla* -kohtana. IMS CP -spesifikaation mukaan materiaalia ei voi lisätä suoraan <organizations>-elementin alle, joten ainakin yksi <organization>-elementti on pakko luoda. Käytännössä enempää ei tarvitse eikä kannatakaan luoda, jos koostettu materiaali on tarkoitus viedä WebCT:lle tai Moodleen. WebCT ei näytä sisällysluettelossaan <organization>-elementtien nimiä lainkaan; sisällysluettelo on muuten samanlainen kuin RELOADissa luotu, paitsi WebCT jättää nämä sisältöryhmien otsikot pois. Moodleissa sisältöryhmät ovat valittavissa alavetovalikosta, kuva 19.

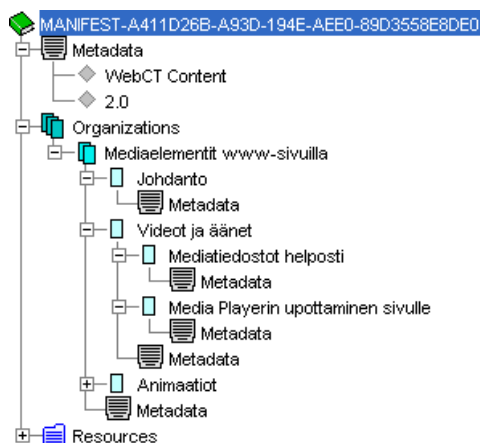


Kuva 19. Moodlen tapa esittää organization-sisältöryhmät.

Resurssitiedostot viedään halutun <organization>-elementin alle hiirellä vasemman reunan tiedostoluettelosta vetämällä. IMS CP:n mukaan <item>-elementtien alle voi luoda toisia <item>-alielementtejä, joka onnistuu RELOAD Editorissa kätevästi niin ikään hiirellä vetämällä. Samojen resurssitiedostojen käyttö eri kohdissa sisältöpakettia onnistuu myös spesifikaation mukaisesti.

Useimmiten sisällön koostamisessa käytetään html-tiedostoja, joissa sitten voi olla linkitettyinä erilaisia muita resurssitiedostoja, kuten kuvat, äänet, videot ja flash-animaatiot. Aivan vastaavalla tavalla koostaminen onnistuu myös esimerkiksi PDF-, Word- ja PowerPoint-tiedostoista.

Jokaiselle elementille on niin haluttaessa mahdollista lisätä metatietokuvaus, kuva 20. Tämä kuva on verrattavissa kohdassa 5.3.2 *Julistusosan elementit* olevaan kuvaan 11, jossa esitellään IMS CP:n rakennetta. Ylimpänä on koko pakettia kuvaava metatieto, muut ovat sisältöryhmiä (kuvassa *Mediaelementit www-sivuilla*) ja sisältöjä kuvaavia tietoja. Metatietojen syöttämistä varten avautuu oma ikkunansa, jossa on valittavana IMS LRM-, LTSN- ja UKCMF-profiilit, jotka sisältävät jonkin verran toisistaan poikkeavan sanaston. Myös muiden sovellusprofiilien, vaikkapa itse luotujen, lisääminen RELOADiin on mahdollista.



Kuva 20. Metatiedon liittäminen sisältöpakettin eri elementeille RELOAD Editorissa.

RELOADissa koostettu SCORM- tai IMS-paketti on siirrettävissä sellaisenaan ilman ongelmia Moodleen. WebCT:lle viettäessä joudutaan tekemään pieniä muutoksia imsmanifest.xml-julistustiedostoon. Ensiksikin <manifest>-elementin nimiavaruusmäärittäystä on muokattava Windowsin Muistiolla (Notepad), alla on alkuperäinen ja muutettu koodin osa.

Alkuperäinen:

```
<manifest xmlns="http://www.imsglobal.org/xsd/imscp_v1p1"
  xmlns:imsmd="http://www.imsglobal.org/xsd/imsmd_v1p2"
```

Muutettu:

```
<manifest xmlns="http://www.imsproject.org/content"
  xmlns:imsmd="http://www.webct.com/IMS"
```

Joissakin tapauksissa <metadata>-elementit saattavat sisältää <lom>-elementtejä, jotka estävät paketin siirtämisen WebCT:lle. Näin ollen kaikki <lom>-elementit on poistettava tiedostosta. Lopuksi tiedoston alussa olevan merkistön koodaustavan määrittävän encoding-attribuutin arvoksi on asetettava ISO-8859-1, jotta sisällysluettelossa mahdollisesti olevat skandimerkit näkyisivät oikein. Tiedosto on myös tallennettava ANSI-muodossa.

SCORM-pakettien koostaminen ja esikatselu

RELOAD Editorissa ei ole mahdollista lisätä SCORM-paketteihin niissä normaalisti olevia JavaScript-kutsuja, vaan se on tehtävä jossakin erillisessä ohjelmassa kuten muukin materiaalintuotanto. Editorissa pystyy kuitenkin katsomaan ja muokkaamaan kunkin pakettiin kuuluvan osan eli sisältöobjektin (item, SCO) ominaisuuksia. Mahdollista on valita, mitkä osat on käytävä läpi ennen kyseiseen osaan pääsemistä (prerequisites), miten pitkään tarkastelu on sallittu (max time

allowed), mitä tehdään kyseisen ajan kuluttua umpeen (time limit action), oppimisalustasta pyydetty tieto (data from LMS), sekä asettaa kysymyksille se pistemäärä (mastery score), joka on saavutettava, jotta kyseinen osa merkittäisiin suoritetuksi.

RELOAD Editorin esikatselutoiminto ei kykene simuloimaan sellaisia SCORM-sisältöjä, joissa on ohjelmoitua toiminnallisuutta. Tätä varten on olemassa erillinen RELOAD SCORM 1.2 Player, jossa sisältöpaketteja voi kokeilla ilman että niitä tarvitsee viedä oppimisalustalle. Playerissa on näkyvissä CMI-tietomallin eri osa-alueiden alkuarvot. Kun ne on otettu talteen, paketti käynnistetään ja navigoidaan läpi. Lopetuksen jälkeen tutkitaan, mitkä arvot ovat muuttuneet, josta voidaan päätellä kommunikointirajapinnan toimivuus.

Sisältöpakettien yhdistäminen ja purkaminen

RELOAD Editorilla on helppo tehdä kahden tai useamman sisältöpaketin yhdistäminen yhdeksi paketiksi Aggregate-toiminnolla ja vastaavasti niiden purkaminen takaisin erillisiksi paketeiksi Disaggregate-toiminnolla. Yksinkertaistettuna koodina imsmanifest.xml-tiedoston rakenne on seuraava kahden eri paketin koosteessa:

```

<manifest>
  <organizations/>
  <resources/>
  <manifest>
    <organizations>
      <organization>...</organization>
      <organization>...</organization>
    </organizations>
    <resources>...</resources>
  </manifest>
  <manifest>
    <organizations>
      <organization>...</organization>
    </organizations>
    <resources>...</resources>
  </manifest>
</manifest>

```

tyhjät elementit

ensimmäiseen koostepakettiin kuuluvat sisältöryhmät

ensimmäinen koostepaketti

toinen koostepaketti


Tässä esimerkissä molemmat alijulistukset (sub-manifest) ovat samantasoisia muuten tyhjän sisältöpakettin pääjulistuksen sisällä. Toinen mahdollisuus on tehdä kooste siten, että toinen sisältöpaketti on ensimmäisen alijulistuksena kuten julistusosan elementtejä esittävässä kuvassa 11. Lopputuloksen kannalta molemmat tavat ovat samanarvoisia.

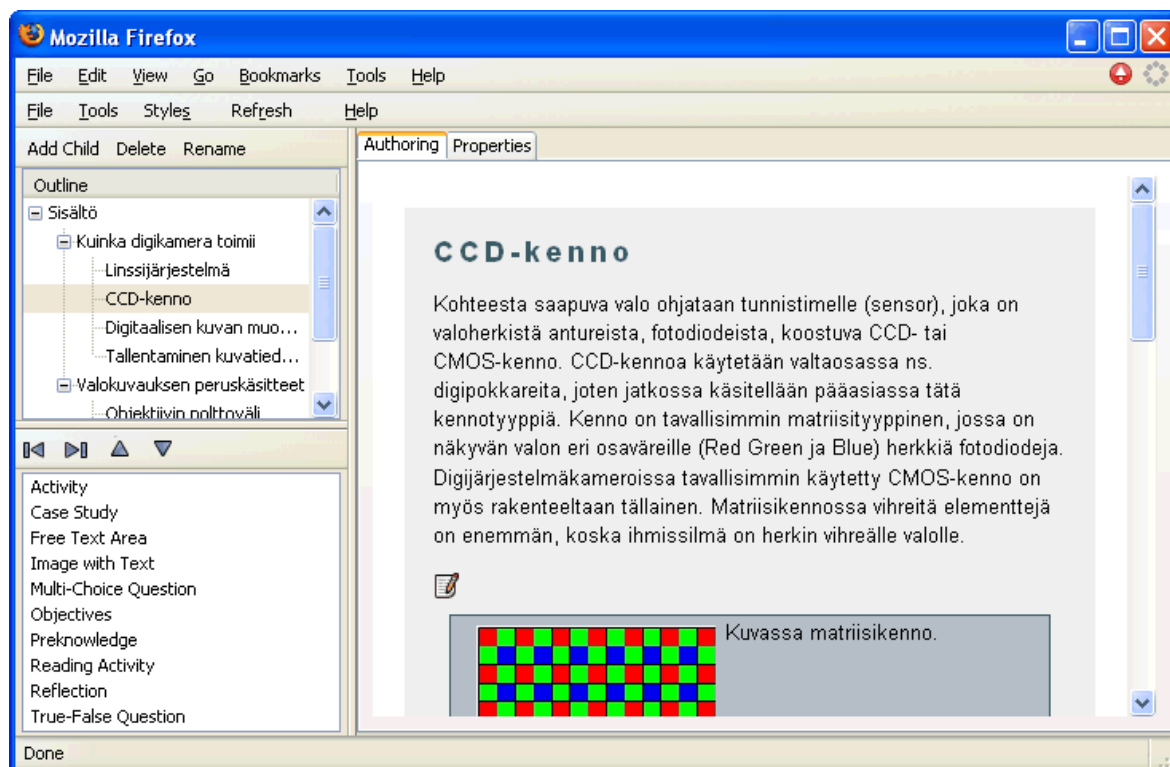
Oppimisalustalle vietyinä koostetun paketin eri osat näkyvät samalla tavalla eri sisältöryhminä kuten aikaisemmin kuvassa 19 on esitetty.

6.4.2 eXe – eLearning XHTML editor

Avoimen lähdekoodin periaatteella tehty eXe on varsin helppokäyttöinen työkalu oppimissisältöjen tekemiseen. Editorissa luodaan oppimateriaalisivuja halutunlaiseksi rakenteeksi (sisällysluetteloksi), kirjoitetaan sivuille tekstit, lisätään kuvat ja itsearviointikysymykset. Tehty sisältö tallennetaan jatkumuokkausta varten ohjelman omalla formaatilla. Lopputulos tallennetaan RELOAD Editorin tapaan IMS- tai SCORM-paketiksi oppimisalustoille siirrettäväksi.

eXe-editori on sikäli näppärämpi RELOADiin verrattuna, että asiasisältö voidaan luoda samassa ohjelmassa jossa esitysrakennekin luodaan. Toisaalta eXe on mediaelementtien suhteen rajoitetumpi, sillä käytettävissä ovat vain teksti ja kuvat. On kuitenkin huomattava, että työkalu on vasta aikaisessa kehitysvaiheessaan.

Editori tarvitsee toimiakseen Mozilla Firefox -selaimen. Käytön periaate on hyvin yksinkertainen: Add Child -toiminnolla (kuva 21) lisätään tyhjiä sivuja, joille luodaan sisältöä valitsemalla sopivia iDevices-työkaluja käyttöliittymässä vasemmalla alhaalla olevasta luettelosta. Kullekin sivulle on mahdollista lisätä useita iDevices-työkaluja (Instructional Devices) tarpeen mukaan. Tärkeimmät niistä ovat *Free Text Area*, jolla luodaan perustekstit, sekä *Image with Text*, joka mahdollistaa kuvien lisäämisen. Kuhunkin työkaluun pääsee lisäämään sisältöä -kuvakkeella.



Kuva 21. eXe-editorin käyttöliittymä.

eXe-editorin tallentamat SCORM- ja IMS-paketit avautuvat ongelmitta sekä RELOAD Editoriin että Moodleen. WebCT:lle viettäessä IMS-paketit vaativat imsmanifest.xml-tiedoston muokkaamista ennen kuin WebCT:n CMU hyväksyy paketin. Tarvittavat nimiavaruusmäärittämis- ja merkistönkoodausmuutokset on kuvattu RELOAD Editorin yhteydessä. Sen lisäksi <metadata>-elementin <schema>- ja <schemaversion>-elementtien arvoja on muutettava seuraavasti, muutettut kohdat on merkitty lihavoinnilla.

Alkuperäinen:

```
<metadata>
  <schema>IMS Content</schema>
  <schemaversion>1.1.3</schemaversion>
</metadata>
```

Muutettu:

```
<metadata>
  <schema>WebCT Content</schema>
  <schemaversion>2.0</schemaversion>
</metadata>
```

Moodleen ja WebCT:lle viedyissä paketeissa sisällysluettelon skandimerkit näkyvät oikein, samoin perustekstissä merkistö on kunnossa. Skandiongelma näkyy kuitenkin sisältösivujen otsikoissa. Tämä johtuu siitä, että html-tiedostoissa olevien otsikkotekstien skandimerkkejä ei ole koodattu symbolisella notaatiolla kuten muussa tekstissä.

eXen IMS- ja SCORM-paketteihin luomassa sisällysluettelossa on kauneusvirheenä yksi turha <item>-elementti, joka luo sisällysluetteloon ylimääräisen jäsennostason. Seuraavaan ims-manifest.xml-tiedoston koodiesimerkkiin tämä kohta on merkitty lihavoinnilla. Kohta näkyy esimerkiksi WebCT:llä kuvassa 22 näkyvänä *Sisältö*-tekstinä. Elementti on niin haluttaessa poistettavissa jossakin tekstieditorissa tai RELOAD Editorissa.

```
<organizations default="eXe_7418">
  <organization identifier="eXe_7418" structure="hierarchical">
    <title>Sisältö</title>
    <item identifier="ITEM-eXe_7419" isvisible="true" identifierref="RES-eXe_7741a">
      <title>Sisältö</title>
      <item identifier="ITEM-eXe_7741b" isvisible="true" identifierref="RES-eXe_7741c">
        <title>Kuinka digikamera toimii</title>
```

Sisällysluettelo

- ▼ 1. Sisältö
 - ▼ 1.1. Kuinka digikamera toimii
 - 1.1.1. Linssijärjestelmä
 - 1.1.2. CCD-kenno
 - 1.1.3. Digitaalisen kuvan muodostaminen
 - 1.1.4. Tallentaminen kuvatiedostoksi
 - ▼ 1.2. Valokuvauksen peruskäsitteet
 - 1.2.1. Objektiivin polttoväli
 - 1.2.2. Himmenninaukko ja objektiivin valovoima
 - 1.2.3. Kertauskysymyksiä

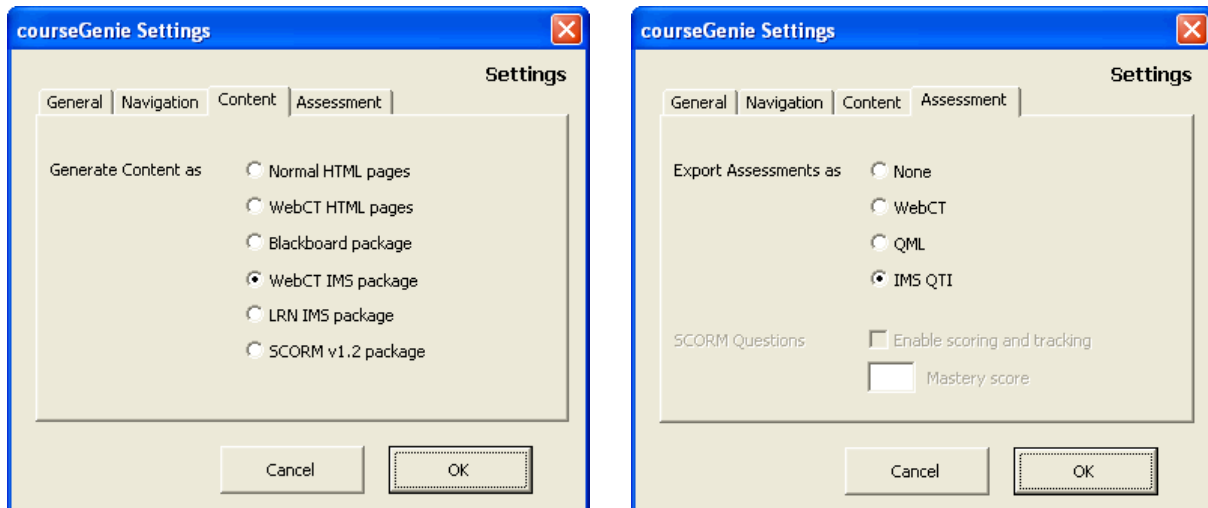
Kuva 22. Ylimääräinen Sisältö-taso WebCT:n Sisältökokonaisuuden sisällysluettelossa.

6.4.3 Coursegenie

Coursegenie on Microsoft Word -tekstinkäsittelyohjelmaan integroitava työkalu, jonka avulla Word-dokumenteista on helppo luoda kurssimateriaalia. Materiaali luodaan tuttuun tapaan Word-dokumentiksi, joka voi tekstin ja kuvien lisäksi sisältää myös flash-animaatioita, ääntä ja videoleikkeitä. Lisäksi materiaaliin on mahdollista luoda erityyppisiä kysymyksiä. Lopullista käyttöä varten dokumentti julkaistaan html-sivuiksi haluttuun käyttökohteeseen siirrettäväksi. Julkaistun dokumentin rakenne määräytyy sen perusteella, millä Coursegenien otsikkotyylillä mikin sisällön osa on Wordissa merkitty.

Sisältömateriaali ja kysymykset ovat tallennettavissa monissa eri muodoissa, kuva 23. Valinnalla *WebCT IMS package* sisältö tallennetaan siirrettäväksi WebCT:n 4.0- tai sitä uudemmille alustaversioille. Materiaali on helppo liittää WebCT:lle IMS CMU:n Tuo sisältöobjekti -toiminnolla, joka muotoilee IMS-paketista Sisältökokonaisuus-työkalulla jäsenetyn sisällön.

Vanhemmille WebCT-versioille, joissa CMU:ta ei ole, Coursegeniessä on tallennusvaihtoehto *WebCT HTML pages*. Materiaali esitetään niin ikään WebCT:n Sisältökokonaisuus-työkalulla, mutta sisällysluettelo on luotava käsin. SCORM-yhteensopivia järjestelmiä varten, kuten Moodle, on valittavissa tallennusmuoto *SCORM v1.2 package*.



Kuva 23. Coursegenien mahdollistamat sisällön ja kysymysten tallennusmuodot.

Sisältömateriaalin lisäksi Coursegeniessä on mahdollista luoda pelkkiä kysymyksiä. Käytettävissä on viisi peruskysymystyyppiä: Multi Choice, Multi Response, Text Entry, Matching ja Gap Fill. Luodut kysymykset on mahdollista tallentaa WebCT-, Questionmark QML - ja IMS QTI-Lite -muodoissa. Kaikki tehdyt kysymykset eivät kuitenkaan välttämättä tallennu riippuen valitun tallennusmuodon tukemista kysymystyypeistä. Toinen vaihtoehto on luoda kysymykset sisältötekstin yhteyteen itsearviointikysymyksiksi. Tällöin käyttäjä näkee, oliko vastaus oikein vai väärin, mutta niistä ei välity tietoa oppimisalustaan. Jos sisällön tallennusvaihtoehdoksi on valittu *SCORM v1.2 package*, asetuksissa on mahdollista valita otettavaksi arviointi ja seuranta käyttöön (*Enable scoring and tracking*), jolloin pisteet vastauksista tallentuvat alustajärjestelmään.

Coursegeniellä luodut WebCT IMS -sisältöpaketit siirtyivät WebCT:lle ongelmitta, samoin siirtyivät SCORM- ja WebCT IMS -paketit Moodleen. SCORM-pakettien html-sisältösivuille Coursegenie luo jostain syystä taulukkorakenteen, jonka vasemmassa solussa on sisällysluettelo ja oikeassa solussa varsinainen sisältö. Tämän takia käyttöliittymä sisältöön muodostuu hieman sekavaksi Moodleessa, koska näkyvillä on myös Moodleen oma navigointikehys, jossa tosin ei näy muuta kuin SCORM-paketin otsikko. Sisältöpaketit kannattaakin tallentaa Moodleen vietäväksi WebCT IMS -muodossa.

Muuten hyvässä ohjelmassa on yksi merkittävä puute suomalaisten käyttäjien kannalta, eli ongelma skandinaavisen merkistön esittämisessä. Coursegenie muuntaa sisältöpakettien XML-tiedostojen ja html-sisältösivujen merkistöt UTF-8-muotoon. Tällöin sisällössä olevat skandimerkit näkyvät oikein sellaisilla WebCT-kursseilla, jotka on alun perin luotu UTF-8-merkistölle. Kaikki nykyiset kurssit ovat kuitenkin ISO-8859-1-merkistöllä luotuja, joten Coursegeniellä luodussa materiaalissa sekä WebCT:n Sisältökokonaisuus-sisällysluettelon että html-sivujen skandit näkyvät väärin. Sisällysluettelo on Coursegenien luomassa imsmanifest.xml-tiedostossa, joten sen voi helposti muuntaa ISO-8859-1-merkistölle, mutta sisältösivujen muuttaminen on hankalaa.

Samanlainen skandiongelma kuin WebCT:llä on Moodleenkin viedyllä SCORM- tai IMS-pakettien sisältösivuilla. Sisällysluettelo puolestaan näkyy Moodleessa oikein silloin, kun imsmanifest.xml-tiedosto on tallennettu UTF-8-muodossa, toisin kuin WebCT:llä, joten kyseisen tiedoston merkistönkoodaustapaa ei tarvitse vaihtaa. On myös huomattava, että jos imsmanifest.xml-tiedostossa on skandimerkkejä, sisältöpaketti ei edes siirry Moodleen, ellei tallennusmuotona ole UTF-8. Kuriositeettina mainittakoon, että Moodle kuitenkin muutoin käyttää ISO-8859-1:tä pääasiallisena merkistönään, ja UTF-8:n käyttäminen WebCT:n tapaan jonkin yksittäisen kurssin merkistönä ei ole mahdollista.

7 SISÄLLÖN SIIRTÄMINEN

Seuraavissa kohdissa kuvataan erilaisia tapoja siirtää materiaalia WebCT:ltä Moodleen. Tarkastelun kohteena on ensinnäkin, miten yksittäisiä tiedostoja voidaan helposti siirtää alustalta toiselle. Seuraavaksi käsitellään kysymysten siirtämistä ja lopuksi WebCT:n Sisältökokonaisuustyökalulla koostettujen materiaalien siirtämistä. Koska alustat ovat ominaisuuksiltaan ja käyttötavoiltaan varsin erilaisia, tarkastelu on rajattu vain näihin kolmeen osa-alueeseen, jotka löytyvät samankaltaisina molemmista oppimisalustoista.

Siirtokokeet on tehty WebCT 4.0 - ja Moodle 1.5.2+ -versioilla.

7.1 Yksinkertainen tapa siirtää materiaalia kurssilta toiselle

Varsin usein kurssimateriaali on erilaisia toimisto-ohjelmilla luotuja tiedostoja, kuten PowerPoint-esitykset ja Word-dokumentit sekä niistä tehtyjä PDF-dokumentteja. Lisänä voi olla muita tiedostoja, kuten ääni-, video- ja html-tiedostot.

Sekä WebCT- että Moodle-oppimisalustoissa on kurssikohtaiset tiedostoalueensa. Tiedostoja voi tallentaa kurssilta omalle koneelle ja vastaavasti hakea omalta koneelta kurssin tiedostoalueelle. Tiedostoja voi siirtää vain yksi kerrallaan, joten käytännössä niistä kannattaa ensin muodostaa yksi zip-paketti. Moodlesta onnistuu sekä tiedostojen että kansioden liittäminen samaan pakettiin, WebCT:ssä toiminnon voi suorittaa vain tiedostoille. Molemmissa on mahdollista purkaa zip-paketteja, jotka sisältävät sekä tiedostoja että kansioita. Zip-pakettien käsittelyä varten kummassakin alustassa on valmis toiminto, joten se ei edellytä erillisen pakkausohjelman asentamista ja käyttämistä. On huomattava, että vaikka tässä puhutaan zip-paketeista, niin kyseessä eivät kuitenkaan ole Content Packaging -menetelmällä paketoitua sisältöä.

Kurssimateriaalin siirtäminen tapahtuu siis yksinkertaisesti siten, että tehdään kurssiympäristössä tiedostoista zip-paketti ja viedään se omalle koneelle. Seuraavaksi siirrytään toisella alustalla olevalle kurssille, ladataan ja puretaan paketti sinne haluttuun kansioon. Jos paketti sisältää sekä tiedostoja että kansioita, palautettu kansiorakenne säilyy alkuperäisen kaltaisena. Tämä helpottaa materiaalin siirtämistä Moodlesta WebCT:lle. Toisin päin siirrettäessä tiedostot on pakattava WebCT:llä kansioittain, jonka jälkeen Moodlesta on luotava vastaavat alikansiot ennen palautusta. Tämä altistaa virheille ja hidastaa prosessia erityisesti monimutkaisemmissa kansiorakenteissa, kuten web-sivustotyyppisissä sisältömateriaaleissa. WebCT:ltä Moodleen päin siirrettäessä kannattaakin hyödyntää WebCT:n WebDAV-ominaisuutta, jolla kurssin tiedostoalue saadaan

näkymään yhtenä oman koneen kansiona. Tällöin menetellään siten, että halutut tiedostot ja kansiot kopioidaan WebDAV-kansiosta johonkin oman koneen levyllä olevaan tilapäiskansioon, jossa zip-paketti luodaan jollakin koneelle asennetulla pakkausohjelmalla.

Moodleen on asetettu 10 Mt:n raja ladattaville tiedostoille. Raja ylittyy helposti, joten tiedostot on siirrettävä pienemmissä erissä, tai ne on ”tiputettava” ylläpitäjän toimesta oikeaan kansioon palvelinkoneella.

Siirretyt tiedostot linkitetään Moodlesta tarpeen mukaan kurssin etusivulle. Jos tiedostoja on paljon, on suositeltavaa, että ensin luodaan Moodlen *Tee uusi web-sivu* -toiminnolla eräänlainen luettelosivu, jonne sitten lisätään linkit tiedostoihin. Näin menetellen alkusivu saadaan pysymään käyttäjän kannalta selkeämpänä.

7.2 Kysymysten siirtäminen

7.2.1 Siirtäminen WebCT-muodossa

WebCT:ssä on viisi peruskysymystyyppiä, mutta monivalintakysymyksen asetuksia muuttamalla erilaisten vaihtoehtojen määrä kasvaa seitsemään. Moodlesta on samat kysymystyyppit kuin WebCT:ssä Essee-tehtävää lukuun ottamatta. Lisäksi Moodlesta on Tosi/Epätosi-kysymys, joka on WebCT:ssä yhdistetty Monivalintaan. Moodlesta on erilliset Numeerinen- ja Lyhyt vastaus -kysymykset, jotka WebCT:ssä on toteutettava samalla Sanallinen vastaus -kysymystyyppillä. Lisäksi Moodlesta on Aukkotehtävä-kysymystyyppi, jossa tekstiin on mahdollista luoda useita aukkoja. Sekä Moodlesta että WebCT:ssä kaikkiin kysymyksiin on mahdollista asettaa kuva kysymystekstin lisäksi.

Pääosin WebCT:n ja Moodlen kysymystyyppit vastaavat toisiaan. Niiden nimeämisessä on kuitenkin jonkin verran eroavaisuutta IMS QTI -spesifikaation käytäntöön verrattuna. Kummankin alustan sisältämät kysymystyyppit on esitetty taulukossa 5 niiden suomen- ja englanninkielisillä nimillä, sekä kysymysten IMS QTI -vastaavuus. Taulukon vaakariveillä olevat kysymystyyppit vastaavat jotakuinkin toisiaan loppukäyttäjän näkökulmasta. Teknisestä näkökulmasta katsottuna eri järjestelmien kysymystyyppit eivät välttämättä sisällä samoja ominaisuuksia.

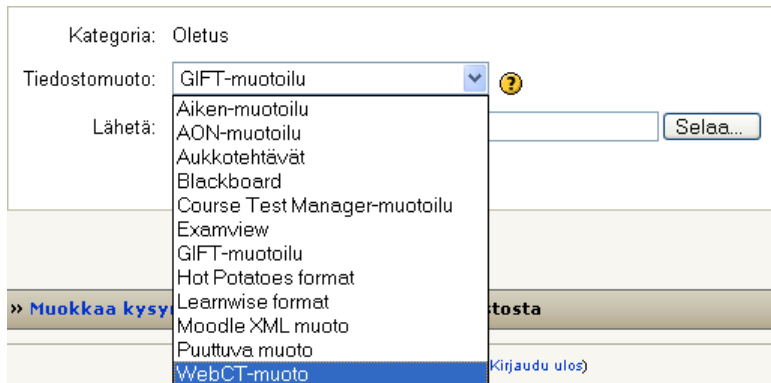
Taulukko 5. WebCT:n ja Moodlen kysymystyyppit ja niiden IMS QTI -vastaavuudet.

IMS QTI -kysymystyyppi	WebCT:n kysymystyyppi	Moodlen kysymystyyppi
Multiple Choice: monivalinta-kysymys, jossa vain yksi vastausvaihtoehto on valittavissa kerrallaan.	Monivalinta (Multiple Choice): määritetään kysymyksen asetuksissa.	Monivalinta (Multiple Choice): määritetään kysymyksen asetuksissa.
Multiple Response: monivalintakysymys, jossa useiden vastausvaihtoehtojen valitseminen on mahdollista.	Monivalinta (Multiple Choice): määritetään kysymyksen asetuksissa.	Monivalinta (Multiple Choice): määritetään kysymyksen asetuksissa.
True/False: kysymys, jossa oikein/väärin-vastausvaihtoehdot.	Monivalinta (Multiple Choice): kysymykselle asetetaan kaksi vastausvaihtoehtoa, joista vain jompikumpi on valittavissa.	Tosi/Epätosi (True/False)
Match Items: joukko tekstipareja, jotka on yhdistettävä.	Yhdistämistehtävä (Matching)	Yhteensopivat vastaukset (Matching)
Ei ole	Laskettu (Calculated): kysymykseen liittyy jokin kaava, jonka muuttujille arvotaan testitulanteessa arvot.	Lasku (Calculated): kysymykseen liittyy jokin kaava, jonka muuttujille arvotaan testitulanteessa arvot.
Fill-In-Blank (text): vastaukseksi annetaan lyhyt teksti	Sanallinen vastaus (Short Answer)	Lyhyt vastaus (Short Answer)
Fill-In-Blank (decimal, integer): vastaukseksi annetaan lukuarvo	Sanallinen vastaus (Short answer)	Numeerinen (Numerical Question)
Multiple Fill-In-Blank (text): aukko tehtävä, jossa useita aukkoja.	Ei ole	Aukkotehtävä (Cloze)
Short Answer: vastaus annetaan vapaamuotoisena tekstinä, jonka opettaja arvioi.	Essee (Paragraph)	Ei ole

WebCT:n kurssikohtaisesta kysymystietokannasta valitut kysymykset (yksi tai useampia) on mahdollista tallentaa tekstitiedostona omalle koneelle. Kysymyksiin liittyvät kuvatiedostot eivät kuitenkaan tallennu mukana. Kysymykset tallentuvat WebCT:n omassa formaatissa – muita tallennusmuotoja ei ole. Kokonaisia testejä kysymyksineen ja asetuksineen ei pysty tallentamaan, joten testit on luotava uudelleen Moodlella.

Moodlella on toiminto, jolla kysymyksiä voi tuoda tekstitiedostosta. Moodlen oma oletusformaatti on GIFT, mutta valittavana on myös muun muassa WebCT-formaatti (kuva 24). IMS QTI -muotoisten kysymysten tuontimahdollisuutta ei ole Moodleen toteutettu. Kun tarkastellaan taulukkoa 5, voidaan havaita, että lähes kaikille Moodlen (kuten WebCT:nkin) kysymystyypeille

löytyy vastaavuus IMS QTI -spesifikaation kysymystyypeistä. Näin ollen QTI-muotoisten kysymysten tuonti Moodleen pitäisi olla toteutettavissa, joten on syytä olettaa, että kyseinen toiminto on tulossa Moodleen seuraavien versioiden myötä. Mainittakoon, että kysymysten ja niihin liittyvien kuvien tallentaminen Moodlesta on jo mahdollista IMS QTI 2.0 -spesifikaation mukaisena zip-pakettina.



Kuva 24. Moodleen tuotavien kysymysten formaatit.

Valittaessa tuotavan tiedoston tyyppiä *WebCT-muoto*, kysymykset siirtyvät ongelmitta Moodleen, jos ne on WebCT:ssä laadittu ilman loogisuusvirheitä. Kuvat eivät kuitenkaan siirry kysymysten mukana, vaikka ne olisi erikseen tallennettu WebCT:ltä samaan kansioon kysymysten kanssa. Tuodut kysymykset toimivat samalla tavalla kuin WebCT:ssä arviointeinen ja palaute-teksteinen. Huomattavaa kuitenkin on, että WebCT:n Monivalinta-kysymystyyppin avulla toteutetut kaksi vaihtoehtoa sisältävät oikein/väärin-kysymykset eivät muunnu Moodlesta Tosi-/Epätosi-kysymyksiksi, vaan säilyvät monivalintakysymyksinä. WebCT:n Laskettu-tyyppiset kysymykset siirtyvät Moodleen vain osittain virheilmoituksen kera, vaikka kysymystyyppi on molemmissa varsin samanlainen. WebCT:n Essee-kysymykset eivät luonnollisestikaan ole siirrettävissä Moodleen, koska kyseinen kysymystyyppi puuttuu Moodlesta. Tätä kysymystyyppiä käytettäneen muutenkin varsin vähän, koska vastaukset on arvioitava opettajan toimesta.

Vaikka sekä WebCT:ssä että Moodlella on mahdollisuus järjestellä kysymykset aiheen mukaisiin kategorioihin, niin kategorioita ei voi sellaisenaan siirtää WebCT:ltä Moodleen. Suositeltavin tapa kysymysten siirtämisessä on se, että kysymykset tallennetaan WebCT:ssä kategorioittain tekstitiedostoiksi. Kaikki saman kategorian kysymykset valitaan lukuun ottamatta Essee- ja Laskettu-tyyppisiä kysymyksiä ja tallennetaan yhdeksi tiedostoksi. Tämän jälkeen Moodlella luodaan vastaavat kategoriat ja tuodaan kysymystiedostot niihin.

7.2.2 IMS QTI -muodon käyttäminen

WebCT:n IMS CMU:n Lataa sisältöobjekti -toiminnolla on mahdollista tallentaa kurssilla olevat testit, kyselyt ja kysymystietokannassa olevat kysymykset IMS-sisältöpaketti, kuva 25.

Lataa sisältöobjekti

Valitse ladattava sisältö

Testit, kyselyt ja kysymystietokanta
Lataa kaikki testit ja kyselyt kurssilla, mukaanlukien kaikki kysymykset kysymystietokannassa.

Sisältökokonaisuus
Lataa **yksittäinen** sisältökokonaisuus kaikkine sisältöineen objektina.
Sisältökokonaisuus:

Valitse kohde

Valitse hakemisto tiedostonhallinnan hakemistolistasta. Sisältöobjekti ladataan valitsemaasi kansioon.

Kohdehakemisto:

Kuva 25. Testien, kyselyjen ja kysymystietokannan tallennus WebCT:ltä.

Tallennettu sisältöobjekti on IMS CP:n mukainen zip-tiedosto. Tiedosto sisältää imsmanifest.xml-julistustiedoston ja alikansioita, joista löytyvät kysymyksiin liittyvät resurssitiedostot ja varsinaiset XML-kysymystiedostot. Julistustiedosto ei ole aivan IMS QTI:n syntaksin mukainen eikä siten avaudu suoraan esimerkiksi Respondus-ohjelmaan. Erästä alikansiosta löytyy kuitenkin questionDB.xml-tiedosto, joka sisältää kaikki WebCT-kurssin kysymystietokannassa olevat kysymykset. Tämä tiedosto on avattavissa muihin ohjelmiin, esimerkiksi Respondukseen, jotka tukevat QTI-muotoisten kysymysten sisään tuontia. Respondus ei osaa automaattisesti liittää kysymyksiin niissä olevia kuva- yms. mediatiedostoja, vaan antaa ilmoituksen puuttuvista resurssitiedostoista. Resurssitiedostot kuitenkin löytyvät sisältöpaketin alikansioista, joten puuttuvat resurssit on kohtuullisen pienellä vaivalla lisättävissä kysymyksiin.

Kysymysten siirron apuvälineenä käytettyyn Respondukseen WebCT:ssä luodut testit eivät avautuneet. WebCT:n IMS CMU:sta parhaan hyödyn saakin kysymyksiä ja testejä WebCT-kurssilta toiselle siirrettäessä. CMU vaikuttaa kuitenkin varsin käyttökelpoiselta siirtotavalta, jota kannattaa ehdottomasti hyödyntää, kun on tarve siirtää kysymyksiä IMS QTI -yhteensopiviin työvälineisiin ja järjestelmiin. Moodle ei tätä siirtotapaa vielä tue, mutta mahdollisesti jo seuraavissa versioissa kylläkin.

WebCT:n IMS-pakettien tallennusta voi hyödyntää myös Moodlen kanssa sikäli, että tällä tavalla saadaan kysymyksiin liittyvät kuvat tallennettua WebCT:ltä omalle koneelle. Jos WebCT:n tie-

dostoalueella on paljon kuvia ja muuta materiaalia, voi olla muuten vaikea tietää, mitkä kuvat liittyvät kysymyksiin ja mitkä johonkin muuhun käyttötarkoitukseen.

7.3 Materiaalin siirtäminen sisällön paketoinnin avulla

WebCT:n Sisältökokonaisuus-työkalulla koostettu materiaali on tallennettavissa IMS Content Packaging -spesifikaation mukaisena sisältöpakettina, joka pienen muokkauksen jälkeen on siirrettävissä Moodleen SCORM-pakettina. Kuvassa 17 sivulla 52 näkyy, miltä saman materiaalin sisällysluettelo näyttää WebCT- ja Moodle-kursseilla. Sisältö on sama, mutta esitystapa vaihtelee alustasta riippuen.

WebCT-kurssin Sisältökokonaisuuden tallentaminen tapahtuu jo aikaisemminkin esitellyn IMS CMU -sisällöntallennustoiminnon avulla. Sen Lataa sisältöobjekti -toiminnolla voidaan tallentaa halutut kurssilla olevat sisältökokonaisuudet IMS-sisältöpaketiksi, kuva 26.

Lataa sisältöobjekti

Valitse ladattava sisältö

Testit, kyselyt ja kysymystietokanta
Lataa kaikki testit ja kyselyt kursseilla, mukaanlukien kaikki kysymykset kysymystietokannassa.

Sisältökokonaisuus
Lataa **yksittäinen** sisältökokonaisuus kaikkine sisältöineen objektina.
Sisältökokonaisuus:

Valitse kohde

Valitse hakemisto tiedostonhallinnan hakemistolistasta. Sisältöobjekti ladataan valitsemaasi kansioon.

Kohdehakemisto:

Kuva 26. Sisältökokonaisuuden tallentaminen WebCT:ltä.

Tallennuksen jälkeen zip-paketti on purettava, ja sen sisältämää imsmanifest.xml-tiedostoa on hieman muokattava. WebCT jättää pois joistakin <resource>-elementeistä href-tribuutin. Virhe näyttäisi koskevan kaikkia sellaisia <resource>-elementtejä, jotka sisältävät vain yhden <file>-elementin. Puuttuvan tiedon saa lisättyä kopioimalla seuraavalla rivillä olevan <file>-alielementin href-tribuutin. On huomattava, että vain sellaisia <resource>-elementtejä tarvitsee muokata, joissa tyyppinä (type) on webcontent. Webctproperties-tyyppiset <resource>-elementit ovat Moodlen kannalta tarpeettomia ja ne voi joko poistaa tai antaa olla sellaisenaan. Seuraavassa on esitetty alkuperäinen ja muutettu esimerkkikoodi, lihavoinnit esittävät lisättyjä kohtia.

Alkuperäinen:

```
<resource identifier="CMD_4017643" type="webcontent">  
  <file href="CMD_4017639_M/my_files/Materiaali/alku.htm"/>
```

Muutettu:

```
<resource identifier="CMD_4017643" type="webcontent"  
  href="CMD_4017639_M/my_files/Materiaali/alku.htm">  
  <file href="CMD_4017639_M/my_files/Materiaali/alku.htm"/>
```

Muutokset tallennetaan UTF-8-merkistönkoodaustavalla, jotta sisällysluettelossa mahdollisesti olevat skandimerkit näkyisivät Moodlessa oikein. Tämän tallennusvaihtoehdon voi valita esimerkiksi Windowsin Muistiossa. Moodle ei ilmeisesti tutki XML-tiedoston alussa olevaa encoding-attribuutin arvoa, koska lopputuloksen kannalta on aivan sama, onko se ISO-8859-1 vaiko UTF-8. On huomattava, että tallennus on tehtävä UTF-8-muodossa senkin takia, että muutoin Moodle antaa virheilmoituksen eikä lataa sisältöpakettia, jos imsmanifest.xml-tiedosto sisältää skandimerkkejä.

Lopuksi päivitetty imsmanifest.xml-tiedosto ja kaikki muut pakettiin kuuluvat tiedostot pakataan zip-tiedostoksi ja ladataan Moodleen SCORM-työkalun avulla.

8 SUOSITUS TOIMENPITEIKSI SISÄLLÖN LUOMISESSA JA SIIRROSSA

8.1 Kysymysten ja testien luominen

Kokeilujen perusteella oli nopeasti havaittavissa, että harvassa sovelluksessa on toteutettu kaikkia IMS QTI -spesifikaation ominaisuuksia. Tästä seuraa se, että kysymyksiä sovelluksesta toiseen siirrettäessä osa ominaisuuksista ei siirry ollenkaan tai siirtyy väärin. XML-tiedostojen muodostamisessa on myöskin erilaisia tapoja. Tästä voi seurata esimerkiksi se, että yhdellä ohjelmalla tehdyt kysymykset eivät avaudu toisessa skandimerkkien takia, jos merkistön koodaus-tapamerkintä puuttuu. Skandimerkit ovat ongelmallisia myös sikäli, että jotkut ohjelmat koodaavat erikoismerkit sellaiseen muotoon, että kysymysten muokkaaminen muissa ohjelmissa hankaloituu tekstin vaikean luettavuuden takia.

Moodlessa ja WebCT:ssä on valmiit toiminnot kysymysten ja testien luomiseen; työkalut ovat varsin hyvät ja monipuoliset. Näin ollen erillisten kysymyseditorien käyttäminen ei ole sinällään tarpeellista. Kysymys on ennemminkin siitä, miten kysymyksiä ja testejä hallitaan – luodaanko ne jokaiselle kurssille erikseen vai voisiko niitä hyödyntää kurssilta toiselle? Sekä Moodlessa että WebCT:ssä on mahdollisuus siirtää kysymyksiä saman alustan sisällä kurssilta toiselle tekstitiedostona (testien siirtäminen ei ole mahdollista). Parempi kuitenkin olisi, jos kysymykset saisi ilman siirtämistä näkymään saman käyttäjän eri kursseilla. Tämä on toteutettu Moodlessa, tosin hieman kömpelösti: kunkin kysymyskategorian voi julkaista näkymään globaalisti, mutta jos kategoriat on huonosti nimetty ja niitä on paljon, oikean löytäminen on hankalaa. Lisäksi kaikki muutkin käyttäjät näkevät julkaistut kategoriat ja voivat vaikkapa poistaa niissä olevia kysymyksiä.

WebCT:llä ainoa mahdollisuus saada kysymykset näkymään muilla kursseilla on siirtää kysymykset tekstitiedostona kurssilta toiselle. Tällöin ongelmana on muistaa tai saada selville, että mikä on se kurssi, joka sisältää kaikki kysymykset. Kysymyksiä siirrettäessä ongelmana on lisäksi se, että WebCT ei luo kohdekurssille automaattisesti kysymyskategorioita, joka hankaloittaa merkittävästi suuren kysymysjoukon hallinnointia.

Yksi mahdollisuus ratkaista kysymysten hallinnointi on käyttää jotakin ulkopuolista ohjelmaa, jolla kysymykset luodaan. Kysymykset säilytetään omalla tai mieluummin verkon levyllä, josta kootaan tarvittavat kysymykset eri kursseille. Tällä hetkellä IMS QTI -muotoa käyttävistä kysymyseditoreista ei ole EKAMK:lle hyötyä, sillä käytettävissä olevat WebCT:n ja Moodlen versiot eivät tunnista kyseistä formaattia. Respondus on kuitenkin sellainen ohjelma, joka toimii erityi-

sen hyvin WebCT:n kanssa ja tukee myöskin QTI-muotoa. Tällöin jokaisella WebCT-käyttäjällä, joka tekee kysymyksiä kursseille, tulisi olla ohjelma käytettävissään.

Toisena vaihtoehtona kysymysten luomiseen on Hot Potatoes -ohjelmisto, joka on laajalti käytetty. Tästä syystä Moodleenkin on tehty tuki Hot Potatoes -tiedostomuodolle. Varsinaisten testikysymysten tekemiseen Hot Potatoes ei kuitenkaan sovellu hyvin, sillä kysymysmuodoista vain osa on siirrettävissä WebCT:lle ja Moodleen.

Erillisiä kysymyseditoreja parempi vaihtoehto olisi sellainen erillinen järjestelmä, oppimateriaalipankki, joka tukisi kysymysten luomista ja tallentamista. Kysymykset olisi mahdollista luokitella eri kategorioihin joko kaikkien käyttäjien yhteiseen käyttöön tai käyttäjäkohtaisesti. Kursseilla tarvittavat kysymykset kerätään materiaalipankista, tallennetaan tiedostoksi ja siirretään kurssiympäristöön. Tulevaisuutta ajatellen oppimateriaalipankin tulisi tukea IMS QTI -spesifikaatiota.

Testien siirtäminen ei ole WebCT:n ja Moodlen välillä mahdollista. WebCT:n Content Migration Utilityn avulla kysymykset ja testit pystytään kyllä tallentamaan IMS CP -sisältöpaketiksi, mutta sitä ei ole mahdollista siirtää Moodleen testiksi. Testit on muuten varsin helppo luoda kohdejärjestelmässä, mutta ongelma on lähinnä siinä, että miten saada lisättyä juuri oikeat kysymykset kuhunkin testiin. Tässä suhteessa testien siirrettävyydestä olisi etua.

8.2 Sisällön luominen

8.2.1 Työkalut

RELOAD Editor on teknisesti toimiva ja varsin helppokäyttöinen työväline valmiiden sisältöjen koostamiseen WebCT:tä ja Moodlea varten. Se soveltuu erinomaisesti myös muualla tehtyjen IMS- ja SCORM-pakettien katsomiseen ja kokeilemiseen ilman että niitä tarvitsee ensin ladata oppimisolustalle. Ongelmana RELOADin käytössä on ainoastaan sisältömateriaalien tuottaminen, joka on tehtävä ensin jollakin muulla välineellä, esimerkiksi Microsoft FrontPagella.

RELOAD Editoria vieläkin helpompi työväline on eXe XHTML editor, jolla myös sisällön tuottaminen on mahdollista IMS- ja SCORM-paketoinnin lisäksi. Uudelleenkäytettävyyden rajoitteena tosin on se, että tekovaiheessa kaikki tekstisisältö tallentuu ohjelman omaan tiedostomuotoon ollen saatavissa html-sivuina vasta IMS-/SCORM-julkaisun jälkeen. Ohjelma on varsin

valmiin oloinen vaikka onkin vasta kehityksensä alkuvaiheessa. Kehitysvaiheen tiliin voitaneen laittaa julkaistun materiaalin otsikoiden skandiongelmat.

Coursegenie on parhaimmillaan tuottaessa sisältöä, johon voi kuulua myös erityyppisiä kysymyksiä. Pelkäksi kysymysheditoriksi tätä työvälinettä ei kuitenkaan ole tarkoitettu. Kyseessä olisi varsin kätevä työkalu sisällöntuotantoon, mutta koska skandimerkit eivät näy oikein WebCT:llä eivätkä Moodlessa, Coursegenie on käytännössä käyttökelvoton. Ohjelman tuleviin versioihin on valmistajalta saadun tiedon mukaan harkinnassa merkistönkoodauksen valintamahdollisuus.

SCORM-sisältöjen osalta on todettava, että helppokäyttöisiä työkaluja sellaisten SCORM-pakettien tuottamiseen, jotka sisältävät oppimisalustan kanssa kommunikoivia sisältöobjekteja, ei näyttäisi markkinoilla olevan. Lisäksi tällaiset monimutkaisemmat SCORM-toteutukset näyttävät olevan varsin herkkiä virheille joko oppimisalustan SCORM-rajapinnan tai oppimisasihoiden toteutuksesta johtuen. Kaiken kaikkiaan ohjelmoitujen SCORM-sisältöjen tuottaminen vaikuttaa varsin suuritöiseltä. Käytännössä suurimpaan osaan tarpeista riittävätkin tavalliset SCORM- ja IMS-sisältöpaketit.

8.2.2 Sisällön tuottamisen toimintamalli

Sisällön luomiseen tarkoitettujen työkaluohjelmien täytyisi olla nopeasti omaksuttavissa, helposti ja edullisesti saatavilla, sekä niillä tuotetun materiaalin pitäisi olla vaivattomasti uudelleenkäytettävää. Edellä kuvattiin joitakin sisällöntuotantovälineitä, mutta niistä eikä muista vastaavista löydy yhtään sellaista, joka täyttäisi esitetyt kriteerit.

Yhtenä vaihtoehtona on jatkaa samalla linjalla kuin ennenkin, eli sisältömateriaalia tuotetaan Word-tekstinkäsittelyohjelmalla DOC-tiedostoina oppimisalustalla jaettavaksi. Wordin rinnalla voidaan käyttää FrontPagea, jolla www-muotoisen materiaalin tekeminen onnistuu pienen koulutuksen jälkeen. Näillä välineillä luotuja tiedostoja voidaan viedä oppimisalustalle sellaisenaan, tehdä www-sivustoksi ja linkittää se oppimisalustalle, tai koostaa tiedostoista RELOAD Editorilla IMS- tai SCORM-sisältöpaketteja WebCT:lle tai Moodleen vietäväksi. Käytettävyyden kannalta olisi parasta, jos materiaali on html-sivuina, vaikka sisällön paketointi mahdollistaa kaikenlaisten tiedostomuotojen käyttämisen.

Sisällön paketointia hyödyntämällä saavutettaisiin Moodle-ympäristön käytettävyyttä ajatellen selvää etua. Moodle-kurssien toteutushan perustuu varsin pitkälle siihen, että kurssin etusivulle

luodaan linkkejä eri oppimistyökaluihin kuten myös sisältötiedostoihin. Tästä seuraa hyvin nopeasti se, että käyttäjän on jatkuvasti vieritettävä sivua ylös ja alas siirtyessään kurssin aiheesta toiseen. Kaiken lisäksi etusivun päivitys hidastuu sitä mukaa kuin linkkien määrä siinä lisääntyy. Tämän takia olisi suotavaa luoda kurssin sisältömateriaali jollakin muulla tavalla kuin useina erillisinä Word-, PowerPoint- ja PDF-tiedostoina. Vaihtoehtoina ovat juuri sisällön paketointi, tai sitten luodaan erillinen ns. luettelosivu Moodlen *Tee uusi web-sivu* -toiminnolla, jonne tarvittavat tiedostolinkit ryhmitellään. Käytännössä jälkimmäinen on huomattavasti helpompi toteuttaa.

WebCT:ssä on oma Sisältökokonaisuus-työkalunsa, jonka avulla sisältöjen koostaminen onnistuu helposti kurssialueelle tuoduista valmiista tiedostoista. Vaihtoehtoinen ratkaisu olisi se, että sisältö koostetaan kurssialueen ulkopuolella esimerkiksi RELOAD Editorilla, jolloin sisältöpaketit ovat helposti muuallakin uudelleenkäytettävissä.

Sisällön paketoinnin toteuttamisessa eräs merkittävä ongelma on oppimateriaalin luomisessa ja hallinnassa. Sisällön tuottajan on luotava koostettava materiaali itse käytettävissään olevilla työkaluilla, ja jos valmista lähdemateriaalia olisikin olemassa, sitä ei ole missään keskitetysti saatavilla. Tämä johtaa siihen, että materiaalien tekninen toteutus on hyvin kirjavaa, kun sitä luodaan monilla erilaisilla välineillä ja tavoilla.

Toimintamalli, jossa sisältöä luodaan esimerkiksi FrontPagella ja koostetaan RELOAD Editorilla, on suositeltavissa ainoastaan tietoteknisiltä taidoiltaan edistyneemmille käyttäjille. Oppimateriaalien luomisen ja hallinnan ongelmaan sopiva ratkaisu peruskäyttäjienkin ajatellen voisi olla oppimateriaalipankki, jossa olisi saatavilla erityyppisiä oppimisaihioita joko sellaisenaan kursseilla käytettäväksi tai koostettavaksi. Koostetut materiaalipaketit voitaisiin tallentaa niin ikään materiaalipankkiin. Käyttöä varten materiaalia olisi helppo etsiä metatietokuvausten perusteella. Oppimisalustalle siirtämistä varten aineistot on ensin tallennettava omalle koneelle, josta ne voidaan siirtää joko kurssiympäristön tiedostoalueelle tai liittää suoraan IMS- tai SCORM-sisällöksi. Erillinen oppimateriaalipankki sopisi hyvin sellaisten oppimisalustojen – kuten WebCT ja Moodle – yhteydessä käytettäväksi, joissa ei ole omia sisällönhallintaominaisuuksia.

8.3 Sisällön siirtäminen WebCT:ltä Moodleen

WebCT:n ja Moodlen erilaisuudesta johtuen sisällön siirtämisessä on rajoitettu tarkastelemaan vain yksittäisten tiedostojen, kysymysten ja Sisältökokonaisuus-työkalulla koostettujen oppisältöjen siirtämistä WebCT-kursseilta Moodleen. Täydellistä kurssien siirrettävyyttä ei ole syytä

tavoitella senkään takia, että kurssit pitäisi suunnitella pedagogis- eikä materiaalilähtöisesti oppimisalustan ominaisuudet huomioiden.

Työhön liittyen laadittiin ohjeistus, jossa kuvataan ne työvaiheet, joilla yksittäisiä tiedostoja, kysymyksiä ja Sisältökokonaisuus-työkalulla koostettuja sisältöjä voidaan siirtää WebCT:ltä Moodleen. Ohje löytyy osoitteesta

http://www.corrie.fi/oppipooli/download/Materiaalin_siirto_WebCT_Moodle.pdf

Sisältömateriaali on siirrettävissä joitakin harvoja kysymystyyppisiä lukuun ottamatta.

Menettelyohjeet kattavat suurimman osan olemassa olevista kurssisisällöistä. Käytännössä joudutaan kuitenkin tekemään sisällön uudelleenjärjestelyä Moodlen puolella etenkin silloin, kun kyse on yksittäisistä tiedostoista. WebCT ja Moodle poikkeavat toisistaan kurssirakenteeltaan, mutta silti samankaltaiset ratkaisut ovat mahdollisia, jos on tarpeen. Materiaalin siirtäminen WebCT:ltä Moodleen on teknisesti varsin suoraviivaista, mutta tässä vaiheessa kannattaa kuitenkin miettiä uudelleen pedagogista toteutusta ja sen mukaista materiaalin sijoittelua. Kuten kohdassa 8.2.2 *Sisällön tuottamisen toimintamalli* on jo mainittu, niin Moodleessa kannattaa hyödyntää erillisen luettelosivun käyttämistä tiedostojen linkittämisessä, jolla voidaan parantaa kurssin käytettävyyttä.

Siirtoprosessin pitäisi onnistua ohjeita noudattamalla keskimääräiset tietotekniikkataidot omaavalta opettajalta. Suositeltavaa olisikin, että siirron tekisi kurssista vastaava opettaja itse, koska toteutuksesta voi muutoin tulla liiaksi aihe- eli materiaalipainotteinen. Moodlelessahan on mahdollista valita aihe- tai viikkopohjainen kurssinäkökulma, ja ulkopuolisen on melko vaikea suunnitella viikkopohjaista toteutusta.

8.4 Spesifikaatiot sisällöntuotannossa

Sisällöntuotannossa kannattaa suosia työvälineitä, jotka tukevat yleisimpiä spesifikaatioita. Tällainen on kysymysten ja testien osalta IMS QTI, jota useimmat kysymyseditorit ja monet oppimisalustat tukevat. Sisällön paketoinnissa käytetään IMS CP:tä ja SCORMia, jonkin verran myös AICCia. Selvää paremmuusjärjestystä IMS CP:n ja SCORMin kesken on vaikea määrittää. Koska SCORM perustuu IMS CP -paketointiin, sisällöt siirtyvät ristiin ongelmitta tai pienin muokkauksin; WebCT-tukee IMS CP:tä, mutta ottaa vastaan myös SCORM-paketteja. Vastaavasti Moodleen, jossa on SCORM-tuki, siirtyvät IMS CP -paketit ongelmitta. Tällä hetkellä oppimisalustoissa ja sisällöntuotantotyökaluissa on pääasiallisimmin tuki SCORM-spesifikaatiolle.

Koska SCORM-spesifikaatio on laajempi sisältäen mahdollisuuden oppimisaihioiden ja alustan väliseen vuorovaikutukseen, on SCORM-yhteensopiva työkalu käyttökelpoisempi valinta pelkän IMS CP -paketoinnin mahdollistavaan työkaluun verrattuna.

9 POHDINTA JA JOHTOPÄÄTÖKSET

9.1 Työn tavoitteisiin pääseminen ja tulokset

Alussa esitetyistä tutkimusongelmista suurimpaan osaan löydettiin vastaukset. Vastauksena kysymykseen, miten kurssisisällöt kannattaisi alun perin luoda uudelleenkäytettävyyks ja siirrettävyyks huomioiden, on yleisten spesifikaatioiden hyödyntäminen. Testikysymysten luomisessa suositeltavinta on suosia sellaisia työkaluja, jotka käyttävät IMS QTI -muotoa kysymysten tallentamiseen. Tällä hetkellä EKKY:ssä käytettävät Moodle- ja WebCT-versiot eivät kuitenkaan kyseistä muotoa pysty juurikaan hyödyntämään. Sisältömateriaalien osalta puolestaan kannattaa käyttää työkaluja, jotka tukevat sisällön paketointiin tarkoitettuja SCORM- tai IMS CP -muotoja.

Työssä on myös etsitty vastausta kysymykseen, millä välineillä IMS- tai SCORM-spesifikaatioiden mukaista sisältömateriaalia on mahdollista tehdä. Erilaisia näitä määrittäviä välineitä näyttäisi olevan runsaasti markkinoilla, joista muutamia koekäytettiin. Pääasiassa tulokset olivat myönteisiä, eli sisältöjä on mahdollista koostaa järkeviki kokonaisuuksiksi, jotka ovat WebCT:llä ja Moodlessa esitettävissä kirjan sisällysluetteloa muistuttavan käyttöliittymän kautta. Oppimisalustan kanssa kommunikoivien SCORM-sisältöjen toteuttaminen vaikuttaa kuitenkin varsin monimutkaiselta, eikä niiden tekemiseen löytynyt helppokäyttöisiä työvälineitä. Merkittävimpana ongelmana voidaan pitää monien työkalujen huonoa tukea skandimerkeille, jonka takia ääö-kirjaimet näkyvät väärin oppimisalustoille viedyillä sisältösivuilla.

Vastauksena kysymykseen IMS- ja SCORM-spesifikaatioiden keskinäisestä yhteensopivuudesta ja paremmuudesta toisiinsa verrattuna todettiin, että koska SCORM perustuu IMS CP -paketointiin, sisällöt siirtyvät ristiin alustalta toiselle sellaisenaan tai pienin muokkauksin. Kyseessä eivät siis ole keskenään kilpailevat erilaiset määrittäykset. Koska SCORM on IMS CP:tä laajempi spesifikaatio, on sisällöntuotantovälinettä valittaessa kannattavampaa valita SCORMia tukeva työkalu. Kysymykseen, onko jompikumpi näistä spesifikaatioista saavuttamassa valta-aseman, voidaan todeta, että sisällöntuotantotyökaluissa SCORM-yhteensopivuus on merkittävästi yleisempi.

Keskeisimpiä asetettuja kysymyksiä oli, voidaanko olemassa olevat kurssisisällöt siirtää alustalta toiselle ja miten se tehdään. Työn tuloksena on sisällön siirtämiseen tehty ohjeistus, josta on kerrottu kohdassa *8.3 Sisällön siirtäminen WebCT:ltä Moodleen*. Siirtäminen onnistuu varsin helposti kursseilla käytettävien keskeisten materiaalien osalta, joita ovat yksittäiset tiedostot, testikysymykset ja WebCT:n Sisältökokonaisuus-työkalulla koostetut sisällöt.

Työn yhtenä tavoitteena oli luoda toimintamalli ja ohjeistus kurssisisältöjen luomiseen ja siirtämiseen alustalta toiselle. Tähän tavoitteeseen päästiin vain osittain aiemmin mainitun sisällön siirtämiseen tehdyn ohjeistuksen osalta. Sisällön luomiseen sen sijaan ei löydetty mitään sellaista yksittäistä työkalua, jota voisi erityisesti suositella. Parhaimmalta ratkaisulta vaikuttaisi sen tyyppinen järjestelmä, jossa samasta löytyisi välineet sekä sisällön tuottamiseen että sen hallintaan. Tämä toteutuisi LCMS- eli oppimateriaalien hallintajärjestelmissä. Nyt EKKY:ssä on kuitenkin käytössä Moodle ja WebCT:stä sellainen versio, jotka eivät mahdollista tällaista.

Oppimateriaalien uudelleenkäytettävyyteen näyttäisi eniten vaikuttavan niiden löydettävyys, ei niinkään se, missä muodossa tai minkä kokoinen oppimisaihio on. Tästä huolimatta oppimateriaalien tuottamisessa kannattaa hyödyntää teoriaosassa kuvattuja sisällöntuotantoon tehtyjä spesifikaatioita. Tällä tavoin saadaan aikaiseksi määrämuotoisia sisältöjä, oppimisaihoita, jotka ovat helpommin hyödynnettävissä kyseisiä spesifikaatioita tukevilla oppimisalustoilla, eli mahdollinen alustanvaihto on jatkossa helpompaa.

9.2 Jatkossa selvitettävää

EKAMK/EKKY:n oppimateriaalien hallinnan järkipäristämiseksi ja löydettävyyden parantamiseksi on syytä selvittää, millaisia valmiita oppimateriaalipankkitoteutuksia on olemassa, vai olisiko järkevää luoda sellainen itse. Hallintaominaisuuksien lisäksi olisi toivottavaa, että järjestelmä sisältäisi myös helpot työvälineet sisältöjen luomiseen ja yhdistelyyn. Tämän lisäksi olisi syytä tarkastella vaihtoehtoa, jossa siirryttäisiin käyttämään WebCT:n versiota 6 ja siihen integroitavissa olevaa Learning Object Manager -oppimateriaalipankkia – olisiko oppimateriaalien luominen ja hallinta toteutettavissa tällä kokoonpanolla?

Tämän työn ulkopuolelle on jätetty opetuksen suunnittelu, jota varten on olemassa IMS Learning Design (LD) -spesifikaatio. LD:n mukaisia työkalujakin on jo olemassa, esimerkiksi samalta valmistajalta, joka on tehnyt RELOAD Editorin. LD:n ominaisuudet ja mahdollisuudet olisi syytä selvittää – mitä sillä on annettavaa kurssien suunnitteluun.

LÄHDELUETTELO

ADL. 2005. Advanced Distributed Learning Initiativen verkkosivut [viitattu 14.3.2005]. Saatavissa: <http://www.adlnet.org>

AICC. 2004. Aviation Industry CBT Committeen verkkosivut [viitattu 28.1.2005]. Saatavissa: <http://www.aicc.org>

Bailey, W & Currier, S. 2003. What Is ADL SCORM? [verkkodokumentti]. CETIS Standards Briefing Series 12.5.2003 [viitattu 14.1.2005]. Saatavissa: http://www.cetis.ac.uk/lib/media/WhatIsSCORM_web.pdf

Bohl, O. & al. 2002. The Sharable Content Object reference Model (SCORM) – A Critical Review [verkkodokumentti]. Proceedings of the International Conference on Computers In Education (ICCE'02) [viitattu 26.5.2005]. Saatavissa: <http://www.dlsi.ua.es/~carrasco/xmldocs/0283.pdf>

CEN/ISSS. 2004. Business Plan for the CEN/ISSS Workshop on Learning Technologies CEN/ISSS/WS-LT [verkkodokumentti]. [viitattu 19.2.2005]. Saatavissa: <http://www2.ni.din.de/sixcms/detail.php?id=5306>

CETIS. 2004. IMS QTI Examples [verkkodokumentti]. [viitattu 15.4.2005]. Saatavissa: <http://130.159.236.5/qthtml/cetisexamples3/examples.htm> tai <http://ford.ces.strath.ac.uk/QTI/>

Cisco. 2000. Reusable Learning Object Strategy. Definition, Creation Process, and Guidelines for Building [verkkodokumentti]. Cisco Systems, Inc. [viitattu 15.1.2005]. Saatavissa: http://www.reusablelearning.org/Docs/Cisco_rlo_roi_v3-1.pdf

Cisco. 2003. Reusable Learning Object Strategy: Designing and Developing Learning Objects for Multiple Learning Approaches [verkkodokumentti]. Cisco Systems, Inc. [viitattu 15.1.2005]. Saatavissa: http://business.cisco.com/servletw13/FileDownloader/iqprd/104108/104108_kbns.pdf

CMI001. 2004. Document No. CMI001 – CMI Guidelines for Interoperability [verkkodokumentti]. Aviation Industry CBT Committeeen spesifikaatioita [viitattu 3.2.2005]. Saatavissa: <http://www.aicc.org/docs/tech/cmi001v4.pdf>

Discendum. 2005. Optima-oppimisympäristön esittely Discendum Oy:n verkkosivuilla [viitattu 27.6.2005]. Saatavissa: <http://www.discendum.com>

Duval, E. 2004. Learning Technology Standardization: Making Sense of it All. ComSIS Vol. 1, No. 1, February 2004. Saatavissa: <http://www.comsis.fon.bg.ac.yu/ComSISpdf/Volume01/InvitedPapers/ErikDuval.pdf>

Duval, E. & Hodgins, W. 2003. A LOM Research Agenda [verkkodokumentti]. The Twelfth International World Wide Web Conference 20-24 May 2003, konferenssijulkaisuja [viitattu 4.2.2004]. Saatavissa: <http://www2003.org/cdrom/papers/alternate/P659/p659-duval.html.html>

Duval, E. et al. 2002. Metadata Principles and Practicalities [verkkodokumentti]. D-Lib Magazine April 2002, Volume 8 Number 4 [viitattu 21.6.2005]. Saatavissa: <http://www.dlib.org/dlib/april02/weibel/04weibel.html>

Fallon, C. & Brown, S. 2003. e-Learning Standards. A guide to Purchasing, Developing, and Deploying Standards-Conformant e-Learning. Boca Raton, St. Lucie Press.

Greenberg, L. 2002. LMS and LCMS: What's the Difference? [verkkodokumentti]. [viitattu 12.5.2005]. Saatavissa: <http://www.learningcircuits.org/2002/dec2002/greenberg.htm>

Heery, R. & Patel, M. 2000. Application profiles: mixing and matching metadata schemas [verkkodokumentti]. Ariadne Issue 25 [viitattu 21.6.2005]. Saatavissa: <http://www.ariadne.ac.uk/issue25/app-profiles/>

Hodgins, W. & Conner, M. 2000. Everything you ever wanted to know about learning standards but were afraid to ask [verkkodokumentti]. LiNE Zine Fall 2000 [viitattu 25.1.2005]. Saatavissa: <http://www.linezine.com/2.1/features/whewywtkls.htm>

IEEE LOM. 2002a. Draft Standard for Learning Object Metadata [verkkodokumentti]. IEEE Learning Technology Standards Committee [viitattu 6.1.2005]. Saatavissa: http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf

IEEE LOM. 2002b. Oppisisällön metatieto (LOM) [verkkodokumentti]. IEEE LTSC:n LOM-standardin suomennos, TIEKE/Jukka Korpela [viitattu 20.6.2005]. Saatavissa: http://www.tieke.fi/mp/db/file_library/x/IMG/12146/file/LOM_6.4_fi.rtf

IEEE LTSC. 2004. IEEE Learning Technology Standards Committeeen verkkosivut [viitattu 28.1.2005]. Saatavissa: <http://ltsc.ieee.org>

Ilomäki, L. 2004. Oppimisaihiot opetuksen ja oppimisen tukena. Teoksessa: Ilomäki, L. (toim.) Opi ja onnistu verkossa – aihiot avuksi. Helsinki, Opetushallitus.

IMS. 2005. IMS Global Learning Consortiumin verkkosivut [viitattu 20.1.2005]. Saatavissa: <http://www.imsglobal.org>

IMS CP. 2005. IMS Content Packaging Specification [verkkodokumentti]. IMS Global Learning Consortium [viitattu 26.4.2005]. <http://www.imsglobal.org/content/packaging/index.html>

IMS QTI. 2005. IMS Question & Test Interoperability Specification [verkkodokumentti]. IMS Global Learning Consortium [viitattu 29.3.2005]. Saatavissa: <http://www.imsglobal.org/question/index.html>

ISO/IEC. 2005. ISO/IEC JTC1 SC36:n verkkosivut [viitattu 28.1.2005]. Saatavissa: <http://jtc1sc36.org>

Jaakkola, T. et al. 2004. Erilaiset oppimisaihiot osana joustavaa kokonaisuutta. Teoksessa: Ilomäki, L. (toim.) Opi ja onnistu verkossa – aihiot avuksi. Helsinki, Opetushallitus.

Jaakkola, T. & Nirhamo, L. 2003. Who forgot the learner? [verkkodokumentti]. Turun yliopisto [viitattu 26.5.2005]. Saatavissa: http://users.utu.fi/lasnir/docs/Who_Forgot_the_Learner_JAAKKOLA_&_NIRHAMO.doc

Jacobsen, P. 2002. Reusable Learning Objects – What does the future hold? [verkkodokumentti]. Artikkelit e-Learning Magazinessa 1.11.2001 [viitattu 14.1.2005]. Saatavissa: <http://www.mcli.dist.maricopa.edu/ocotillo/retreat02/docs/rlos.pdf>

L'Allier, J. J. 1997. NETg's Map to Its Products, Their Structures and Core Beliefs [verkkodokumentti]. Vaatii rekisteröitymisen [viitattu 22.1.2005]. Saatavissa: <http://www.netg.com/DemoDownloads/>

Leeuwe, M. de. 2001. White paper on optimal development structure of re-usable building blocks for training modules based on exemplary based learning [verkkodokumentti]. Euroopan komission Information Society Technologies -ohjelman CODEX-IP-projektin julkaisuja [viitattu 8.1.2005]. Saatavissa: <http://www.fontysinteractive.nl/vooru/PDF/CODEX-IP.pdf>

Mackenzie, G. 2004. SCORM 2004 Primer. A (Mostly) Painless Introduction to SCORM [verkkodokumentti]. McGill Digital Solutions Inc. [viitattu 28.7.2005]. Saatavissa: http://www.mcgill.com/media/SCORM_2004_Primer_v1_McGill_Digital_Solutions_Gord_Mackenzie.pdf

Manninen, J. 2000. Kurssikoulutuksesta oppimisympäristöihin. Aikuiskoulutuskäytäntöjen kehityslinjoja. Teoksessa: Matikainen, J. & Manninen, J. (toim.) Aikuiskoulutus verkossa. Verkko-pohjaisten oppimisympäristöjen teoriaa ja käytäntöä. Helsingin yliopiston Lahden tutkimus- ja koulutuskeskus.

Masie. 2003. Making Sense of Learning Specifications & Standards: A Decision Maker's Guide to their Adoption, 2nd Edition [verkkodokumentti]. The MASIE Center [viitattu 8.1.2005]. Saatavissa: http://www.masie.com/standards/s3_2nd_edition.pdf

Nirhamo, L. 2002. Metatiedon käyttäminen. Selvitys metatiedon kuvaus- ja käyttötavoista sekä suositus niiden soveltamisesta Suomessa [verkkodokumentti]. Turun yliopiston Opetusteknologiayksikkö [viitattu 20.6.2005]. Saatavissa: http://users.utu.fi/lasnir/docs/Metatiedon_kayttaminen.doc

Nirhamo, L. 2004. Metatieto löytämisen apuna [verkkodokumentti]. [viitattu 20.6.2005]. Saatavissa: http://users.utu.fi/lasnir/docs/Metatieto_loytamisen_apuna.doc

Ostyn, C. 2000. Best location for metadata in a Single SCO content [verkkokeskustelu]. Keskustelufoorumi ADL:n verkkosivuilla [viitattu 29.6.2005]. Saatavissa:

<http://www.adlnet.org/forums/messageview.cfm?catid=14&threadid=1215&enterthread=y>

Ostyn, C. 2004. In the Eye of the SCORM. An introduction to SCORM 2004 for Content Developers [verkkodokumentti]. [viitattu 28.7.2005]. Saatavissa:

http://ostyn.com/standards/docs/Eye_Of_The_SCORM_draft.pdf

Peace River. 2004. Elearning Standards – Why Standards? [verkkodokumentti]. Resource Pool - projektin Wiki-sanakirja [viitattu 25.1.2005]. Saatavissa: <http://careo.prn.bc.ca/cgi-bin/wiki.pl?ElearningStandards/WhyStandards>

Polsani, P. 2003. Use and Abuse of Reusable Learning Objects [verkkodokumentti]. [viitattu 24.1.2005]. Saatavissa: <http://jodi.ecs.soton.ac.uk/Articles/v03/i04/Polsani/>

Reusable Learning. 2004a. Standards Primer [verkkodokumentti]. Reusable Learningin verkkosivusto [viitattu 25.1.2005]. Saatavissa: <http://www.reusablelearning.org/index.asp?id=77>

Reusable Learning. 2004b. SCORM Primer [verkkodokumentti]. Reusable Learningin verkkosivusto [viitattu 27.7.2005]. Saatavissa: <http://www.reusablelearning.org/index.asp?id=78>

Saarinen, J. (toim.). 2002. Kouluttajana verkossa – menetelmät ja tekniikat. Hämeen ammattikorkeakoulun julkaisu C: 2/2002.

SCORM CAM. 2001. The SCORM Content Aggregation Model. Sharable Content Object Reference Model (SCORM) Version 1.2. Advanced Distributed Learning Initiative [viitattu 29.6.2005]. Saatavissa: <http://www.adlnet.org/scorm/index.cfm>

SCORM RTE. 2001. The SCORM Run-Time Environment. Sharable Content Object Reference Model (SCORM) Version 1.2. Advanced Distributed Learning Initiative [viitattu 29.6.2005]. Saatavissa: <http://www.adlnet.org/scorm/index.cfm>

SCORM SN. 2004. SCORM Sharable Content Object Reference Model. SCORM Sequencing and Navigation Version 1.3.1. Advanced Distributed Learning Initiative [viitattu 28.7.2005]. Saatavissa: <http://www.adlnet.org/scorm/index.cfm>

Silander, P. 2003. Oppimisaihiot. Teoksessa: Silander, P. & Koli, H. Verkko-opetuksen työkalupakki – oppimisaihiosta oppimisprosessiin. Helsinki, Oy Finn Lectura Ab.

Silander, P. 2004. Oppimisaihioilla laatua verkko-opetukseen. VirtuaaliAMK-seminaarin luentomateriaalia, Mikkeli 11.11.2004.

Wiley, D. 2000. Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy [verkkodokumentti]. Osa teoksesta The Instructional Use of Learning Objects [viitattu 14.1.2005]. Saatavissa: <http://reusability.org/read/chapters/wiley.doc>