

Metadata älykkäässä oppimisympäristössä

Jarkko Suhonen

18.12.2000

Joensuun yliopisto
Tietojenkäsittelytiede
Pro gradu -tutkielma

Tiivistelmä

Metadataa ja oppimisolioita voidaan hyödyntää toteutettaessa heterogeenisen opiskelijajoukon opetusta virtuaalisissa oppimisympäristöissä. Niiden avulla opetuksessa käytettävä oppimateriaali on mahdollista jakaa sopiviin, loogisiin kokonaisuuksiin, joiden ominaisuuksia pystytään soveltamaan rakennettaessa älykkäästi toimivia, opiskelijan persoonallisiin ominaisuuksiin reagoivia oppimisympäristöjä. Lisäksi oppimisolioiden periaate tukee luonnollisella tavalla muun muassa uudelleenkäytettävyyttä, jonka avulla voidaan vähentää oppimateriaalien valmistukseen liittyviä tuotantokustannuksia. Opetuksen apuna käytettävien oppimisolioiden ja niiden välisten suhteiden kuvaamiseen käytettävistä metadastandardeista voidaan mainita esimerkkinä Dublin Core.

Esipuhe

Miten tutkielmani sitten loppujen lopuksi syntyi? Ohjaajani Erkki Sutisen kanssa sovimme toukokuussa 2000 alustavasti aiheen, joka jäi sitten reilun kuukauden ajaksi hautumaan ajatuksiini. Kesäkuussa tarkensimme aiheen koskemaan nimenomaan metadatan hyödyntämistä älykkäissä oppimisympäristöissä, koska katsoimme niillä olevan suurta merkitystä tulevaisuudessa. Kesä- ja heinäkuu kuluivat pitkälti etsiessä sopivia lähteitä. Tutkimukseni luonteesta johtuen osa lähdemateriaalista on pelkästään sähköisessä muodossa, esimerkiksi Dublin Coren virallinen World Wide Web- sivusto tuli tutuksi kesän aikana.

Heinäkuun loppu, koko elokuu ja syyskuun alku sujuivat sitten lähdemateriaaliin tutustumisen ja muistiinpanojen tekemisen parissa. Varsinainen kirjoitusurakka alkoi varsinaisesti syyskuun loppupuolella, joten siihen kului aikaa suurin piirtein kaksi kuukautta. Kaiken kaikkiaan tutkielman tekoon voidaan katsoa kuluneen noin kuusi kuukautta, josta viimeiset kolme kuukautta olivatkin sitten työn täyteisiä.

Haluaisin mainita pari henkilöä, jotka vaikuttivat eniten tämän tutkielman valmistumiseen. Sadun ahkera osallistuminen tutkielman kielellisen ulkoasun muokkaamiseen ja hänen kriittinen suhtautumisena kaikkeen kirjoittamaani tekstiin helpotti minun työtäni huomattavasti. Myös Erkki Sutisen panos tutkielmani valmistumisessa oli merkittävä, koska hänen ansioitaan tiesin mitä polkua piti edetä lopputulokseen pääsemiseksi. Suuret kiitokset heille molemmille!

Sisällysluettelo

1. JOHDANTO	1
2. TIETOVERKKO AVOIMENA OPPIMISYMPÄRISTÖNÄ	3
2.1 OPPIMISYMPÄRISTÖ.....	3
2.1.1 Määritelmä.....	3
2.1.2 Avoin, suljettu ja joustava oppimisympäristö.....	4
2.1.3 Teknologian merkitys oppimisympäristössä.....	6
2.2 TIETOVERKOSSA TOIMIVA OPPIMISYMPÄRISTÖ.....	7
2.2.1 Määritelmä.....	8
2.2.2 Opiskeluvälineet.....	8
2.2.3 Oppilaat.....	10
2.2.4 Ohjaus.....	12
2.2.5 Oppimateriaali ja käyttöliittymä.....	13
2.3 TIETOVERKKOPOHJAISEN OPPIMISYMPÄRISTÖN TOIMINTAVAATIMUKSET.....	17
2.3.1 Opetuksen järjestäminen.....	17
2.3.2 Kanssakäyminen eri tahojen kesken.....	18
2.3.3 Materiaali opiskelun tukena.....	19
2.3.4 Oppijakeskeisyys.....	20
2.3.5 Oppimista tukevat välineet.....	21
2.4 VERKKO-OPISKELUN ONGELMAT.....	24
2.4.1 Opiskelijan asema.....	24
2.4.2 Ongelmat käytettävässä teknologiassa.....	27
3. METADATA JA OPPIMISOLIOT	29
3.1 ONTOLOGIA.....	29
3.2 MITÄ METADATA ON?.....	30
3.2.1 Määritelmä.....	31
3.2.2 Metadata materiaalien kuvaamisessa.....	31
3.2.3 Metadataelementit.....	32
3.2.4 Miksi metadatatalla on merkitystä tietoverkoissa?.....	33
3.3 METADATAN KÄYTTÄMINEN OPISKELUMATERIAALIEN KUVAAMISEEN.....	33
3.3.1 Metadataskeemat.....	34
3.3.2 Opetuksellisesti käyttökelpoisen metadatan vaatimukset.....	35

3.3.3	<i>Erilaiset metadastandardit</i>	36
3.3.4	<i>Standardien vertailua</i>	43
3.3.5	<i>Keinot metadatan toteuttamiseksi</i>	45
3.4	OPPIMISOLIOT.....	51
3.4.1	<i>Määritelmä</i>	51
3.4.2	<i>Oppimisolioiden käyttö</i>	52
3.4.3	<i>Metadatan ja oppimisolioiden suhde</i>	54
4.	METADATA JA OPPIMISOLIOT ÄLYKKÄÄSSÄ OPPIMISYMPÄRISTÖSSÄ ...	56
4.1	ÄLYKÄS OPPIMISYMPÄRISTÖ	56
4.1.1	<i>Mikä on älykäs oppimisympäristö?</i>	56
4.1.2	<i>Syyt mielenkiintoon älykkäitä oppimisympäristöjä kohtaan</i>	58
4.1.3	<i>Älykkään oppimisympäristön ominaisuudet</i>	59
4.1.5	<i>Ympäristön rakentamisen ongelmat ja ratkaisuja ongelmiin</i>	62
4.2	ÄLYKKÄÄN OPPIMISYMPÄRISTÖN TOTEUTTAMINEN	64
4.2.1	<i>Ontologiapohjainen arkkitehtuuri opetettavan tiedon esittämiseksi</i>	64
4.2.2	<i>Opiskelijatiedot ja -malli</i>	66
4.2.3	<i>Älykkäät agentit</i>	69
4.2.4	<i>Materiaalin esittäminen ja käyttöliittymän merkitys</i>	73
4.2.5	<i>Ongelmanratkaisua tukevat välineet</i>	75
4.3	METADATA JA OPPIMISOLIOT ÄLYKKÄÄSSÄ OPPIMISYMPÄRISTÖSSÄ	76
4.3.1	<i>Oppimisolioiden merkitys älykkäässä ympäristössä</i>	76
4.3.2	<i>Metadata oppimisolioiden kuvaamisessa älykkäissä ympäristössä</i>	78
4.3.3	<i>Oppimisolioiden ja metadatan soveltaminen</i>	79
5.	TULEVAISUUDEN NÄKYMÄT	84
5.1	METADATAELEMENTTIEN SISÄLLÖN MÄÄRITTELY	84
5.2	METADATA OPPIMISYMPÄRISTÖN ERI TEKIJÖIDEN KANNALTA KATSOTTUNA	85
5.3	TULEVAISUUDEN OPPIMATERIAALIT	86
6.	YHTEENVETO	88
	LÄHDELUETTELO	89
	Liite1. Dublin Core Metadata Element Set, Version1.1: Reference Description	
	Liite2. IMS Core metadataelementit	

1. JOHDANTO

Tietoverkkojen käytön yleistymisen kaikilla elämäntiloilla on osaltaan vaikuttanut siihen, että opetusta tarjoavat oppilaitokset ovat kiinnostuneet järjestämään opetusta myös verkkojen välityksellä. Oppilaat on varsin usein jätetty pärjäämään yksin verkosta löytyvän oppimateriaalin ja oppimistehtävien kanssa. Kuitenkin oppilaat ovat yksilöitä tietoineen, taitoineen ja tarpeineen ja näin ollen tarvitsevat henkilökohtaista ohjausta. Tämä johtaa siihen, että opiskelun kannalta tärkeään asemaan nousevan oppimisympäristön pitää pystyä tarjoamaan keinoja kohdata erilaisten oppilaiden vaatimuksia ja tarpeita joustavalla tavalla.

Eräs mielenkiintoisimmista tavoista toteuttaa kyseinen tarve on käyttää niin sanottuja älykkäitä oppimisympäristöjä, jotka erilaisia välineitä käyttäen pystyvät vastaamaan homogeenisten oppilasjoukkojen opetukselle asettamaan haasteeseen. Millä tavalla näitä älykkäitä toimivia oppimisympäristöjä voidaan sitten toteuttaa? Vastausta lähdin etsimään metadatan ja oppimisolioiden avulla ja päädyn siihen tulokseen, että ne voivat osaltaan tarjota ratkaisuja älykkäiden oppimisympäristöjen rakentamisen ongelmaan.

Aloitin tutkimukseni tutustumalla luvussa kaksi tietoverkkopohjaiseen avoimeen oppimisympäristöön, jossa opiskelu on toteutettu varsin usein World Wide Webin avulla. Luvun tarkoituksena on antaa teoreettista pohjaa seuraavia lukuja varten siten, että luvussa selvitetään mikä on tietoverkkojen merkitys opettamisen ja oppimisen kannalta katsottuna. Lähteet lukuun kaksi sain osittain aikaisemmin tekemiäni tutkielmien pohjalta. Esimerkiksi opettajaseminaaritutkielmani toinen luku käsitteli osittain samoja teemoja kuin tämän tutkielman samainen luku, joten katsoin osan lähteistä sopivan myös tähän tutkielmaan varsin hyvin.

Oppimisympäristöihin tutustumisen jälkeen siirryn käsittelemään metadatan ja oppimisolioita. Luvussa kolme pyrin selvittämään, mitä kyseiset käsitteet tarkoittavat ja miten niitä voitaisiin hyödyntää esimerkiksi toteutettaessa oppimateriaaleja. Esittelen muutaman metadatastandardin, joiden avulla oppimisolioita ja niiden välisiä suhteita voidaan kuvata. Kolmannen luvun lähteistä suuri joukko on World Wide Web -sivuja ja siihen on myös varsin pätevä selitys. Kyseisiltä sivuilta löytyy muun muassa metadatastandardien määrittelyjä, joiden löytäminen tieteellisistä artikkeleista olisi ollut suuren työn takana. Kyseiset sivut edustavat metadatan liittyvän tiedon alkulähdettä, jolloin niihin viittaaminen on mielestäni erittäin perusteltua.

Neljännessä luvussa pyrin yhdistämään edellisten lukujen sisällön älykkään ympäristön toiminnan kannalta. Luvussa käyn läpi niitä keinoja, joilla älykkäitä ympäristöjä voidaan toteuttaa. Luvun lopussa pohdin sitä, miten metadata ja oppimisoliot voisivat olla apuna toteutettaessa älykkäästi toimivia oppimisympäristöjä. Tämän luvun lähteet perustuvat osittain Erkki Sutisen tarjoamiin lähdeoteuksiin ja ison roolin ottavat Peter Brusilovskyn ajatukset älykkäästi toimivista oppimisympäristöistä. Mielestäni hänellä on varsin hyviä ja selkeitä ajatuksia siitä, millainen tulevaisuuden älykkäästi toimivan oppimisympäristön pitäisi olla ja miten kyseinen ympäristö voidaan toteuttaa.

Viidennen luvun tarkoituksena on selvittää niitä jatkotutkimuksen kohteita, joita tämän tutkimuksen perusteella nousee esiin. Lisäksi pyrin tuomaan esille omia mielikuviani tulevaisuuden oppimisympäristöstä, joka voidaan toteuttaa osittain metadatan ja oppimisolioiden avulla.

Kaikissa tutkielman luvuissa tulen viittaamaan virtuaaliapprobatur-opintoihin, jolla tarkoitan keväällä 2000 Joensuun yliopiston tietojenkäsittelytieteen laitoksella alkanutta projektia. Projektin tarkoituksena on tarjota Pohjois-Karjalan maakunnan lukiolaisille 15 opintoviikon tietojenkäsittelytieteen approbatur-opintoja opiskeltavaksi verkon kautta. Koska virtuaaliapprobatur-opintoja on tarkoitus kehittää älykkäiden periaatteiden mukaisesti, niin metadata ja oppimisoliot nähtiin eräänä mahdollisena tapana kehittää kyseisiä opintoja. Seurauksena tästä kehittyi ajatus aiheeksi Pro gradu -työlleni, johon lukija voi tutustua tarkemmin seuraavissa luvuissa.

2. TIETOVERKKO AVOIMENA OPPIMISYMPÄRISTÖNÄ

Opiskelun ja oppimisen katsotaan tämän hetkisten oppimisteorioiden mukaan tapahtuvan aina oppimisympäristössä, joten katson aiheelliseksi aloittaa tutkielmani tutustumalla oppimisympäristön käsitteeseen (Levi 1994, Wilson 1995). Määrittelen aluksi avoimen oppimisympäristön, jonka pohjalta laajennan käsittelyn koskemaan tietoverkossa toimivaa oppimisympäristöä sekä siihen liittyviä erityisiä vaatimuksia ja ongelmia.

2.1 Oppimisympäristö

Oppimisympäristö voi olla erittäin merkittävässä asemassa onnistuneessa oppimisessa. Etenkin tietokoneiden ja -verkkojen avulla tapahtuvassa opiskelussa oppimisympäristö voi olla ratkaisevassa asemassa oppimistavoitteiden saavuttamista ajatellen. Oppiminen tapahtuu aina tietyssä ympäristössä, jolloin ideaalisessa tilanteessa oppimisympäristö antaa oppijalle lisämotivaatiota ja innostusta opiskeluun.

2.1.1 Määritelmä

Määrittelemme *oppimisympäristön* Auerin ja Pohjolaisen (1995) mukaan toimintaympäristöksi, joka sisältää oppiaineksen lisäksi myös oppijat, kouluttajat, oppimisenäkemykset, toimintamuodot, oppimislähteet, tekniikat ja mediat. Oppimisympäristön käsite on siis paljon laajempi kuin perinteinen käsitys opettajasta, oppilaasta ja oppimateriaalista. Tästä johtuen opetus- ja oppimistilanteeseen katsotaan vaikuttavan usean eritasoisen tekijän hyvin eri tavoilla.

Wilson (1995) jatkaa oppimisympäristön määrittelyä toteamalla todellisen oppimisympäristön olevan konstruktivismiin pohjautuva järjestelmä, jossa oppijat voivat työskennellä yhdessä ja tukea toisiaan erilaisin tavoin. Oppijoilla on riittävästi välineitä tiedon muokkaamiseen ja omien konstruktioiden luomiseen. Hän laajentaa oppimisympäristön koskemaan myös vallitsevaa kulttuuria ja yhteiskuntaa, jolloin niilläkin katsotaan olevan oma vaikutuksensa oppimisympäristöjen muodostumiseen. Esimerkiksi tietoyhteiskunta on tuonut vahvasti tietokoneet mukanaan myös opetuksen piiriin ja sitä kautta oppimisympäristöihin.

Edellä olevista määritelmistä voimme päätellä oppimisympäristöjen olevan kaikessa oppimisessa mukana oleva taho, joka on erittäin ratkaisevassa asemassa oppimisen kannalta. Oppimisympäristö antaa puitteet mielekkäälle ja onnistuneelle oppimiselle, joten sen toteuttaminen oikean tyyppisellä tavalla on erittäin tärkeää.

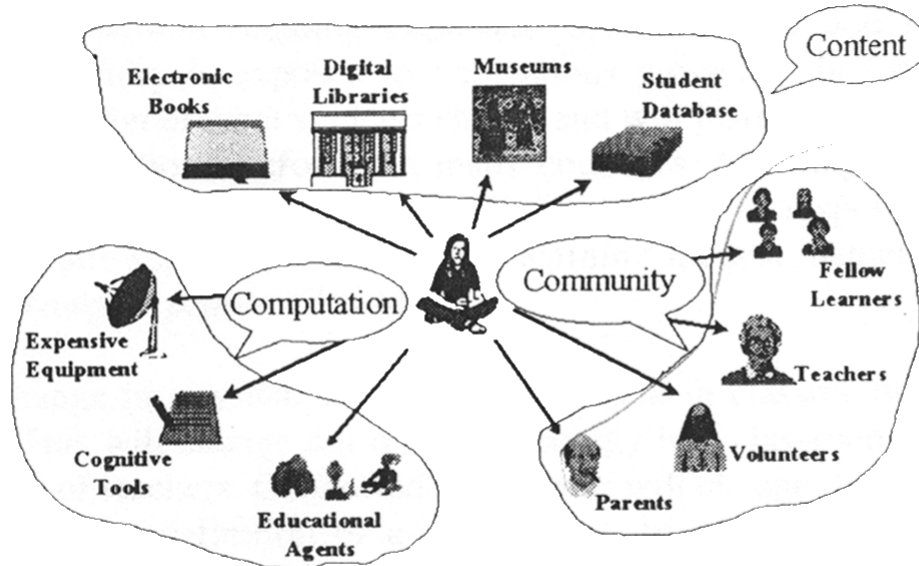
2.1.2 Avoin, suljettu ja joustava oppimisympäristö

Kuten edellä olevasta määritelmästä voidaan päätellä, niin oppimisympäristö on tila tai prosessi, johon vaikuttavat hyvin monet tekijät. Minkälainen on sitten ominaisuuksiltaan ihanteellinen oppimisympäristö, jossa kaikkien yllämainittujen tekijöiden vaikutuksesta oppimista tapahtuu?

Perinteisen opetuksen on yleensä katsottu tapahtuvan *suljetussa oppimisympäristössä*, jossa opettaja on yksin päättänyt esimerkiksi käytetyistä välineistä, oppimateriaaleista, aikatauluista ja etenemistahdista. Wilson (1995) kuvaa suljettua oppimisympäristöä tilaksi tai rakennelmaksi, jossa oppilaat toimivat käyttäen välineitä ja laitteita informaation keräämiseen ja tulkitsemiseen. Hän kokee suljetun oppimisympäristön rajoittavaksi ja jäykäksi, jossa opiskelijat eivät pääse toteuttamaan itseään riittävästi. Suljetut ympäristöt ovat vielä hyvin yleisiä erityyppisessä opiskelussa ja tietyissä tilanteissa suljetun ympäristön turvallinen ilmapiiri voi auttaa oppimisprosessissa, joten se ei kaikissa tilanteissa ole suinkaan huonoin ratkaisu.

Suljetun oppimisympäristön tilalle Miller (1997) ja Levi (1994) esittävät *avointa oppimisympäristöä*, joka antaa mahdollisuuden olla joustavampi ja vastaanottavampi sekä oppijoiden että teknologian mahdollisuuksille ja vaatimuksille. Heidän mukaansa avoimen oppimisympäristön tunnusmerkkejä ovat muun muassa: oppijakeskeisyys, rikas kommunikointi, tutkivan oppimisen hyödyntäminen, oppimistavoitteiden asettelu ja oppijoiden mahdollisuus tehdä päätöksiä sekä yksin että ryhmässä erilaisissa ongelmatilanteissa. Avoimessa oppimisympäristössä oppimista ohjataan ja tuetaan, mutta sitä ei kontrolloida tai pakoteta tiettyyn malliin (Wilson 1995). Avoimen oppimisympäristön puolesta puhujien mukaan oppiminen lähtee jokaisen ihmisen omista haluista ja motivaatioista, jolloin mahdollisimman moneen opiskelua koskevaan päätökseen pitäisi oppijan päästä vaikuttamaan ainakin jollain tasolla.

Kuvassa 1 on Chanin et al. (1999) käsitys oppijakeskeiseen avoimeen oppimisympäristöön kuuluvista eri tekijöistä, joiden avulla edellä mainitut avoimen oppimisympäristön ominaispiirteet saadaan toteutettua.



Kuva 1: Oppijakeskeinen avoin oppimisympäristö (Chan et al. 1999).

Avoimen oppimisympäristön voidaan ajatella olevan kuin lapsuuden elinympäristö, jossa lapsi tietää, missä kaikki ovat, mistä löytää erilaisia asioita, keneltä voi kysyä ja minne mennä. Ympäristö on samanaikaisesti sekä turvallinen että lasten erilaiset tarpeet huomioonottava. Työskentely ja ongelmien ratkaisu avoimessa oppimisympäristössä on usein seurausta samanlaisista tiedoista ja taidoista (Wilson 1995).

Avoimen oppimisympäristön rinnalla käytetään usein hieman erilaisia samaa tarkoittavia termejä, esimerkiksi joustavaa oppimisympäristöä tai uutta oppimisympäristöä. Näillä hieman toisistaan poikkeavilla käsityksillä tarkoitetaan edellä määritellyn avoimen oppimisympäristön kaltaisia järjestelmiä. Avoin oppimisympäristö pystyy tietyissä tilanteissa tarjoamaan suljettua ympäristöä paremmat mahdollisuudet oppilaiden erilaisten tarpeiden huomioonottamiseksi. Tosin termi ”avoin” ei mielestäni kuvaa riittäväällä tarkkuudella ideaalista oppimisympäristöä, sillä ihanteellinen oppimisympäristö tarvittaessa muuntuu joko avoimeksi tai sulje-

tuksi. Tästä johtuen *joustava oppimisympäristö* kuvaa mielestäni paremmin edellä esitettyjä piirteitä hyvästä oppimisympäristöstä.

2.1.3 Teknologian merkitys oppimisympäristössä

Uuden teknologian käyttö opetuksessa luo helposti mielikuvan siitä, että pelkästään tietoverkot tai tietokoneet itsessään tekevät oppimisympäristöstä avoimen tai joustavan. Facer et al. (2000) toteavatkin, että alun alkaen tietokoneita käytettiin opetuksen apuna suljetuissa, tiukasti johdetuissa ympäristöissä, jotka eivät antaneet paljon mahdollisuuksia joustavaan opiskeluun.

Levi (1994) määrittelee *opetusteknologian* monimutkaiseksi prosessiksi, johon kuuluvat ihmiset, menetelmät, ideat, laitteet ja organisaatio. Näiden tekijöiden avulla pyritään analysoimaan, arvioimaan ja selvittämään ongelmia, jotka liittyvät ihmisen oppimiseen. Ihmiset saattavat esimerkiksi joutua käsittelemään oppimisympäristössä uusia menetelmiä ja laitteita, joiden käyttöönotto ei ole välttämättä yksinkertainen asia. Opetusteknologia pyrkii löytämään ratkaisuja siihen, miten teknologia saadaan mahdollisimman toimivaksi ja hyödylliseksi osaksi oppimisympäristöä. Teknologiaa ei pidä siis nähdä erillisenä osana oppimisympäristöä ja oppimista, vaan se tarjoaa erilaisia ratkaisuja oppimisympäristön eri tekijöiden toteuttamiseksi.

Miller (1997) korostaa, että uusi teknologia luo mahdollisuuden tehdä asioita uudella tavalla, mutta pelkkä teknologian käyttö ei tuo ratkaisua opetuksellisiin ongelmiin. Itse asiassa uuden teknologian käytöstä ei ole suurtakaan hyötyä, jos opetusmenetelmät ja -periaatteet eivät kehity samanaikaisesti. Sokea usko pelkästään tekniikan kaiken parantavaan voimaan ilman uusia opetusmenetelmiä ei tunnu järkevältä, koska vielä ei tiedetä varmuudella teknologian todellista vaikutusta oppimisprosessiin. Oppilaitokset ovat usein hankkineet tietokoneita, laitteita ja verkkoja, mutta niiden todellista merkitystä oppimiseen ei ole ajateltu riittävästi. Laitteita on mitä erilaisimpien tehtävien suorittamiseksi, mutta niitä ei osata käyttää oikealla tavalla. Haran ja Klingin (2000) mielestä tietokoneiden opetuskäytössä korostetaan yleensä pelkästään positiivisia asioita ja väheksytään mahdollisia vaikeuksia, koska halutaan pysyä kehityksen mukana kaikin keinoin.

Levi (1994) ja Lloyd (2000) näkevät teknologian tärkeässä osassa suunniteltaessa ja toteutettaessa edellisessä kohdassa esitettyyn avoimeen oppimisympäristöön pohjautuvia järjestelmiä. Teknologian avulla pystytään esimerkiksi laajentamaan oppimisympäristöä luokkahuoneesta melkein koko maailman kattavaksi, jolloin oppijan ei tarvitse tyytyä tiettyihin tiedonlähteisiin, vaan hän voi käyttää useita erilaisia tiedon muotoja. Miller (1997) toteaa tietokoneiden muutaneen koko yhteiskuntaa, niinpä myös opetuksessa pitäisi vastata tietoyhteiskunnan haasteisiin. Luonnollinen tapa on käyttää teknologiaa monipuolisesti, jotta oppilaat harjaantuisivat muun muassa tiedon etsinnässä, arvioimisessa ja analysoinnissa (Maureen 2000).

Maureen (2000) haluaisi mennä vielä pidemmälle ja hän näkeekin teknologian avulla olevan mahdollista siirtyä pelkästä tiedon hankintaprosessista tiedon konstruointiprosessiin. Konstruointiprosessissa tietoa muokataan omien näkökulmien mukaisesti ja samalla tietokoneen avulla voidaan esimerkiksi simulaatioiden toimesta tarjota virtuaalisia kokemuksia, joita ei normaalissa ympäristössä pääse välttämättä kokemaan.

Seuraavissa luvuissa siirryn käsittelemään pelkästään tietoverkossa toimivien oppimisympäristöjen piirteitä, sillä niillä on keskeinen osa tutkimuksen viitekehyksessä. Teknologialla on tietoverkkopohjaisissa oppimisympäristöissä luonnollisesti erittäin suuri merkitys, joten siihen voidaan suhtautua hieman eri tavalla kuin reaali maailman oppimisympäristöissä. Käyttäessämme tietoverkkopohjaisia oppimisympäristöjä olemme tietoisesti hyväksyneet teknologian käyttökelpoisuuden oppimisessa ja opiskelussa. Jäljelle jää ainoastaan kysymys siitä, millä tavoin voimme maksimoida tietoverkkojen ja tietokoneiden tarjoamat mahdollisuudet ja haasteet opetuksessa. Tästä eteenpäin puhuessani oppimisympäristöistä tarkoitan edellä määritellyn avoimen ja joustavan oppimisympäristön toteuttamista tietoverkkojen avulla.

2.2 Tietoverkossa toimiva oppimisympäristö

Edellisessä luvussa käsitelin yleisesti oppimisympäristöjen käsitettä, jonka pohjalta ryhdyn selvittämään muun muassa useiden yliopistojen tietoverkkopohjaisten, etäopetukseen tarkoitettujen oppimisympäristöjen erikoispiirteitä.

2.2.1 Määritelmä

Miller (1997) määrittelee World Wide Webiä, sähköpostia ja muita Internetin palveluita hyväksi käytävän oppimisympäristön *virtuaaliseksi oppimisympäristöksi*, johon opiskelijat pystyvät tietoverkkojen välityksellä ottamaan yhteyttä kaikkialta maailmasta. Miller (1997) ja Bork (1997) näkevät tietoverkkojen avulla tapahtuvan opetuksen (distance education) olevan jo nyt merkittävässä asemassa oppilaitosten koulutustarjonnassa. Verkossa tapahtuva opiskelu mahdollistaa monipuolista oppilaskuntaa kiinnostavan tavan tarjota opetusta 24 tuntia vuorokaudessa seitsemänä päivänä viikossa, joten oppilaitoksen mielenkiinto tietoverkossa tapahtuvaa opiskelua kohtaa ei ole yllättävää (Hara ja Kling 2000).

Vaikka sana ”virtuaalinen” ei ole kaikista parhain termi kuvaamaan tietoverkkojen välityksellä tapahtuvaa opetusta, silti siitä on tullut varsin yleinen puhuttaessa tietoverkoissa toimivista oppimisympäristöistä. Tomak (1999) mainitsee tietoverkossa olevan oppimisympäristön yleisimmiksi piirteiksi muun muassa mahdollisuuden päästä käsiksi monipuoliseen materiaaliin Internetin välityksellä, synkronisen ja asynkronisen kommunikoinnin, yhteistyön mahdollisuuden sekä oppimista tukevien välineiden käytön.

Tietoverkossa toimiva oppimisympäristö tuo mukanaan luonnollisella tavalla Internetin tarjoamat tiedonlähteet, joita voidaan hyödyntää muun muassa oppimistehtävissä. Monipuoliset kommunikaatiomuodot (asynkroninen ja synkroninen) ovat tärkeässä asemassa ympäristön toimivuuden kannalta, jotta tietoverkoista puuttuva ihmismäinen kanssakäyminen voitaisiin korvata mahdollisimman hyvin. Oppimista tukevilla välineillä voidaan vähentää osittain opettajan merkitystä oppimisprosessissa erityisesti vaikeiden asioiden opettamisessa. Näitä välineitä voi olla esimerkiksi animaatiot, simulaatiot sekä videot, joilla opetettavan asian ymmärtämistä voidaan syventää ja parantaa samalla oppimistuloksia.

2.2.2 Opiskeluvälineet

Lifländer (1999) määrittelee *verkkokoulutuksen* avoimessa (Internet) tai suljetussa (intranet) verkossa tapahtuvana järjestettynä opetuksena ja opiskeluna. Verkko-oppiminen on osittain verkkokoulutuksen tulosta ja osittain koulutuksesta riippumatonta, omatoimista oppimista. Verkkokoulutus tarjoaa mahdollisuuden hoitaa monimuoto-opetuksen tärkeimpiä osa-alueita

uudella tavalla, jolloin esimerkiksi seuraavat toimet voidaan suorittaa tietoverkon välityksellä: tiedotus, kurssinhallinto, materiaalin jako, oppimistehtävät ja oppimistulosten arviointi. Siis kaikki ”perinteisen” oppimisympäristön, esimerkiksi oppilaitoksen, toimet voidaan toteuttaa tietoverkkojen avulla yhdessä ja samassa oppimisympäristössä.

Tietoverkko-opetuksessa käytetään hyväksi normaaleja tietoverkkojen palveluja, kuten World Wide Webiä, sähköpostia, uutispalveluita ja postituslistoja. Opiskelu ei ole periaatteessa mihinkään paikkaan tai aikaan sidottua, vaan opiskelija voi osallistua opetukseen verkkoyhteyden avulla kaikkialta maailmasta (Miller 1997). Tämä mahdollistaa sen, että tietoverkoissa olevat oppimisympäristöt tarjoavat varsin toimivan keinon oppimateriaalien jakamiseksi suu- relle joukolle opiskelijoita. Valitettavasti tämä on johtanut siihen, että oppilaitokset tarjoavat kurseissaan pelkästään World Wide Webiin koostettuja oppimateriaaleja, joita opiskelija joutuu itsenäisesti tutkimaan (Lloyd 2000). Lifländer (1999) muistuttaa siitä, että verkko- oppimista voidaan toki hyödyntää myös lähiopetuksessa, jolloin Internet toimii muun muassa lisämateriaalin ja oppimistehtävien jakeluvälineenä.

Oppilaitoksilla on käytännössä kaksi erilaista tapaa rakentaa tietoverkossa toimiva oppi- misympäristö: joko oppilaitos rakentaa ympäristön itse tai se käyttää valmiiksi tuotettuja ympäristöjä. Jälkimmäinen ratkaisu on varsin yleinen, sillä harvassa oppilaitoksessa on resursseja ryhtyä rakentamaan omaa, teknisesti hyvinkin monimutkaista järjestelmää. Myöskään opetta- jien ei tarvitse opetella ohjelmointia, vaan he pystyvät varsin pienellä vaivalla siirtämään tar- vittavan materiaalin oppilaiden saataville (Tomak 1999).

Esimerkkeinä näistä valmiista ympäristöistä voidaan mainita virtuaaliapprobatur-opinnoissa käytettävä WebCT ja Lotus Notes -teknologiaan perustuva LotusLearningSpace, jotka ovat käytössä useissa oppilaitoksissa sekä Suomessa että ulkomailla (Tomak 1999). Kuvassa 2 on virtuaaliapprobatur-opintojen ”Tietokone, käyttöjärjestelmä ja tietoverkko”-kurssin etusivu.

Kyseiseltä etusivulta on linkit tärkeimpiin ohjelmiston toimintoihin muun muassa oppimateri- aaliin, oppimistehtäviin, keskustelufoorumiin ja kalenteriin, jotta ne olisivat oppilaiden saata- villa nopeasti ja yksinkertaisesti. WebCT-ympäristöön voi ottaa yhteyttä yleisimmillä se- laimilla eikä sen käyttöön tarvita välttämättä aivan uusinta tekniikkaa. WebCT tarjoaa siis edellä kuvailun mahdollisuuden sisällyttää kaikki opiskeluun liittyvät seikat verkossa toimi- vaan ympäristöön, joten WebCT:n avulla rakennettu ”oppimaisema” toimii tietyissä tilanteis-

sa opiskeluvälineenä. Virtuaaliapprobatur-opintojen kaikki verkkojen välityksellä tarjottavat kurssit on tarkoitus toteuttaa WebCT -ohjelmistolla, josta virtuaaliapprobatur-tiimillämme on suhteellisen positiivisia kokemuksia.



Kuva 2: Oppimisympäristö tietoverkossa.

2.2.3 Oppilaat

Tietoverkkojen välityksellä tapahtuva opiskelu vaatii opiskelijalta aktiivisempaa suhtautumista omaan oppimiseen ja opiskeluun kuin perinteisessä luokkaympäristössä tapahtuvassa, tiukasti johdetussa opetuksessa. Voidaan ajatella, että verkko-opiskelun perimmäisenä filosofiana on se, että oppilas itse on mahdollisimman paljon vastuussa omasta opiskelustaan ja oppimisestaan. Tämän takia Alpert et al. (1997) karakterisoivat verkossa tapahtuvan opiskelun aktiiviseksi oppimiseksi, jossa tarvitaan muun muassa reflektointi- ja itsetutkiskelutaitoja.

Verkko-opiskelu sisältää paljon perinteisen etäopetuksen keinoja opetuksen toteuttamiseksi, mutta Internet mahdollistaa Grangerin ja Passerinin (2000) mukaan aikaisempien etäopetusmenetelmien tueksi todellisen oppilaiden välisen kommunikaation. Koulutusta tarjoavilla or-

ganisaatioilla on kuitenkin hyvin erilaiset näkökannat, miten kyseistä ominaisuutta käytetään opiskelutilanteissa. Seuraavaksi esittelen Bostockin (1997) esiin tuomat erilaiset skenaariot verkko-opiskelusta oppilaiden kannalta katsottuna.

Ensimmäisessä ja Bostockin (1997) mielestä painajaismaisimmassa tilanteessa suuret määrät opiskelijoita osallistuu aikatauluun sidottuun luento- ja harjoitustilanteeseen, jota tarjotaan verkkojen välityksellä. Oppilaat toimivat verkotetussa ympäristössä käyden läpi luentomuistiinpanoja (materiaaleja) yksin ja yrittävät selviytyä omien kykyjensä mukaan. Kyseisessä tapauksessa opiskelusta puuttuu sosiaalinen konteksti, opiskelijat ovat yksin ja kokemus on erittäin passiivinen, rajoittunut ja oppilaiden kannalta katsottuna toisista oppilaista erkaannuttava.

Toisessa tilanteessa oppilaat käyttävät verkotetun ympäristön palveluita hyväkseen. Selaimen käyttöliittymän (vertaa kuva 2) kautta he pääsevät käsiksi kurssimateriaaliin, simulaatioihin ja videoihin. Oppilaat etsivät lisämateriaaleja ja kirjoittavat raportteja, jotka tulevat seuraavien opiskelijoiden oppimisresursseiksi. Bostockin (1997) mielestä tämän tapainen lähestymistapa on yleisin toteutettaessa verkko-opiskelua, mikä pitää paikkansa vielä nykypäivänä.

Kolmannen skenaarion Bostock (1997) näki utopiana, johon ei olla välttämättä valmiita menemään nykykulttuurissa. Siinä oppilaille on vapaa pääsy kotoaan tai koulusta oppimisympäristöön ja oppilaat toimivat joko yksin tai ryhmissä tehden projekteja, joita ei tarkasteta lainkaan. Oppilaat etsivät kaiken tiedon Internetistä, tutor-opettaja voi olla paikalla mutta hän ei ole opetettavan aiheen ekspertti. Joka tapauksessa kaikki edellä esitetyt mallit vaativat oppilalta aktiivista otetta omaan opiskeluun, jonka seurauksena oppimista tapahtuu.

Karjalaisen (1997) mukaan tyypillinen etäopiskelija on valmis kantamaan vastuun omasta oppimisestaan, sillä oppilas ja ohjaaja eivät välttämättä aina tapaa todellisessa elämässä. Itse asiassa aktiivinen ote opiskeluun on välttämätöntä, jotta opiskelu ylipääntensä onnistuisi halutulla tavalla. Oppilaalla on vastuu organisoida omaa tekemistään annettujen aikataulujen mukaan, mutta hänellä on myös mahdollisuus tutustua tarjottuihin materiaaleihin ja tehtäviin omaan tahtiinsa (Alpert et al. 1997).

Tärkeänä ominaisuutena kyseisessä tilanteessa Bostock (1997) näkee oppilaiden kyvyn käyttää monipuolisia oppimisstrategioita asetettujen tavoitteiden saavuttamiseksi. Alpert et al. (1997) kirjoittavat samasta aiheesta viitatessaan tietoverkkojen mahdollisuuteen tarjota ope-

tusta siten, että oppilaat voivat käyttää omia oppimistyylejään toimiessaan ympäristössä. Parhaimmassa tapauksessa oppilas itse asettaa tavoitteet, mutta varsin usein tavoitteet tulevat opettajalta tai ohjaajalta, mikä ei välttämättä ole kuitenkaan negatiivinen asia. Varsinkin nuoremmissa opiskelijoilla omaehtoinen tavoitteiden asettelu saattaa olla ongelmallista.

2.2.4 Ohjaus

Ohjauksella on merkittävä rooli tietoverkkoihin pohjautuvissa oppimisympäristöissä, sillä opiskelijalla on oltava mahdollisuus turvautua erilaisissa ongelmatilanteissa asiantunteisiin henkilöihin. Alpert et al. (1997) sisällyttävät ohjaajalle seuraavia tehtäviä verkko-opiskelussa; hän voi olla tiedonlähde, opas, oppimistilanteiden luoja ja oppisisältöjen suunnittelija. Ohjaaja toimii esimerkiksi oppimismenetelmiin tutustuttajana ja vähitellen, oppilaiden taitojen karttuessa, hän siirtyy taka-alalle.

Koska opiskelu tapahtuu tietokoneiden ja tietoverkkojen avulla, on erityisesti tekninen tuki erittäin ratkaisevassa asemassa. Opiskeluun vaikuttavat häiriötekijät on karsittava minimiin, jotta ne eivät pääse vaikuttamaan liikaa oppimistuloksiin ja -motivaatioon (Hoffman ja Ritchie 1997). Vaikka ohjauksen antaja on yleensä kurssista vastuussa oleva henkilö eli ohjaaja, niin ympäristössä toimivien muiden henkilöiden tuki on myös erittäin arvokasta. Tomak (1999) toteaa oppimisen tapahtuvan ideaalissa oppimisympäristössä silloin, kun oppilaiden, ohjaajien, avustajien, hallinnon ja muiden tahojen välillä on rakentavaa ja vilkasta kommunikaatiota.

Tietoverkossa toimivat oppimisympäristöt tarjoavat usein sekä synkronisia että asynkronisia kommunikaatiovälineitä, joiden avulla oppilaille pystytään tarjoamaan riittävää ohjausta. Ohjauksen avulla voidaan esimerkiksi parantaa opiskelijoiden kykyä hahmottaa olennaisimpia asioita tietoverkkojen sisältämästä valtavasta tietomäärästä, jotta opiskelijat eivät turhautuisi liiasta, usein tarpeettomastakin tiedosta.

2.2.5 Oppimateriaali ja käyttöliittymä

Luonnollisesti tietoverkossa olevan oppimisympäristön opiskelumateriaalit on toteutettu usein samalla tavoin kuin muut tietoverkosta löytyvät materiaalit. Tietoverkkojen materiaali perustuu pitkälti hypermediaan, jonka lisänä on World Wide Webin tarjoama käyttöliittymä. Nykyisin käytetyillä tekniikoilla, kuten HTML (HypertText Markup Language)-dokumenteilla, Java-appleteilla, digitaalisilla kuvilla ja animaatioilla, on tulevaisuudessakin suuri merkitys oppimisympäristöjen materiaalien tuottamisessa. Taulukossa 1 on esitelty World Wide Webin tarjoamia opetuksessa hyödynnettäviä tekniikoita.

Taulukko 1: Opetuksen apuna käytettävät WWW-tekniikat.

Tekniikka	Käyttö
HTML	Oppimateriaali, oppimisympäristöt
Java-appletti	Interaktiiviset sovelmat, animaatio
CGI (Common Gate Interface)	Palautteen antaminen oppijalle
Javascript	Tiedon testaaminen, kyselyt, dynaaminen oppimateriaali
Flash	Interaktiiviset animaatiot, visualisointi

Pantzar (1997) on löytänyt kolme erilaista ratkaisua World Wide Webin hypermedia-ominaisuuksien hyödyntämisessä oppimateriaalien rakennettaessa:

- 1) perustapa, www-pohjainen lineaarinen teksti, jossa on joko luentojen runko tai kirjoitettu esitelmä luennoista. Erona perinteiseen materiaaliin on vain materiaalin erilainen saatavuus.
- 2) sisäisesti linkitetyt tekstit, joissa on käytetty kuvia ja piirroksia opettävien asioiden havainnollistamiseksi. Hypermedia-ominaisuudet ovat käytössä, koska ei-lineaarinen navigointi on mahdollista.

3) sisäisten linkkien lisäksi on myös ulkoisia linkkejä. Muistuttaa elävää tiedon etsimistä ja tarjoaa opiskelijalle teoriassa loputtoman määrän tietolähteitä.

Brown (1997) on sitä mieltä, että yllä olevan Pantzarin esittämän jaon ensimmäisen kohdan mukaiset oppimateriaalit ovat hyvin yleisiä monissa oppimisympäristöissä. Usein perinteinen luento on siirretty sellaisenaan digitaaliseen muotoon ja annettu oppilaille. Tämän mallin mukaisesti oppilas on passiivinen ja hän seuraa pelkästään lineaarista, opettajan valmiiksi pureskelemaa oppimispolkua.

Farquhar ja Jones (1997) näkevät erittäin tärkeänä osana oppimisympäristöä sen, miten oppimateriaali on esitetty ja miten navigointi on määritelty ja toteutettu. Heidän mukaansa käyttöliittymän suunnittelu oppilaiden näkökulmasta on toivottavaa, jotta heidän ei tarvitsisi keskittyä liikaa epäolennaisiin, oppimista haittaaviin tekijöihin oppimisympäristössä.

Farquhar ja Jones (1997) antavat oppimisympäristön käyttöliittymää ja HTML:llä toteutettua materiaalia varten seuraavia ohjeita. Käyttöliittymän suunnittelun kannalta olisi toivottavaa tarjota rakenteellisia vinkkejä tiedosta (vertaa kohta 3.2, metadata) eli yleiskatsauksia, sisällysluetteloita ja indeksejä. Lisäksi esimerkiksi otsikoiden, alaotsikoiden ja selventävien esimerkkien tyylin ja asettelun tulisi olla yhtenäistä, jotta opiskelijalle muodostuu selvä kuva erilaisten tietokokonaisuuksien merkityksestä ja suhteesta. He kannustavat käyttämään yleisesti tunnettuja menetelmiä ilmoittamaan linkkien asemasta eli muun muassa vierailtujen linkkien kohdalla kannattaa käyttää tuttua punaista väriä.

Taulukossa 2 on Farquharin ja Jonesin (1997) esittämä tyyliopas HTML:n käytöstä oppimateriaalien tekemisessä. Tyyliopasta voidaan käyttää ohjenuorana suunniteltaessa verkko-opetukseen käytettäviä oppimateriaaleja ja oppimisympäristöjä käymällä läpi askel askeleelta taulukon ohjeet ja tarkastamalla toteutuvatko oppaan mainitsevat asiat omassa materiaalissa.

Taulukko 2: Farquharin ja Jonesin (1997) esittämä tyyliopas oppimateriaaleja varten.

Askel:	Neuvo:
Ohje 1:	Tarjota useita versioita materiaalista: tekstiversio, vähän grafiikkaa ja paljon grafiikkaa. Näin oppilailla on mahdollisuus valita omiin opiskeluolosuhteisiin parhaiten sopiva materiaali.
Ohje 2:	Tarjota apua selaimen asetuksissa. Esimerkiksi kerro oppilaille, mistä materiaalin tutustumiseen tarvittavat plug-init kannattaa hakea ja miten ne saadaan asennettua.
Ohje 3:	Pidä sivut lyhyenä. Käyttäjät eivät pidä liikkumisesta samalla sivulla edestakaisin eli skrollauksesta. Mitä pitempi sivu, sen kauemmin sen lataaminen kestää.
Ohje 4:	Vältä käyttämästä linkkejä sivun sisällä, koska se lisää käyttäjän epävarmuutta. Suosi sitä, että linkit osoittavat aina muuhun sivuun.
Ohje 5:	Valitse tai asettele näytettävät linkit tarkoin. Jos kaikki ovat esillä, niin mikään ei jää mieleen. Priorisoi esitettävät asiat tai ilmoita käyttäjälle, miten aihe on esitelty.
Ohje 6:	Anna linkeille niitä hyvin kuvaavat nimet. Älä käytä linkin nimenä esimerkiksi pelkkää URL-osoitetta, joka on periaatteessa vain oppilaan muistia turhaan kuormittavaa informaatiota.
Ohje 7:	Pidä sivun yläalaidassa sitä tietoa, jota opiskelijat tarvitsevat eniten. Yläalaidassa oleva tieto on nopeasti saatavilla eikä sitä tarvitse etsiä skrollaamalla sivun alalaidasta.

Kuvassa 3 on esimerkki virtuaaliapprobatur-opintojen ”Ohjelmointi, osa 1” -kurssin verkkomateriaalin aloitussivulta. Kyseinen sivu antaa oppilalle yhteenvedon kurssilla käsiteltävistä asioista ja sivulta on myös linkit varsinaiseen materiaaliin. Sivun yläreunasta löytyy materiaalissa käytetty navigointiperiaate, joka mahdollistaa sen, että jokaiselta sivulta pääsee aina käsiksi viikoiksi jaettuihin kokonaisuuksiin. Oletamme, että materiaali tulee oppilaille sen

verran tutuksi, että he pystyvät navigoimaan viikkonumeroiden mukaan. Näin he pystyvät hahmottamaan omien tarpeidensa mukaan, minkä viikon tietoja tarvitaan esimerkiksi ongelmatilanteissa.

Virtuaaliapprobatur-projektissa olemme pyrkineet noudattamaan taulukossa 2 esitetyn tyyli-
pohjan ohjeita 3-7 linkkien asettelujen ja sivujen pituuksien suhteen, jotta oppimateriaali olisi mahdollisimman loogisesti ja helposti hahmotettavissa. Lisäksi pidimme erittäin tärkeänä, että jokaiselta materiaalin sivulta pääsee siirtymään toisille sivuille vain muutaman hiiren napautuksen avulla.

The image shows a screenshot of a web page for a course titled "Ohjelmoinnin perusteet, osa 1". At the top, there is a navigation menu with links for "Kursin etusivu" and "Viikko 1" through "Viikko 10". Below the menu, the page title "Ohjelmoinnin perusteet, osa 1" is displayed. The main content area is titled "Kurssin sisältö" and contains a list of 10 weeks, each with sub-items for "Teoria" and "Harjoitus". To the right of the list is a cartoon illustration of a grey cloud with yellow lightning bolts. On the left side of the page, there is a sidebar with a navigation menu containing links for "Kursin etusivu", "Kurssilla tarvittavat", "Teoriat" (with sub-links for weeks 1-10), and "Harjoitukset" (with sub-links for weeks 1-10).

Kuva 3: Esimerkki HTML-kielellä toteutetusta oppimateriaalista.

2.3 Tietoverkkopohjaisen oppimisympäristön toimintavaatimukset

Edellisessä kohdassa kävin läpi verkko-opiskelussa käytettävän oppimisympäristön osa-alueita, jotka vaikuttavat ympäristön toimintaan. Seuraavaksi käsittelemme niitä vaatimuksia, joita tietoverkossa olevalta oppimisympäristöltä ja sen osa-alueilta vaaditaan, jotta ympäristö olisi oppimisen kannalta ihanteellinen. Osaksi nämä vaatimukset ovat samoja kaikille oppimisympäristöille (Pantzar 1997), mutta tässä luvussa on tarkoituksena keskittyä nimenomaan tietoverkossa toimivan oppimisympäristön tärkeimpiin toimintavaatimuksiin.

2.3.1 Opetuksen järjestäminen

Koska oppiminen voidaan määritellä tapahtuvan riittävän interaktion avulla tiedon tai taidon hankkimiseksi, niin pelkästään esimerkiksi World Wide Web -materiaalin julkaisemista ei voida pitää riittävän keinona opetuksen tarjoamiseksi. Opetustapahtuman katsotaan yleisesti sisältävän seuraavat seitsemän elementtiä: oppilaan motivointi, opiskeltavan asian määrittely, aikaisempaan tietoon palaaminen, uuden tiedon prosessointi, palautteen ja ohjauksen antaminen, ymmärryksen testaaminen ja lisätiedon tarjoaminen. Hoffmanin ja Ritchien (1997) mielestä edellä mainitut elementit voidaan, jopa pitäisi, sisällyttää myös verkko-opiskeluun.

Oppilasta voidaan motivoida esimerkiksi tarjoamalla näyttävää grafiikkaa, värejä, animaatiota ja ääntä, mutta silloin motivaatio tulee oppimisympäristön ulkoisista puitteista. Tämä voi alussa olla oppilaita kiehtova seikka, mutta pidemmällä aikavälillä pelkkä ulkoasu ei riitä kannattamaan toisinaan raskasta ja pitkäkestoista oppimisprosessia. Hoffman ja Ritchie (1997) esittävät motivointiongelman ratkaisemiseksi tiedon tarjoamista muun muassa ristiriitojen tai mysteerin avulla, joka herättää oppilaissa mielenkiinnon itse aiheeseen. Lisäksi tiedon tulisi olla opiskelijalle relevanttia ja arvokasta, jotta tiedon hankkiminen katsotaan tarpeelliseksi.

Oppimistavoitteiden määrittelyllä vältetään mahdollisilta väärinkäsityksiltä, joita tietoverkkojen rajoitetuimmassa kommunikaatiokontekstissa saattaa helposti syntyä. Näin oppilailla on jatkuvasti tiedossa se, mihin on tarkoitus pyrkiä ja mitä heiltä odotetaan. Esimerkiksi oppi-

materiaalin ulkoisista linkeistä saattaa tässä tapauksessa tulla ongelma, jos niiden merkitystä ei ole tarkoin selvitetty. Erittäin motivoituneet ja ahkerat oppilaat saattavat turhaan opiskella ulkoisten linkkien tarjoamia tietoja, vaikka ne on tarkoitettu vain ylimääräiseksi informaatioksi (Hoffman ja Ritchie 1997).

Aikaisempaan tietoon palaaminen on kognitiivisen psykologian mukaan ehtona oppimiselle. Sen avulla uusi tieto linkitetään aikaisempaan ja oppilas pystyy rakentamaan näin merkityksellisiä siteitä vanhan ja uuden tiedon välille (Hoffman ja Ritchie 1997). Pelkkä vanhaan tietoon viittaaminen ei kuitenkaan riitä vaan aktiivinen sekä uuden että vanhan tiedon prosessointi mahdollistaa paremmat mahdollisuudet oppimisen tueksi. Tiedon prosessointi voi olla luokitelua, vertailua, virheiden analysointia ja eri perspektiivien löytämistä. Aktiiviseen prosessointiin kuuluu tärkeänä osana myös omien esitysten rakentaminen opitusta aiheesta, jonka avulla voidaan arvioida oppimisen syvyyttä.

Ohjauksen ja palautteen avulla oppilaat saavat tärkeää tietoa omasta kehityksestään, ja tähän voidaan lisätä myös tiedon testausta. Testauksen tarkoituksena on selvittää, ovatko opetettavat asiat oppilaiden hallinnassa, ja sen avulla voidaan päätellä muun muassa lisäinformaation antamisen tarve (Hoffman ja Ritchie 1997). Pidemmälle edistyneille oppilaille voi syventävällä tiedolla olla ratkaiseva merkitys motivoinnin kannalta. Jos he ovat jo näyttäneet taitonsa tai tietonsa opetettavissa asioissa, voidaan syventävän tiedon antamisella kannustaa opiskelemaan omaehtoisesti vaikeitakin asioita.

2.3.2 Kanssakäyminen eri tahojen kesken

Granger ja Passerini (2000) näkevät kanssakäymisen erittäin tärkeänä osana oppimisprosessia verkotetussa ympäristössä. Heidän mielestään oppimisen tueksi tarvitaan opiskelija–opiskelija-kommunikaation lisäksi myös opiskelija–sisältö-, opiskelija–ohjaaja-, opiskelija–muu materiaali- ja opiskelija–muu ohjaaja-kanssakäymisen muotoja. He eivät halua pitäytyä pelkästään ihmisten väliseen kanssakäymiseen, vaan oppijalla täytyy olla myös mahdollisuus ”kommunikoida” materiaalin kanssa.

Myös Lloyd (2000) nostaa yhteydenpidon (diskursion) oppimisympäristön eri tahojen kesken merkittäväksi tekijäksi, jos tietoverkossa tapahtuvan opiskelun halutaan olevan samassa ase-

massa lähiopetuksen kanssa. Kanssakäymisen tulisi olla hyvin monipuolista, joten suotavaa olisi, jos ympäristössä olisi mahdollisuus sekä synkroniseen että asynkroniseen kommunikointiin. Molemmat kommunikoinnin muodot ovat tärkeitä, jotta ympäristö pystyisi reagoimaan mahdollisimman monipuolisesti opiskelijoiden tarpeisiin. Oppimisympäristössä tapahtuva kommunikointi olisi tehtävä mahdollisimman vaivattomaksi opiskelijalle, jotta kanssakäyminen ympäristön eri osapuolten kanssa olisi luonnollinen osa tietoverkossa tapahtuvaa opiskelua.

Voisi myös ajatella, että etäisyydestä tuleekin siemen oppijoiden väliselle yhteydenpidolle varsinkin silloin, kun opetus on tarkoitettu isolle joukolle ihmisiä. Kyseisessä tapauksessa verkot voivat tarjota perinteistä opetusta paremman vaihtoehdon ideoiden jakamiselle, keskustelulle ja yleiselle kanssakäymiselle (Granger ja Passerini 2000). Isoissa luentosaleissa istuvat opiskelijat saattavat pelästyä suurta ihmisten määrää, eivätkä näin uskalla tuoda esille omia mielipiteitään tai kysymyksiään. Tästä johtuen kommunikointi sekä toisten oppilaiden että luennoitsijan kanssa voi jäädä minimaaliselle tasolle. Tämä kyseinen ilmiö on ”varsin yleinen” Suomessa, mutta toisissa kulttuureissa oman mielipiteen esille tuominen koetaan etuoikeudeksi ja jopa välttämättömyydeksi. Tietoverkkojen tarjoamat palvelut voivat toimia eräänlaisena vapaampana keskustelufoorumina sekä oppilaiden että opettajien kysymyksille ja omille ideoille. Ideaalisessa tapauksessa tietoverkot voivat jopa antaa opiskelijalle tunteen kuulumisesta tiettyyn isompaan ryhmään, joka antaa turvallisuuden tunnetta opiskelua varten.

2.3.3 Materiaali opiskelun tukena

Oppimateriaali ja sen jakelutapa on Grangerin ja Passerinin (2000) mukaan yksi kriittisimmistä elementeistä oppimisympäristössä. Materiaalin merkitys oppimisympäristössä koetaan tärkeäksi, koska usein eri aloilla on oma looginen sisältönsä ja rakenteensa. Useilla aloilla tiettyä käsitettä tai tietokokonaisuutta sekä edeltää tietty määrä käsitteitä ja asioita että seuraa jokin tietty tietokokonaisuus. Opiskelijan täytyy monissa tapauksissa ymmärtääkseen ja oppiakseen jonkun tietyn asian osata n kappaletta edeltäviä asioita, joten materiaali kannattaisi esitellä loogisen rakenteensa mukaan (Karjalainen 1997).

Brown (1997) erottaa tiedon informaatiosta, koska tieto ei ole pelkästään kokoelma tosiasioita ja oikea oppiminen vaatii tiedon sisäistämistä. Me hajotamme tiedon, sulatamme sen ja sijoi-

tamme sen olemassa olevaan, erittäin monimutkaiseen linkittyneeseen tiedon ja ideoiden verkkoon. Materiaalilla on oma merkityksensä, kun oppilas rakentaa omaa tietorakennettaan opiskeltavasta aiheesta, jolloin materiaali voi ja sen pitääkin toimia osaltaan oppimisen ohjaajana. Väärin rakennettu materiaali saattaa vaikeuttaa esimerkiksi kokonaisuuden hahmotamista tai sitten opituista tiedoista tulee pelkästään yksittäisiä tiedonmurusia, joilla ei ole mitään tekemistä toistensa kanssa.

Suunnittelun kannalta vaatimukset keskittyvät yksittäisten osien laatuun ja siihen, koostuuko näistä osista kokonaisuus, joka tukee oppimista. Oppimateriaalin täytyy täyttää tietyt laatuvaatimukset ja sovelluskohtaiset kriteerit, jotta niiden varaan voidaan rakentaa toimiva oppimistilanne. Pantzarin (1997) mukaan materiaalilla pitäisi olla oppijalle myös mahdollisimman paljon merkitystä, jotta opiskelija kokee opiskeltavan materiaalin mielekkäänä ja merkityksellisenä.

Karjalainen (1997) esittää hyvän oppimateriaalin ominaisuuksiksi seuraavia piirteitä: materiaali pitäisi jakaa sopivan kokoisiksi osa-alueiksi, joilla olisi omat tavoitteensa, ja materiaalin rakenteen tulisi olla selkeä, sillä sekava rakenne voi viedä opiskelijan mielenkiinnon pois itse tekstistä. Hän toteaa myös, että hypertekstiin pohjautuvan materiaalin potentiaali hyvänä oppimismateriaalina perustuu sen kahteen päätekijään: solmuihin ja linkkeihin. Solmut mahdollistavat tietyn tietokokonaisuuden esittelyn ja linkit rakentavat solmujen välille assosiatiivisia tai loogisia yhteyksiä. Hypertekstillä on siis potentiaalia rakentaa opiskeltavan aiheen mukaisia materiaaleja, jotka pystyvät tarjoamaan sekä loogisesti että ajatuksellisesti merkityksellisiä oppimiskokemuksia.

2.3.4 Oppijakeskeisyys

Eräs oppimisympäristöistä usein puuttuva tekijä on ympäristön mukautuminen erilaisten oppilaiden tarpeisiin. Tällainen ympäristö ei pysty ottamaan huomioon oppilaiden henkilökohtaisia ominaisuuksia ja tavoitteita. Brusilovsky (1999) määrittelee oppilaiden tarpeet huomioon ottavia ympäristöjä joustaviksi (adaptive), jotka käyttävät älykkääksi luokiteltavia keinoja oppijakeskeisyyden saavuttamiseksi. Luvussa neljä keskityn tarkemmin niihin keinoihin ja periaatteisiin, joilla joustavasti ja älykkäästi toimivia oppimisympäristöjä toteutetaan.

Tietokoneet ja -verkot mahdollistavat jo opiskelijakeskeisten ympäristöjen rakentamisen, mutta opettajakeskeinen oppimismalli ei hetkessä muutu koulutuksen eri tasoilla. Pantzar (1997) löytää opettajakeskeisessä mallissa pysymisen syiksi muutamia argumentteja, jotka estävät innovatiivisten, oppijakeskeisten menetelmien käyttöönottoa koululaitoksissa. Hänen mukaansa yliopistoissa "elävällä" kielellä on edelleen keskeinen rooli opetuksessa, eikä kouluilla välttämättä ole resursseja tehdä muutoksia. Lisäksi laitteistojen ja ohjelmistojen nopea kehitys jarruttaa opetuksellista kehitystä, koska opettajien ja oppilaiden energiaa kuluu uusien laitteiden ja ohjelmistojen toiminnan ja ominaisuuksien opetteluun.

Levi (1994) tuo esille, että oppimisympäristön pitäisi ottaa huomioon seuraavat jokaiseen oppijaan eri lailla vaikuttavat tekijät: oppijoiden mahdollisuudet ja kokemukset, oppimistavoitteet, oppimisympäristön luonne ja opiskeluun käytettävä aika. Pantzar (1997) puolestaan näkee oppijan mahdollisuuden hallita ja muuntaa erilaisia oppisisältöjä omien näkökulmien ja osaamisen perusteella yhdeksi tärkeimmistä tekijöistä rakennettaessa oppijakeskeisiä ympäristöjä. Hän kaippaa mekanismeja materiaalin ja esitysten järjestyksen hallintaan, oppimisaktiiviteettien valintaan ja ohjeiden rytmitykseen. Näiden toimien avulla ympäristöistä voidaan luoda paremmin oppijan henkilökohtaiset ominaisuudet ja oppimistyyliä huomioon ottavia järjestelmiä, jotka pystyvät reagoimaan oppijan käyttäytymiseen sopivin keinoin.

2.3.5 Oppimista tukevat välineet

Edellä olevien kohtien lisäksi katson viidennen tärkeän asian olevan erilaiset välineet, joilla oppimista voidaan tukea mahdollisimman tehokkaasti. Alpert et al. (1997) toteavat näiden välineiden parantavan oppijan kykyä oppia, visualisoida ja ymmärtää vaikeita käsitteitä. Taulukossa 3 on esitelty viisi Perkinsin (1991) mainitsemaa erityyppistä oppimista tukevaa välinettä, joita voidaan hyödyntää opiskelussa hyvin monin eri keinoin.

Bostock (1997) tekee samansuuntaisen jaon ohjelmistotyypeistä, joita verkotetut tietokoneet pystyvät tarjoamaan opiskelun avuksi. Hän haluaa korostaa tiedonlähteiden asemaa oppimisessa, koska tietoon pääsemisen määrä, laatu ja helppous on tärkeätä, vaikka oppiminen ei itsessään ole pelkkää tiedon hankkimista. Bostock (1997) haluaa korostaa yhteistyö- eli kommunikaatiivälineiden merkitystä erityisesti verkko-opiskelussa, joita vuorostaan Perkins (1991) ei tuo esille oppimista tukevinä välineinä.

Taulukko 3: Perkinsin (1991) esittämä luokittelu oppimista tukevista välineistä.

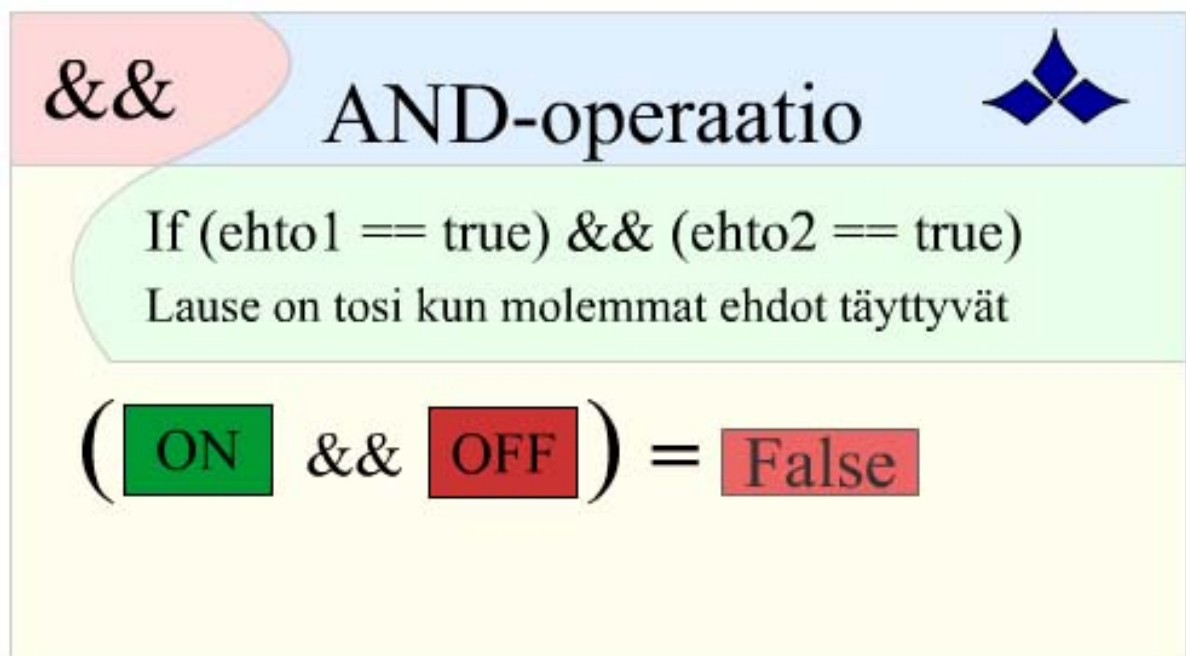
Väline:	Kuvaus:
Tietopankit eli tiedonlähteet	Kirjat, opettajat, sanakirjat, Internet ja oppimateriaali.
Symbolialustat	Muokataan ja konstruoidaan kieltä ja symboleita. Esimerkiksi muistiinpanot, tekstinkäsittelyohjelmat, piirto-ohjelmat ja käsitekartat.
Simulaatiot (ilmiön mallintaminen)	Ohjelmia, joilla esitellään, tarkkaillaan ja muokataan reaali maailman ilmiötä. Voidaan kokeilla asioita, joita ei normaaleissa olosuhteissa olisi mahdollista kokea.
Konstruktiovälineet	Samantapaisia kuin simulaatiot, mutta eivät ole niin paljon yhteydessä luonnolliseen ilmiöön. Konstruktiovälineet voivat olla komponentteja, joita kootaan ja muokataan halutulla tavalla.
Tehtävähallinta-, kontrollointi- ja ohjausvälineet	Elementit, jotka määräävät tehtäviä sekä tarjoavat ohjausta ja palautetta (eivät ole opiskelijan käytettävissä).

Kaikilla tässä esitetyillä välineillä voidaan oppilaille tarjota muun muassa visuaalisia konstruktioita opetettavasta aiheesta. Tämä on erittäin ratkaisevaa esimerkiksi monimutkaisten ja abstraktien asioiden opiskelussa, koska pelkästä tekstitiedosta opiskelija ei välttämättä voi hahmottaa kokonaisuuksia ja kaikkein tärkeimpiä asioita. Maureen (2000) näkee erilaisten välineiden merkityksen relevantin tiedon saamisessa esimerkiksi valokuvien ja dokumenttien avulla sekä virtuaalisten kokemusten hankkimisena. Erilaisten simulaatioiden ja virtuaali maailmojen avulla oppilaille voidaan tarjota ja he voivat myös itse konstruoida kokemuksia ja elämyksiä, joita ei välttämättä pysty kokemaan todellisessa elämässä.

Esimerkkinä oppimista tukevista visuaalisista välineistä voidaan todeta muun muassa Excel-animaatiot, joilla voidaan rakentaa alustoja erilaisten ilmiöiden tutkimiselle (Dybdahl et al. 1998). Myös ohjelmoinnin opetuksen apuna käytettävä Jeliot-ohjelma voidaan luokitella visuaaliseksi välineeksi. Jeliot mahdollistaa Java ohjelmointikielellä tehtyjen ohjelmien ja algo-

ritmien visuaalisen esityksen, jolla toivotaan olevan syventävää vaikutusta oppimisen kannalta. Jeliot-ohjelmaa pyritään käyttämään hyödyksi myös virtuaaliapprobatur-opinnoissa ohjelmointiin liittyvien ja yleisesti vaikeaksi koettujen asioiden opettamisessa. Esimerkiksi lajittelualgoritmien hahmottamisessa hyvin toteutettu animaatio antaa opiskelijalle syvemmän ja konkreettisemmän kuvan lajittelun etenemisestä kuin sanallisesti muutamien kuvien avulla toteutettu esitys.

Kuvassa 4 on esimerkki virtuaaliapprobatur-opinnoissa käytetystä Flash-animaatiolla toteutetusta interaktiivisesta toiminnasta, jolla pyritään selventämään ohjelmoinnissa erittäin tärkeässä asemassa olevien loogisten operaatioiden käyttöä ja merkitystä. Ohjelmointikurssilla käytimme tämän animaation lisäksi paria muutakin Flash-animaatiota samalla tavoin selventämään eräitä muita ohjelmoinnissa tarpeellisia käsitteitä. Näiden animaatioiden tarkoituksena oli saada oppilaat kokeilemaan esimerkiksi, miten loogiset operaatiot toimivat eri tilanteissa, ja näin toivoimme oppilaiden saavan hieman syvällisemmän kuvan kyseisestä aiheesta. Tulevilla kursseilla pyrimme toteuttamaan samantyyppisiä aktiviteetteja muun muassa Java-applettien ja lyhyiden videoesitysten avulla.



The image shows a slide from a Flash animation. At the top left, there is a pink oval containing the text '&&'. To its right, the text 'AND-operaatio' is written in a large, black, serif font. Further right is a blue logo consisting of three overlapping diamond shapes. Below this header, a light green rounded rectangle contains the text: 'If (ehto1 == true) && (ehto2 == true)' followed by 'Lause on tosi kun molemmat ehdot täyttyvät'. At the bottom, a light yellow rounded rectangle contains a logical expression: '(ON && OFF) = False'. The words 'ON' and 'OFF' are each inside a colored square (green and red respectively), and 'False' is inside a red square.

Kuva 4: Esimerkki oppimisen apuna käytettävästä Flash-animaatiosta.

2.4 Verkko-opiskelun ongelmat

Kohdissa 2.2 ja 2.3 käsittelin tietoverkon tarjoamia mahdollisuuksia ja vaatimuksia oppimisen kannalta, mutta varsin usein kyseiset seikat jäivät verkko-opetuksessa toteutumatta. Tässä kohdassa käyn läpi niitä ongelmia, joita tulee esille, kun kaikkia seikkoja ei ole otettu huomioon. Nämä probleemat ovat osittain samoja etäopetuksessa kohdattavien kanssa, mutta toki tietoverkot tuovat mukanaan myös omanlaatuisia ongelmatilanteita. Tässä kappaleessa esiin tulevien ongelmakohtien ja puutteiden ratkaisemisella on merkittävä asema kehitettäessä ideaalisesti toimivia tietoverkkopohjaisia oppimisympäristöjä.

2.4.1 Opiskelijan asema

Verkossa tapahtuva opiskelu näyttää olevan yhä suositumpaa kaikkialla maailmassa ja kaikilla koulutustasoilla. Messingin (2000) mukaan verkko-opetuksen tarjoamiseen on useita syitä. Ensinnäkin hän toteaa verkko-opetuksen olevan perinteisiä opetusmenetelmiä, siis luentoja ja tunteja, halvemmän ja toiseksi hänen mielestään oppimateriaalit ovat korkealaatuisempia tietoverkoissa. Lisäksi verkkojen avulla on mahdollista tarjota entistä toimivampi oppimisympäristö erilaisia oppilaita varten. Opetusjärjestelyt ovat kuitenkin usein suunniteltu organisaation toiminnan kannalta, jolloin yksittäisten oppilaiden tarpeet saattavat jäädä täysin huomioimatta, vaikka periaatteessa tähän pyrittäisiinkin. Seuraavaksi käyn läpi niitä vaikeuksia, joita oppilaat saattavat kokea opiskellessaan tietoverkossa olevissa oppimisympäristöissä.

Epävarmuus ja stressi

Hara ja Kling (2000) määrittelevät tutkimuksessaan opiskelijoiden tuntemuksista verkkokursilla *stressin ja epävarmuuden* (distress) tilaksi opiskelun aikana, jolloin opiskelijat tuntevat ylitsempääsemättömiä ongelmia. Opiskelijat saattavat tuntea muun muassa turhautuneisuutta, yksinäisyyttä, paniikkia, epävarmuutta ja ahdistuneisuutta. Useat verkko-opetukseen tarkoitetut ympäristöt eivät välttämättä tarjoa oppilailleen riittävää tukea, jos oppilas ei sitä itse aktiivisesti hanki. Tämä saattaa varsin usein johtaa edellä mainitun kaltaisten tuntemusten heikentymiseen ja pahimmassa tapauksessa aiheuttavat tilanteen, jossa opiskelusta tulee mahdotonta. Hyvä esimerkki epävarmuuden aiheuttajasta on ohjaajan opiskelijoille antamat ohjeet ja tehtävät. Jos oppilaat eivät esimerkiksi osaa tulkita opettajan viestejä ja ohjeita oikealla taval-

la, niin he saattavat kokea itsensä epävarmaksi, koska eivät ole täysin varmoja itseensä kohdistuvista odotuksista ja vaatimuksista.

Jos opetus järjestetään tiukasti konstruktivististen lähtökotien perusteella, niin ympäristö voi vaikuttaa joistakin oppijoista sekavalta ja huonosti määritellyltä. Tämä tuo esille epävarmuutta ja kontrolloimattomuutta, joka voi pahimmassa tapauksessa estää oppimisen kokonaan. Opiskelija voi kokea oppimisympäristön kaoottiseksi, jolloin siihen sisään pääseminen voi aiheuttaa suuriakin ongelmia.

Messingin (2000) tutkimuksen mukaan verkko-opiskelu ei välttämättä ole kaikille oppijoille palkitsevaa, mikä puolestaan lisää epävarmuuden ja epäonnistumisen tunnetta. Tämän lisäksi hän näkee tietoverkoissa tapahtuvan opiskelun ongelmana sen, että opiskelijan vaaditaan toteuttavan useita erilaisia rooleja opiskelun aikana. Opiskelijalta odotetaan esimerkiksi perinteisesti opettajille kuuluvien tehtävien, kuten oppimisen arvioimisen ja oppimisprosessin hallintaa. Tämä saattaa aiheuttaa ylimääräisiä vaikeuksia, sillä opiskelijat eivät ole välttämättä valmiita ottamaan näitä erityyppisiä ominaisuuksia ja taitoja vaativia rooleja vastaan.

Sosiaaliset tekijät

Vaikka oppimisympäristöt tarjoavat usein monipuolisia välineitä yhteydenpitoon, vielä ei ole kuitenkaan tekniikan avulla kokonaan saavutettu kasvotusten tapahtuvan opetuksen tasoa. Messing (2000) puhuu epämukavasta opiskelutilanteesta, koska etäopetus ei pysty tarjoamaan oppilaan ja opettajan välistä todellista kanssakäymistä. Vaikka tekniikka tarjoaa periaatteessa synkronisia oppimistapahtumia, ne ovat vielä erittäin kaukana reaali maailman monisyisen kanssakäymisen tasosta. Vertaisten tuen ja kasvotusten tapahtuvan opetuksen vaikutuksen arvolle on erittäin vaikeaa löytää korvaavia keinoja.

Greening (1998) kuvaa Internetin tarjoamia kanssakäymisen muotoja vielä erittäin primitiiviksi, joten ne eivät pysty tarjoamaan merkityksellistä tukea opiskelulle. Perkins (1991) toteaaakin, että puutteellisesti toteutetut oppimisympäristöt epäonnistuvat usein huonon sosiaalisen tuen takia, jolloin opiskelijat jäävät yksin ja he saattavat kokea kohtuuttomia odotuksia ja vaatimuksia. Puuttuvan sosiaalisen tuen takia heillä ei ole keinoja käsitellä monimutkaisia ja monitahoisia ongelmia, joita opiskelussa saattaa ilmetä.

Haran ja Klingin (2000) tutkimuksen mukaan puuttuva tiivis sosiaalinen yhteys erityisesti opettajaan koettiin ongelmalliseksi. Opiskelijat eivät olleet oikein varmoja, mitä kurssilla oikeastaan pitäisi tehdä ja mitä opettaja heiltä odottaa. Koska oppilaat eivät tieneet opettajastaan oikeastaan mitään, niin heidän oli vaikea määritellä minkälaiset vastaukset tyydyttäisivät häntä. Normaalisissa opetustilanteissa oppilas todennäköisesti hyvin lyhyessä ajassa pystyy päättämään mitä opettaja häneltä oikeastaan vaatii ja mitkä vastaukset ovat toivottavia. Verkossa tapahtuvan opiskelun aikana opettaja voi jäädä hyvinkin kaukaiseksi ja kuten Haran ja Klingin (2000) tutkimuksen eräs opiskelija sanoo: ”On erittäin hankala tehdä vaadittuja tehtäviä, koska ei tiedä minkälainen vastaus oikeasti tyydyttää opettajaa”.

Kustannukset

Kuten aikaisemmin mainitsin, on hinta yksi tärkeimmistä seikoista, joka vaikuttaa varsinkin tietoverkkojen tarjoamien elektronisten materiaalien hyödyntämiseen opetuksessa. Oppilaan kannalta verkko-opiskelu saattaa olla huomattavasti kalliimpaa kuin perinteinen luentojen seuraaminen. Verkkoympäristöön täytyy aina ottaa yhteys toiselta tietokoneelta eikä yhteyden ottaminen ole välttämättä halpaa. Messingin (2000) ja Pantzarin (1997) mukaan ne opiskelijat, joilla ei ole varaa tietokoneeseen tai eivät saa yhteyttä Internetiin, ovat eri asemassa muiden opiskelijoiden kanssa ja verkko-opiskelusta voi tulla lähes mahdotonta.

Messing (2000) haluaa tuoda esille paperin erittäin tärkeänä kustannustekijänä oppilaan kannalta. Hänen Hong Kongissa ja Australiassa tehdyn tutkimuksensa mukaan yli puolet tutkimukseen osallistuneista verkko-opiskelijoista tulosti suuren määrän elektronisista materiaaleista. Verkko-opetus siis siirtää painetusta materiaalista tulevat kustannukset oppijalle ja lisäksi opiskelijan tulostama paperiversio oppimateriaalista ei välttämättä ole laadultaan parhaimmasta päästä.

Miksi sitten opiskelijat tulostavat materiaaleja? Samaisesta Messingin (2000) tutkimuksesta käy ilmi seuraavat syyt paperiversioiden suosimiseen opiskelijoiden keskuudessa: elektronisia materiaaleja ei voi levittää pöydälle, kuvaruutu ei anna tarpeeksi informaatioita eikä kuvaruudulla voi olla useita sivuja auki samanaikaisesti, materiaalin selailu vie liikaa aikaa ja elektroninen materiaali pakottaa opiskelemaan tietyssä paikassa.

Myös Carswell (1998) on sitä mieltä, etteivät WWW-materiaalit täysin korvaa painettua tekstiä. Hänellä on osittain samat ajatukset kuin Messingillä: Internetin tarjoamalla tiedolla ei ole käytännössä mitään rajaa, näytöltä lukeminen on hankalaa eikä elektronisia materiaaleja saada joka paikkaan vaikka teknologia onkin kehittynyt. Näistä argumenteista voimme päätellä sen, että paperin asema opetuksen jakeluvälineenä tulee olemaan vahva myös tulevaisuudessa.

2.4.2 Ongelmat käytettävässä teknologiassa

Internetiä käytetään verkko-opetuksessa luonnollisena jakelukanavana, jonka negatiivisiin puoliin ei aina haluta kiinnittää tarpeeksi huomioita. Yleinen väite on, että lisääntynyt Internetin käyttö parantaa monien ihmisten oppimista. Tämä ei välttämättä pidä paikkaansa, sillä kritiikiton usko Internetin mahdollisuuksiin oppimisessa voi luoda epärealistisia ja vaikeasti toteutettavia tavoitteita, jotka vaikuttavat negatiivisesti sekä oppijoiden että ohjaajien toimintaan (Carswell 1998). Nykyisin Internetin käyttöä perustellaankin liian usein sanonnalla ”koska se on tarjolla”, jolloin negatiiviset näkökulmat jäävät usein huomiotta (Greening 1998). Haran ja Klingin (2000) mielestä liian usein yliarvioidaan kommunikoinnin ja World Wide Webin kapasiteettia ja samalla aliarvioidaan sekä opiskelijoiden että opetusta tarjoavan organisaation henkilökunnan työn määrää, jolloin tuloksena on huonosti toimiva oppilaita ja opettajiakin liikaa rasittava oppimisympäristö.

Greening (1998) esittää lisää kritiikkiä World Wide Webin käytöstä opiskeluympäristönä. Ympäristöissä käytetyt hypertekstidokumentit eivät aina ole yhtenäisiä: toinen sivu voi olla täynnä informaatioita, kun taas toinen voi olla täysin merkityksetön oppijan kannalta. Hypertekstin avulla luotu ei-lineaarinen teksti aiheuttaa sen, että lukija kadottaa tekijän tarkoittaman logiikan, rakenteen ja järkevyyden. Tästä voi olla seurauksena ns. ”random facts”-tietoa, joka on oppijan päässä täysin irrallisena toisistaan ja muista tiedoista. Greeningin (1998) mukaan tietoverkossa toimivat etsintärobotit tuottavat usein varsin epäilyttäviä tuloksia, jotka eivät välttämättä auta opiskelijaa ongelmatilanteissa.

Messing (2000) ei halua aliarvioida tekniikan kehityksen mahdollistamia positiivisia ulottuvuuksia, mutta hänen mielestään uusien välineiden tai protokollien käyttö verkko-opetuksessa saattaa aiheuttaa yllättäviäkin ongelmatilanteita. Opiskelijalta voi helposti kulua liikaa aikaa ja energiaa ympäristön toiminnan sekä erilaisten välineiden opiskeluun ja näin ollen varsina-

nen opiskeltava asia saattaa jäädä vähemmälle huomiolle. Myös navigaation vaikeus ja toisinaan pitkät vasteajat saattavat viedä opiskelijan huomiota liikaa ylimääräisiin asioihin (Alpert et al. 1997).

Haran ja Klingin (2000) tutkimuksessa opiskelijat tunsivat usein turhautuneisuutta siitä, että pienet, käytettävästä ohjelmasta riippuvat ongelmat aiheuttivat paljon päänvaivaa ja ajanhukkaa. Ongelmaksi he kokivat sen, että ohjeet olivat riittämättömiä eikä ollut ketään keneltä kysyä apua. Näissä tapauksissa pienet, joskus täysin opiskeltavaan asiaan kuulumattomat vaikeudet saattavat aiheuttaa todellisia esteitä opiskelulle ja oppimiselle.

Taulukkoon 4 olen kerännyt tärkeimmät kohdissa 2.2, 2.3 ja 2.4 esitetyt verkko-opetuksen ominaisuudet neljästä eri näkökulmasta. Nämä näkökulmat muodostavat niin sanotun SWOT (Strength, Weakness, Opportunities and Threats)-taulukon, jossa tulee esille sekä positiiviset että negatiiviset seikat verkko-opiskelusta. Taulukko muodostuu sekä lähteiden antamien näkökulmien että omien mielipiteideni pohjalta siten, että olen valinnut siihen mielestäni kaikkein tärkeimmät seikat. Lisäksi heikkouksien kohdalta olen pyrkinyt tuomaan esille juuri tämän hetkistä tilannetta eli jonkinlaista yleiskatsausta, joka ei siis suinkaan koske kaikkia verkkopohjaisia oppimisympäristöjä.

Taulukko 4: Verkko-opiskelun vahvuudet ja heikkoudet.

Vahvuudet	Heikkoudet
+ opetuksen tarjoaminen suurelle joukolle + opetusmateriaaliin käsiksi pääseminen syrjäisiltä seuduilta + Internetin yleistymisen kaikilla elämäntiloilla	- World Wide Webiä ei ole suunniteltu opetusta varten - navigoinnin vaikeus eli ”lost in hyperspace” - tietolähteiden luotettavuus
Mahdollisuudet	Uhat
+ oppilaiden oppimistyyli voi olla vapaa + tarjotaan henkilökohtaista ohjausta + kurssien globalisoituminen + todellinen kommunikointi ja keskustelu oppilaiden kesken	- oppilaat jätetään yksin suuren tietomäärän keskelle - opiskelukustannukset opiskelijan harteille - opetuksen tuotteistuminen, jossa ei oteta huomioon erilaisten oppilaiden tarpeita

3. METADATA JA OPPIMISOLIOT

Nopeasti kehittyvä tieto, oppilaiden erilaiset tarpeet ja opetusta tarjoavien organisaatioiden vaatimukset yhdessä aiheuttavat sen, että jatkuvasti kehitetään uusia menetelmiä ratkomaan opiskeluun ja oppimiseen liittyviä ongelmia. Metadatan ja oppimisolioiden käsitteet tarjoavat mielenkiintoisen lähtökohdan erityisesti www-ympäristössä tapahtuvan opiskelun ongelma-kohtien ratkaisemiseen. Metadatan ja oppimisolioiden avulla on mahdollista rakentaa uudelleenkäytettäviä ja joustavia oppimateriaaleja, jotka pystyvät tehokkaasti vastaamaan jatkuvasti kehittyvän yhteiskunnan nopeasti muuttuviin vaatimuksiin.

3.1 Ontologia

Norris ja Russel (1995) esittävät kirjassaan *yleisen ontologian* tavaksi yhdistää päätöksiä, joilla voidaan esittää laaja valikoima erilaisia olioita ja niiden välisiä suhteita. He haluavat painottaa nimenomaan yleisiä ontologioita, jotka poikkeavat erikoisontologioista siten, että yleistä ontologiaa voidaan käyttää missä tahansa erikoistilanteessa. Dominique et al. (1999) antavat ontologialle erilaisen merkityksen eri tieteenhaaroissa. Esimerkiksi filosofiassa ontologia tarkoittaa mallia olemassa olevasta maailmasta. Tekoälyn sovellutuksissa ontologialla tarkoitetaan tietomallin konstruktiota, joka määrittelee käsitteet tai oliot, niiden attribuutit ja olioiden sisäiset suhteet.

Eräs ontologinen rakenne on kategorisointi, jonka avulla olemassa olevan maailman oliot pyritään jakamaan samoja ominaisuuksia sisältäviin luokkiin. Olioiden organisointi kategorioihin on tärkeä osa tiedon esittämistä, vaikka interaktio tapahtuu yleensä olioiden välillä, suurin osa päättelystä tapahtuu kategoriatasolla.

Toinen ontologiarakenne on yhdistää erityyppisiä olioita. Idea yhden olion kuulumisesta osaksi toista oliota on tuttu: nenä on osa päätä, Romania on osa Eurooppaa ja luku on osa kirjaa. Yleisesti olion kuuluminen toiseen olioon ilmaistaan PartOf-suhteella, joka kertoo toisen olion olevan osan toista. Lisäksi jokainen olio, jonka osina on toisia olioita, voidaan nimetä koosteolioksi (composite). PartOf-suhde on transitiivinen ja reflektiivinen, jolloin voidaan puhua PartOf-hierarkioista. Kuvassa 5 on PartOf-suhteen mukaan tehty päätelmäketju (Norris ja Russel 1995).

PartOf (Valtimo, Suomi)
PartOf (Suomi, Eurooppa)
PartOf (Eurooppa, Maailma)
=> PartOf (Valtimo, Eurooppa)
=> PartOf (Valtimo, Maailma)

Kuva 5: PartOf-hierarkian mukainen päättelyketju.

Edellä olleen mukaan ontologian avulla kuvataan siis reaalisen maailman olioita ja niiden välisiä suhteita. Miten ontologia sitten liittyy tämän luvun metadata- ja oppimisoliokäsitteisiin? Kuvitellaan tilanne, että opiskelija A on oppimistilanteessa. Ihanteellisessa tapauksessa hänellä on tiedossaan oppimistavoite eli oppilas A tietää, mihin hänen on pyrittävä. Lisäksi voidaan olettaa, että hänellä on käytössään rikas oppimateriaali. Periaatteessa päästäkseen tiettyyn tavoitteeseen hän joutuu arvioimaan erilaisten oppikokonaisuuksien merkitystä ja suhdetta (vertaa kohta 3.4).

Teoriassa voidaan rakentaa eritasoisista ja erilaisista oppimateriaaleista koostuva maailma, josta oppija voi arvioida ja valita hänelle parhaiten sopivat oliot päästäkseen tavoitteeseensa. Metadatan avulla pystytään identifioimaan tämän maailman oliot ja rakentamaan niiden välille suhteita, jotka auttavat kulloinkin tarvittavan oppimispolun hahmottamisessa. Metadatan ja olioiden avulla voidaan rakentaa kuvaus opetuksellisen maailman olioista eli rakentaa maailmaa kuvaava ontologia.

3.2 Mitä metadata on?

Ontologia-käsitteen pohjalta lähdän määrittelemään käsitteen metadatasta, jolla tulee olemaan keskeinen sija tutkimuksessani. Käytän tässä luvussa termiä *tieto-olio*, jolla tarkoitan digitaalista tai ei-digitaalista tietokokonaisuutta. Tietty tietokokonaisuus voi pitää sisällään hyvin erityyppistä informaatioita esimerkiksi tekstiä, kuvaa, animaatiota ja ääntä.

3.2.1 Määritelmä

Metadatan on perinteisen kuvauksen mukaan tietoa tiedosta eli sen avulla pystytään kuvailemaan tieto-olioiden sisältöä (Robson 2000). Markuksen (2000) mukaan termi *metadatan* tarkoittaa sanatarkasti taustatietoa jostakin ja metadatan voi auttaa sekä ihmisiä että koneita päättämään miten tietäntyyppistä tietoa voidaan soveltaa erilaisissa tilanteissa.

Griffin ja Wason (1997) sekä Weibel (1995) antavat metadatalle seuraavan määrittelyn: metadatan on muodollinen (formal) kuvaus tieto-olioista. Kirjoittaja, otsikko ja sisällönkuvaus ovat esimerkkejä metadatan, jotka kuvaavat kirjoja kirjaston kokoelmissa. Markus (2000) mainitsee kartan erinomaisena esimerkkinä metadatan, sillä se on pelkkää metadatan – tietoa kartan kuvaamasta kohteesta. Metadatan avulla pystytään lisäämään tieto-olioihin myös sellaista tietoa, jota ei välttämättä löydy tieto-olion varsinaisesta sisällöstä. Tämä mahdollistaa ylimääräisen tiedon saamisen, joka helpottaa esimerkiksi tiedon soveltamisessa.

Edellä annettujen näkökantojen perusteella päädyn omaa käyttöäni varten seuraavanlaiseen metadatan-käsitteen määrittelyyn. Metadatan on kuvaus tietyistä tietokokonaisuudesta ja sen tarkoituksena on auttaa käyttäjää tiedon soveltamisessa. Metadatan voi antaa myös sellaista informaatiota, jota ei välttämättä saada selville itse tiedosta. Metadatan on tämän määrittelyn pohjalta merkittävä asema käytettäessä ja toteutettaessa kaikenlaisia materiaaleja tietoverkoissa.

3.2.2 Metadatan materiaalien kuvaamisessa

Metadatan tärkein tarkoitus on antaa selvyyttä tieto-olioiden sisällöstä ja merkityksestä ilman, että tietokokonaisuuksien sisältöjä tarvitsisi käydä läpi kokonaisuudessaan (Griffin ja Wason 1997, Feldman ja Milstead 1999a). Metadatan on käytetty jo vuosikatoja kirjastoissa, joissa kirjat luokitellaan ja indeksoidaan tarkoin niiden ominaisuuksien mukaan. Metadatan avulla kirjoille voidaan antaa sisältöä kuvailevaa, luokittelua helpottavia attribuutteja (Feldman ja Milstead 1999a).

Metadatalta pystytään kuvaamaan myös sellaisia tieto-olioita, jotka eivät sisällä lainkaan tekstiä. Näistä tekstittömistä olioista mainittakoon esimerkkinä kuvat, maalaukset, äänet ja animaatiot. Feldman ja Milstead (1999a) kertovat metadatalta käytettävän kuvailtaessa muun muassa museoiden kokoelmia, joissa materiaalin kuvauksella on erittäin tärkeä asema teosten merkitysten selvittämisen kannalta.

Metadata voidaan tehdä tieto-olion luonnin yhteydessä tai se voidaan lisätä myöhemmin mahdollisen luokittelun yhteydessä. Jälkimmäinen tapa on vähitellen häviämässä ainakin tietoverkkojen yhteydessä, koska useiden tieto-olioiden sisältämiä suuria tietomääriä on hankala luokitella jälkikäteen (Feldman ja Milstead 1999a). Esimerkiksi Weibelin (1995) mielestä luokittelijoiden ja indeksoijien on mahdotonta käydä läpi jo olemassa olevia dokumentteja Internetissä, joten uusien tieto-olioiden tekijöillä on vastuu lisätä metatiedot dokumentteihinsa itsenäisesti.

3.2.3 *Metadataelementit*

Usein metadata toteutetaan joukolla *tietoelementtejä*, joiden katsotaan parhaiten kuvaavan olion sisältöä. Liitteestä 1 löytyy listaus, jossa on esitelty Dublin Core -metadataelementtijoukko (Dublin Core Metadata Initiative 1999). Vaikka erityyppiset tieto-oliot saattavat tarvita erilaista metadatalta, niin aina löytyy päällekkäisyyksiä. Esimerkkejä samankaltaisista elementeistä ovat tekijä, otsikko, luontipäivä, muokkauspäivä ja julkaisija (Feldman ja Milstead 1999b). Metadataelementeille onkin pyritty kehittämään erilaisia standardeja, jolloin samankaltaiset ominaisuudet saadaan tehokkaasti esitettyä. Standardeja käsitellään tarkemmin kohdassa 3.3.3.

Metadataelementit voivat joko sijaita erillään varsinaisista tiedonlähteistä tai ne voidaan sisällyttää tieto-olion yhteyteen. Edellisessä tapauksissa metadataelementteihin voidaan lisätä yksityiskohtainen kuvaus siitä, miten kuvailtavaan tietoon päästään käsiksi. Useissa tapauksissa metadata voi sisältää esimerkiksi verkko-osoitteen, josta tieto-olio on saatavissa (Heery 1996).

3.2.4 Miksi metadatalle on merkitystä tietoverkoissa?

Metadatan merkitys on kasvanut Internetin käytön lisääntymisen myötä. Kuten edellisessä kohdassa mainitsin, niin metadatalle pyritään samaan edes jonkinlaista järjestystä useisiin miljooniin www-dokumentteihin. Suuresta määrästä huolimatta sopivan digitaalisen materiaalin löytäminen tietoverkoista on toisinaan hyvin vaikeaa, koska hakukoneet voivat tuoda samalla haulla hyvin erityyppisiä tuloksia tai ne eivät yksinkertaisesti löydä oikeanlaisia dokumentteja.

Esimerkiksi World Wide Web -sivusto voidaan suunnitella ala-asteikäisten oppilaiden matemaattisten taitojen kehittämistä varten, mutta materiaalissa ei välttämättä ole mainintaa termeistä ala-aste ja matematiikka. Tässä tapauksessa pelkästään tekstiä läpikäyvät hakukoneet eivät löydä kyseessä olevaa materiaalia. Metadatan avulla pystytään tarkoin identifioimaan www-sivustojen tarkoitus ja suhde muihin materiaaleihin. Tämä vuorostaan mahdollistaa www-materiaalien helpomman ja tehokkaamman löytämisen (Griffin ja Wason 1997).

Tietoverkossa olevissa materiaaleissa on tiettyjä ominaisuuksia, joihin metadatan avulla pystytään Heeryn (1996) mukaan paremmin reagoimaan. Tietoverkossa voi olla fyysisesti toisistaan kaukanakin olevaa materiaalia, joka liittyy samaan aineistoon. Metadatalle pystytään nitomaan nämä materiaalit yhteen kuvaamalla kaikkien osien sijainti ja niiden suhteet toisiinsa, jotta materiaalin käyttäminen olisi mahdollisimman vaivatonta. Tietoverkoista löytyy usein myös samasta dokumentista eri tiedostotyyppille talletettuja versioita, jolloin metadatalle voidaan määrittellä esimerkiksi eri versioiden suhteet toisiinsa ja niiden käyttötarkoitukset.

3.3 Metadatan käyttäminen opiskelumateriaalien kuvaamiseen

Metadatalle voidaan luonnollisesti kuvata myös opiskelumateriaaleja, mihin on kehitetty useita erilaisia standardeja. Standardien tarkoitus on antaa käyttäjilleen raamit, joiden sisällä standardiin liittyviä asioita voidaan soveltaa. Metadata-tiedon kohdalla standardit määrittelevät muun muassa metadataelementtien käytön rajoituksia.

3.3.1 Metadataskeemat

Metadataskeemalla tarkoitetaan sitä rakennetta, jolla metadata esitetään. Skeemojen muoto riippuu usein seuraavassa luvussa esitellyistä metadatastandardeista. Erilaisten skeemojen käyttö on riippuvainen pitkälti metadatatiedon tarpeesta ja käytöstä, koska erityyppisiin tilanteisiin voidaan tarvita hyvin erilaisia metadataesityksiä. Desire-projekti esittelee taksonomian metadataskeemoista, jotka sisältävät kolme kategoriaa (Feldman ja Milstead 1999b):

- 1) Yksinkertainen muoto. Sisältää luokittelematonta tietoa, johon kuuluvat hakukoneiden ja indeksien tarvitsemat ominaisuudet.
- 2) Jäsentyynyt muoto. Perustuu standardeihin ja sisältää tiettyjä, tarkoin valittuja, elementtejä.
- 3) Rikas muoto. Perustuu kansainvälisiin standardeihin, jotka ovat tarkempia kuin edellisen kohdan standardit. Etsimisen ja paikallistamisen lisäksi tämän muotoisia metadataskeemoja voidaan käyttää olioiden tai kokoelmien dokumentoimiseen.

Myös Markus (2000) kuvailee saman periaatteen mukaan kolme tasoa, jotka pystyvät kuvaamaan erityisesti opetukseen tarkoitettuja materiaaleja. Ensimmäisellä tasolla (yleinen taso) käyttäjä voi yhdellä silmäyksellä tutustua tietoon. Hän saa tällä tavoin yleisen kuvan tiedosta ja tiedon syvyydestä. Ensimmäisen tason kuvaukset ovat lyhyitä esityksiä, jotka voidaan täyttää helposti käsin. Tämän tason mukaiset kuvaukset vastaavat Desire-projektin määrittelemiä ensimmäisen kategorian mukaisia kuvauksia, jotka voivat sisältää luokittelematonta ”vapaata tekstiä”.

Markuksen (2000) esittämällä toisella tasolla (tietojoukkojen taso) käyttäjä saa tarkemman kuvan tiedon sisällöstä, ominaisuuksista ja historiasta. Toisen tason kuvaukset ovat pitkiä ja tarkkoja kuvauksia. Tämän tason kuvaukset vastaavat pitkälti Desire-projektin toista kategoriaa, jossa metadatatieto esitetään luokiteltuina elementteinä. Kolmas ja tarkin taso (ominaisuuksien taso) antaa hyvin tarkan kuvauksen yksittäisen tieto-olion käyttäytymisestä. Viimeisen tason kuvailuun voidaan käyttää Desire-projektin toista tai kolmatta kategoriaa riippuen metadataelementtien käyttötarkoituksesta.

Näistä kahdesta samaan periaatteeseen tukeutuvasta esityksestä voidaan päätellä, että valittu metadataskeema riippuu pitkälti kuvailtavan tiedon tarpeesta. Jos tarve on hyvin yleinen ja määrittelemätön, niin metadataskeemaksi riittää esimerkiksi lista avaintermeistä. Mitä tarkemmin metadatatiedon tarve tiedetään, sitä yksityiskohtaisempaa ja luokittelevampaa metadatatiedon täytyy olla. Jotta tiedon soveltamisesta voidaan tehdä tarkkoja päätelmiä, niin edellä mainitut avainsanalistat eivät tule missään tapauksessa kysymykseen.

Koska tutkielmani tarkoituksena on selvittää metadatan käyttöä hyvinkin tarkasti määritellyssä tilanteessa, niin esimerkiksi Markuksen (2000) esittämän ominaisuuksien tason mukaiset metadataskeemat ovat todennäköisimmin kaikkein sopivimpia jatkon kannalta. Kohdassa 3.4 esiteltävien oppimisolioiden käytön kannalta hyvin tarkat kuvaukset kyseisistä olioista ovat paikallaan. Olioiden merkitysten ja suhteiden kuvaamiseen joudutaan käyttämään erittäin tarkkaa kuvausta, jotta olion merkitys oppimateriaalissa olisi mahdollisimman tarkasti selvitetty. Tulevissa luvuissa käsitelen tarkasti määriteltyjä metadataskeemoja, jotka antavat täsmällistä tietoa tieto-olioiden sisällöstä ja toiminnasta.

3.3.2 Opetuksellisesti käyttökelpoisen metadatan vaatimukset

Opetuksellisten materiaalien metadataelementtien vaatimukset ovat pitkälti samoja muidenkin alojen kanssa. Griffin ja Wason (1997), ARIADNE (1999) sekä Weibel (1995) nostavat esille kaksi tärkeintä asiaa, jotka tulisi ottaa huomioon rakennettaessa metadataelementtejä. Ensiksi indeksoinnin, eli metadatan luonnin, pitäisi olla mahdollisimman yksinkertaista, jotta elementtien käyttöönotto olisi vaivatonta. Toiseksi metadatan hyödyntämisen relevanttien materiaalien etsinnässä ja käytössä tulisi olla tehokasta ja helppoa.

ARIADNE (1999) sekä Griffin ja Wason (1997) korostavat edellä mainittujen seikkojen lisäksi kansainvälisen toiminnan välttämättömyyttä metadataelementtien kehittämisessä. Kansainvälisyydellä ja monikulttuurisuudella päästään siihen lopputulokseen, että metadata olisi neutraali sen kielen kannalta, jota sekä materiaalissa että materiaalia kuvailevassa metadatatassa käytetään. Jos metadataelementeistä saadaan kaikkialla maailmassa toimivia, niin sillä mahdollistetaan käytännössä rajaton yhteistyö materiaalien jakamisen ja käytön kannalta (Heery 1996). Tämä tavoite on hiukan optimistinen, sillä kulttuurit ovat maailmassa hyvin erilaisia ja

yhteisten asioiden löytäminen saattaa olla jopa mahdotonta. Tämä pitää paikkansa erityisesti opetuksen alalla, jossa yksittäisten maidenkin sisällä saattaa olla hyvin erilaisia näkökulmia muun muassa pedagogisista linjanvedoista.

3.3.3 Erilaiset metadatastandardit

Viimeisten viiden vuoden aikana on ilmaantunut useita metadataprojekteja, jotka pyrkivät määrittelemään omia metadataelementtejään. Näiden projektien yhteydessä on määritelty joukko *metadatastandardiehdotuksia*, jotka saattavat olla jopa ristiriidassa toistensa kanssa. Näistä projekteista Feldman ja Milstead (1999b) luettelevat: Dublin Core, IEEE-Metadata Conference (IEEE-1), Digital Library Collaboratory Working Groups (DLL), IMS ja CDWA (Categories for the Description of Works of Art).

Vaikka jokainen erilainen metadataformaatti on itsessään "standardi", viralliset standardoimisesta huolehtivat järjestöt ovat olleet erittäin aktiivisia enemmän muodollisten ja standardin asemaan kuuluvien metadatakuvausten määrittelemiseksi. Esimerkiksi ISO (International Standard Organization) on perustanut Metadata Working Groupin (MWG) vastaamaan metadatan määrittelemiseen ja hallitsemiseen liittyvistä standardeista.

Feldman ja Milstead (1999b) löytävät näiden erilaisten lähtökohtien taustalta kaksi selvästi eri lähtökohtiin perustuvaa ideaa. Osa standardien kehittelijöistä pyrkii pitäytymään kirjastollisissa ja akateemisissa metadatan lähtökohdissa ja heidän tarkoituksenaan on rakentaa uusi elektroninen standardi alkuperäisille ideoille. Toiset ryhmät tunnustavat kyllä standardien tarpeen, mutta ovat sitä mieltä, etteivät vanhat menetelmät tule kysymykseen, vaan uusia on kehitettävä.

Koska tutkimukseni kannalta erityisesti opetusmateriaalien kuvaukseen sopivat sovellukset ovat merkittäviä, keskityn seuraavaksi niiden käsittelyyn. Robson (2000) luettelee opetusmateriaaleja varten metadatastandardeja suunnittelevia organisaatioita: ADL (Advanced Distributed Learning), AICC (Aviation Industry CBT Committee), ARIADNE, Dublin Core, GESTALT (Getting Educational Systems Talking Across Leading-Edge Technologies) ja IMS (Instructional Management Systems).

Keskityn tarkemmin kolmeen oleellisempaan standardiin Dublin Coreen, IMS:ään, ARIADNE:n, jotka ovat Robsonin (2000) ja Markuksen (2000) mukaan tärkeimmässä asemassa kehitettäessä opetusmateriaalien metadatastandardeja. Katson aiheelliseksi esitellä erillään IMS- ja ARIADNE-standardit, vaikka ne käyttävätkin samaa IEEE:n (Institute of Electrical and Electronics Engineers) Learning Object Metadata-määrittelyä pohjanaan ja ovat pitkälti yhteneviä. Itse asiassa IEEE:n LOM-määrittelyn pohjana on IMS:n ja ARIADNE:n vuonna 1998 tekemä yhteinen ehdotus opiskelumateriaalien metadatastandardiksi (Anderson ja Wason 2000b).

Dublin Core

Dublin Core ei ole varsinaisesti suunniteltu oppimateriaaleja varten, mutta monet metadata-standardit ja -määrittelyt pohjautuvat Dublin Coren ydinjoukkoon. Tätä joukkoa voidaan tarkentaa eri käyttötarkoituksia varten, joten tämän takia Dublin Core soveltuu myös opiskeluun ja oppimiseen tarkoitettujen materiaalien metadataelementtien kuvaamiseen (Markus 2000).

Dublin Core syntyi Online Computer Library Centerin (OCLC) ja National Center for Supercomputing Applicationsin (NCSA) tukeman työpajan yhteydessä vuonna 1995 Dublinissa, Ohiossa. Työpajassa oli mukana kirjastonhoitajia, atk-suunnittelijoita ja SGML (Standard Generalized Markup Language)-eksperttejä, joilla kaikilla oli yhteinen tavoite (Ahronheim 1998). Työpajan tarkoituksena oli löytää joukko ydinelementtejä (core), joiden avulla pystyttäisiin kuvaamaan tietoverkoissa olevien materiaalien tärkeimpiä ominaisuuksia. Weibel (1995) korostaakin sitä, että Dublin Coren tarkoituksena oli olla sekä riittävän yksinkertainen www-dokumenttien luojien ja ylläpitäjien käyttöä ajatellen että riittävän kuvaileva dokumenttien haun parantamiseksi.

Weibel (1995) kirjoittaa Dublin Coren kehityksen kannalta keskusteluissa tulleen esille lähinnä metadatan käytettävyyteen vaikuttavat teemat. Ensiksi oltiin yleisesti samaa mieltä siitä, että metadataelementtien määrä kannattaa pitää mahdollisimman pienenä. Toiseksi useampien käyttäjien pitäisi pystyä ymmärtämään helposti elementtien tarkoitus ja kolmanneksi elementtijoukon tulisi vielä olla mahdollisimman joustava, jotta monipuoliset dokumentit pystyttäisiin kuvaamaan Dublin Core metadataelementeillä.

Edellä olevien lähtökohtien perusteella kokouksessa annettiin Dublin Coren kehitykselle seuraavat periaatteet. Dublin Coren tulisi olla Weibelin (1995) ja Ahronheimin (1998) mukaan:

- sisällöltään mahdollisimman kattava
- laajennettavissa
- syntaksista riippumaton
- valinnainen
- toistettavissa
- muunneltavissa

Koska Dublin Coren kaikkia elementtejä voidaan tarvittaessa toistaa, laajentaa, muuttaa tai jättää käyttämättä, niin sitä voidaan soveltaa kuvaamaan hyvin monentyyppisiä materiaaleja (Dublin Core Metadata Initiative 1999). Dublin Coren laajentaminen tapahtuu käytännössä kahdella tavalla. Ensiksi jokaiselle elementille voidaan antaa sen sisältöä tarkentavia osia, jotka pystyvät tarjoamaan lisää käytettävyyttä kuvailtaessa esimerkiksi hyvin spesifiä kohdetta. Toiseksi laajeneminen tapahtuu Ahronheimin (1998) ja Weibelin (1995) mukaan siten, että Dublin Core -formaatin käyttäjä voi lisätä omia elementtejään ja tarkentimiaan, jolloin käyttäjän omat metadataan kohdistuvat vaatimukset tulevat paremmin esille.

Vaikka Dublin Core -yhteisön periaate määrittellä metadastandardi yksinkertaisesti ja epätarkasti lisää elementtien käytettävyyttä eri aloilla, niin epämääräisesti muodostettu standardi lisää myös väärinkäytösten ja -ymmärrysten todennäköisyyttä. Voisi jopa ajatella, että Dublin Core ei olekaan standardi, koska se jättää liian paljon asioita käyttäjänsä päätettäväksi (Markus 2000). Jälleen kerran metadataan kohdistuvat vaatimukset määräävät pitkälti sen kumpi on tärkeämpää, tarkka määrittely vai joustavuus.

Viimeisin Dublin Coren metadatamäärittelystä on versio 1.1, johon kuuluu 15 tietoelementtiä. Versio 1.1 lisää edellisiin versioihin verrattuna Dublin Coren yhtenäisyyttä muiden yhteisöjen koostamiin metadastandardeihin nähden. Version 1.1 myötä Dublin Core -määrittelyä tehostettiin parantamalla elementtien selkeyttä, laajuutta ja sisäistä yhtenäisyyttä (Dublin Core Metadata Initiative 1999). Liitteestä 1 löytyy täydellinen kuvaus Dublin Core-metadatelementtijoukosta, johon viitataan tulevissa luvuissa esimerkkien yhteydessä.

IMS (Instructional Management Systems)

Anderson ja Wason (2000a) toteavat IMS-metadastandardin olevan Yhdysvalloissa kehitetty kuvaus, joka on suunniteltu erityisesti opetuksessa käytettävien digitaalisten materiaalien vaatimuksia ajatellen. IMS on seurausta EDUCOM:in ei-kaupallisesta projektista, johon osallistuivat yhdysvaltalaiset koulutusta tarjoavat organisaatiot ja heidän yhteistyökumppaninsa.

Andersonin ja Wasonin (2000b) mukaan IMS ei pelkästään keskity metadatomäärittelyjen kehittelyyn, vaan organisaation tavoitteena on tulevaisuudessa tuottaa avoin yleisstandardi verkko-opetuksen eri osa-alueita varten. Metadatan soveltaminen on suositeltavaa, koska erityyppisten opiskelumateriaalien etsiminen ja käyttö on ratkaisevassa asemassa kyseisen järjestelmän toiminnan kannalta.

IMS-metadatomäärittely perustuu IEEE:n standardin Learning Object Metadata versioon 3.5, jota on puolestaan kehittänyt Technology Standards Committee's Learning Object Metadata Working Group, yhteistyössä muun muassa IMS:n ja ARIADNE:n kanssa (Robson 2000). Alun alkaen LOM-dokumentaatio ei sisältänyt ohjeita metadatan toteuttamiseksi, mutta IMS-organisaatio suosittelee XML (eXtensible Markup Language)-kuvaukielen käyttöä (Anderson ja Wason 2000b). IMS-standardin jokainen elementti kuvataan viidellä tietoalkiolla: nimi, selitys, monimuotoisuus, tyyppi ja merkitys. Taulukossa 5 on esitelty IMS-standardin joikaista elementtiä kuvaavat tietoalkiot, joiden käytöstä löytyy esimerkki liitteestä 2.

Taulukko 5: IMS-standardin elementtejä kuvaavat tietoalkiot (Anderson ja Wason 2000b).

Tietoalkio:	Selitys:	Esimerkki:
Nimi (name)	Miten metadataelementti tulee kirjoittaa	Title
Selitys (explanation)	Elementin määritelmä	Resurssille annettu nimi
Monimuotoisuus (multiplicity)	Kuinka monta ilmentymää elementistä voi olla ja onko niiden järjestys merkittävä	Yksittäinen arvo

Tyyppi (type)	Onko elementin arvo teksti, numero vai päivämäärä	LangStringType (1024 merkkiä)
Perustelu (rationale)	Tietoa elementin käytöstä ja merkityksestä	Vastaa Dublin Coren DC.Identifier elementtiä

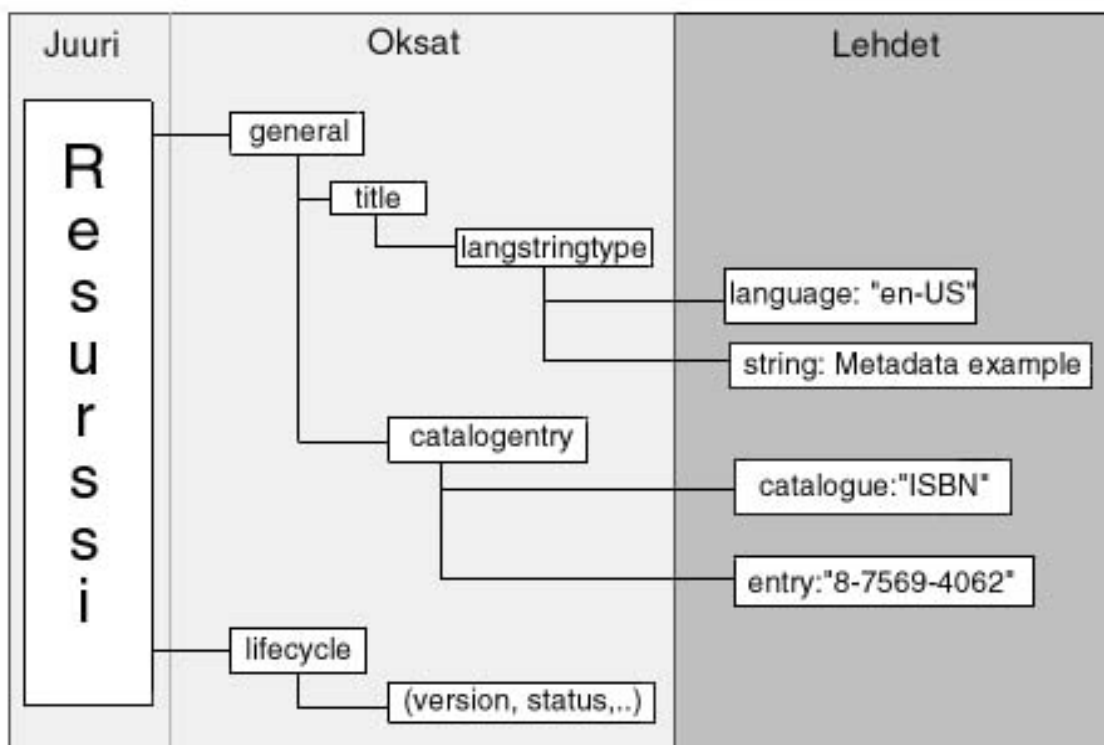
IMS-metadastandardin määrittelyssä oli Griffinin ja Wasonin (1997) mukaan lähtökohtana kolme päälinjaa: helppokäyttöisyys, laajennettavuus ja kansainvälisyys. Koska IMS-organisaatio ei ollut täysin tyytyväinen IEEE:n LOM-malliin, johtuen muun muassa sen monimutkaisuudesta ja laajuudesta, he ovat esitelleet kehitystyönsä tuloksena oman ydinelementtijoukkonsa (IMS Core). IMS perusteli oman ytimen toteuttamisen syyksi sen, että karstittu metadatajoukko parantaa nimenomaan projektin alussa painotettua käytettävyyttä ja helppoa käyttöönottoa. Andersonin ja Wasonin (2000b) mukaan tämä tarve on hyvin ymmärrettävää, sillä täysimittaisesti käytettynä LOM-mallin mukainen yhtä tieto-oliota kuvaava metadataesitys sisältää yli 80 elementtiä.

Helppokäyttöisyyttä IMS-standardissa halutaan erityisesti korostaa, jotta standardin käyttöönotto olisi mahdollisimman yksinkertaista eikä se vaatisi erityistä koulutusta tai taitoa. Laajennettavuudella IMS-standardissa haetaan sen käyttökelpoisuutta mahdollisimman laajassa mittakaavassa. Näin tiettyyn tietokokonaisuuteen voidaan lisätä juuri kyseistä tietoa parhaiten kuvaavia tietokenttiä. Griffin ja Wason (1997) toteavat kansainvälisyyden tuovan mukanaan mahdollisuuden käyttää eri kieliä joustavalla tavalla metadatakenttien rakentamiseksi esimerkiksi monikielisissä kulttuureissa.

IMS-organisaatio on kehittänyt IEEE:n LOM-mallia hieman tarkemmaksi ja jakanut sen kahteen eri osaan: IMS Core ja Standard Extension Library (SEL). Heidän tarkoituksenaan ei ole ollut muuttaa elementtien ominaisuuksia vaan identifioida ydinjoukoksi ne, jotka ovat avainasemassa. IMS-ydinjoukon kuvaus löytyy liitteestä 2. Loput LOM-mallin mukaiset elementit muodostavat laajennusjoukon eli SEL:in (Anderson ja Wason 2000b).

IMS-metadatatamalli on hierarkkinen ja hyvän kuvan metadataelementtien käytöstä saa, kun ne kuvitellaan puumaiseksi rakenteeksi. Puun juuri eli juurielementti, on kuvailtava olio itse. Juurielementti sisältää useita alielementtejä, jotka puolestaan voivat jakaantua useisiin muihin

alielementteihin. Jos alielementti sisältää alielementtejä, niin sitä kutsutaan oksaksi. Vuorostaan alielementit, jotka eivät sisällä muita alielementtejä, ovat lehtiä. Kuvasta 6 voidaan päätellä juuren, oksien ja lehtien suhteita IMS-standardissa.



Kuva 6: IMS-standardin rakenne (Anderson ja Wason 2000a).

ARIADNE (The Alliance of Remote Instructional Authoring and Distribution Networks for Europe)

ARIADNE:n esitys opetukselliseksi metadatastandardiksi on ollut kehityksen alaisena viimeisten viiden vuoden ajan ja kehitystyössä on ollut mukana 20:n eurooppalaisen yliopiston lisäksi viisi kansainvälistä yritystä. ARIADNE:n metadatastandardi on alaspäin yhteensopiva IEEE:n LOM-standardin kanssa (vertaa IMS), mutta ARIADNE-yhteisön omat tarpeet on otettu huomioon standardia kehiteltäessä (ARIADNE 1999).

ARIADNE-projektin päätavoitteena on hallita digitaalisten materiaalien jakamista ja käyttöä sekä yliopistojen että yritysten kannalta. Tätä tavoitetta varten ARIADNE-projektin myötä on

rakennettu muun muassa Knowledge Pool System, joka on Euroopan mittakaavaan tarkoitettu pedagogisten materiaalien tietokanta. Tietokantaan kuuluu asianmukainen indeksointi ja hakuvälineet, jotka helpottavat järjestelmässä olevien materiaalien hakua ja käyttöä.

ARIADNE-yhteisön periaatteena ei ole kehittää metadataa, joka kuvaisi ihmisten panosta opetusprosessissa tai karakterisoida ja tallentaa heidän oppimissuorituksiaan. ARIADNE-standardi ei myöskään ota kantaa miten, metadataelementit esitetään, vaan tähän tehtävään voidaan käyttää esimerkiksi SGML:ää, XML:ää tai RDF:ää (Resource Description Framework) (ARIADNE 1999). ARIADNE-standardi poikkeaa muun muassa Dublin Coren esittämästä standardista siten, että useimmat elementit ovat pakollisia. Tämä perustellaan ARIADNE-yhteisön puolelta sillä, että pakollisuus varmistaa metadataskeeman stabiilisuuden. Toki pakolliset ryhmät voivat sisältää valinnaisia kenttiä, joilla joustavuutta on pyritty tuomaan mukaan.

ARIADNE-standardi jakaa metadataelementit kahdeksaan ryhmään, ryhmät 1)-6) ovat pakollisia pedagogisten resurssien indeksoinnissa ja ryhmät 7) ja 8) ovat valinnaisia (ARIADNE 1999, Markus 2000):

- 1) yleistä tietoa resurssista
- 2) resurssin semantiikka
- 3) pedagogiset ominaisuudet
- 4) teknologiset ominaisuudet
- 5) käyttöehdot
- 6) meta-metadatan informaatio
- 7) ilmoitukset
- 8) fyysiset tietoresurssit

Vaikka ARIADNE-yhteisö pyrkii perustelemaan metadataelementtien pakollisuuden stabiilisuudella, niin mielestäni pakollisuus ei ole toimivan metadastandardin ominaisuus. Pakollisuus tuo mukaan sen, että metadataan joudutaan lisäämään sellaistakin informaatiota, joka ei ole aina tarpeellista. Valinnaisuus ja joustavuus mahdollistaa sen, että metadataelementtijoukko voidaan muuttaa varsin pienellä vaivalla muuttuneiden olosuhteiden mukaan.

3.3.4 Standardien vertailua

Edellisessä kohdassa esittelin kolme erityisesti opetusmateriaalien kuvailemiseen käytettäviä metadatastandardeja. Seuraavaksi tarkoitukseni on vertailla edellä esiteltyjä standardeja Heeryn (1996) antamien metadatakuvauksien vaatimusten mukaan. Esitän aluksi hänen antamansa kriteerit, joiden pohjalta vertailen Dublin Core, IMS:ää ja ARIADNE:a. Vertailussa on tarkoituksena kuvata, toteuttavatko standardit Heeryn asettamia arviointiperusteita ja miten tämä tulee esille kussakin standardissa. Jokaisessa kohdassa tuon esille omia näkemyksiäni esitellyistä standardeista, jotka eivät välttämättä käy yhteen Heeryn antamien kriteerien kanssa.

Vaikka Heeryn (1996) antamat kriteerit ovat jo muutamia vuosia vanhoja, katson niiden silti olevan vielä erittäin käyttökelpoisia, koska samojen periaatteiden mukaisesti toimivat useimmat metadataelementtejä kehittelevät yhteisöt. Kriteerit kattavat mielestäni kaikki olennaiset seikat, jotka kannattaa ottaa huomioon arvioitaessa erilaisten standardien toimivuutta ja käytökelpoisuutta erilaisten tilanteiden mukaan. Heery esittelee viisi erilaista kriteeriä, joiden pohjalta standardeja voidaan arvioida.

Heeryn (1996) kriteerit metadatastandardien vertailemista varten:

- 1) Käyttö. Kuka metadataa käyttää? Mitkä ovat käyttäjän vaatimukset? Yhdistetäänkö metadata tiettyyn tieteenalaan? Miten metadatastandardit vastaavat näihin vaatimuksiin?
- 2) Metadatan luonnin helppous. Tarvitaanko standardin käyttämiseen erityisiä taitoja? Kenen on tarkoitus tehdä metadataa, julkaisijan vai tarjoajan?
- 3) Metadatan sisältö. Millainen on tallenteen rakenne ja sisällön tarkoitus? Onko standardi käytännöllinen ja miten se vastaa erilaisiin vaatimuksiin? Voiko loppukäyttäjä perustellusti päätellä metadatan avulla tietö-olion käytettävyyttä?
- 4) Tietoverkkoon liittäminen. Voidaanko metadata liittää Internet-protokollaan? Miten elementit tehdään? Mitkä ovat dataelementtien toteuttamisen säännöt?

5) Kansainvälisyys. Voiko standardia käyttää kaikkialla maailmassa? Onko standardilla tunnustusta eri puolilla maailmaa?

Miten esittämäni standardit sitten täyttävät Henryn (1996) kriteerit opetuksellisten materiaalien ominaisuuksien suhteen peilattuna? Käyttökriteerit täyttyvät mielestäni kaikissa esiteltyissä standardeissa, sillä kaikkiin standardeihin pystytään lisäämään omia kategorioita. Lisäksi erityisesti IMS ja ARIADNE on suunniteltu opetusalan materiaaleja varten, joten niissä on pyritty ottamaan huomioon alan erityisvaatimuksia. Dublin Core sopii moniin aloihin joustavuutensa ja laajennettavuutensa ansiosta, joten senkin avulla voidaan opetuksellisten materiaalien käyttäjän metadataan kohdistamat vaatimukset täyttää. Voidaan päätellä, että kaikki standardit ovat opetuksellisen metadatan käytön kannalta samankaltaisessa asemassa.

Toinen ja kolmas kriteeri, eli metadatan luonnin helppous ja sisältö, tekevät suurimman eron standardien välille. Dublin Coreen perustuva metadataskeema on ylivoimaisesti helppokäyttöisin verrattuna IEEE:n LOM-standardiin pohjautuviin IMS:ään ja ARIADNE:en. Vaikka IMS ja ARIADNE pyrkivätkin ottamaan huomioon laajasti opetusmateriaaliin liittyviä asioita, tämä on johtanut siihen, että heidän standardeistaan on tullut vaikeaselkoisia ja raskaita käyttää.

IMS-yhteisö on esittänyt oman IMS Coren, joka supistaa käytettävien elementtien lukumäärää, mutta sekään ei mielestäni poista IMS:än monimutkaisuutta. Lisäksi ARIADNE-yhteisö on ilmoittanut, että heidän standardeissaan suurin osa elementeistä on pakollisia (ARIADNE 1999), jolloin standardia tiukasti noudattava joutuu soveltamaan monimutkaista ja turhaakin tietoa sisältävää metadatajoukkoa. Dublin Core lyö laudalta yksinkertaisuudellaan ja joustavuudellaan molemmat standardit. Jokainen käyttäjä voi määritellä omia elementtejään haluamallaan tavalla, jolloin oman aiheen ja käytön kannalta tärkeitä asioita voi lisätä metadataskeemaan täysin vapaasti.

Neljäs kriteeri on hiukan vanhanaikainen, koska kaikki standardit on suunniteltu nimenomaan tietoverkkoja varten. Jokaisessa esiteltyistä standardeista ei oteta kantaa siihen miten metadataelementit tulisi toteuttaa käytännössä. Kaikilla standardeilla on tietenkin ehdotuksensa siitä, mitä keinoja tulisi käyttää, mutta niitä ei ole yksiselitteisesti määrätty.

Viidennen kriteerin eli kansainvälisyyden osalta esitellyillä standardeilla ei ole suurtakaan eroa. Kaikki standardit ovat kansainvälisesti tunnettuja ja organisaatioiden tärkeimpinä tavoitteina onkin kansainvälisyyden korostaminen metadatastandardien kehittämisessä. IMS ja ARIADNE ovat tässä suhteessa hyvässä asemassa, koska ne ovat tehneet laajaa yhteistyötä keskenään yhteisen metadatastandardin kehittämiseksi.

3.3.5 Keinot metadatan toteuttamiseksi

Eräs kaikkein vaikeimmista asioista metadatan käytössä on päättää, millä tavalla metadataelementit tullaan toteuttamaan. Esittelen seuraavaksi kolme hieman toisistaan poikkeavaa tapaa, jotka tarjoavat kukin hiukan erilaisen ratkaisun metadataelementtien toteuttamiseksi. HTML:n ja RDF:n kohdalla on lyhyt esimerkki siitä, miten Dublin Core metadataelementtejä toteutetaan kyseisellä tavalla.

HTML

HTML on tällä hetkellä selvästi käytetyin tapa World Wide Web -dokumenttien luomisessa, joten luonnollisesti myös metadatakuvauksia toteutetaan vielä suurimmaksi osaksi HTML-kielellä. Vaikka XML:llä tarkennettuna RDF:llä on enemmän potentiaalia metadatan toteuttamiseen kuin pelkällä HTML:llä, kuitenkin HTML:llä on vielä pitkään vahva asema myös metadatakuvausten tekemisessä (Kunze 1999). Varsinkin pelkästään HTML-sivuja sisältävien materiaalien kuvaukseen HTML-dokumenttiin sisällytetty metadatakuvaus on toimiva ja helppokäyttöinen ratkaisu.

HTML-formaattiin (versiossa 4.0) on sisällytetty valmiina META- ja LINK-tagit, joihin pystytään tallentamaan kuvailevaa tietoa luodusta www-dokumentista (Ahronheim 1998). META-tagien syntaksi on suunniteltu nimetyn metadataelementin koodausta varten. Kuva 7 esittää virtuaaliapprobatur-opintojen ”Tietokone, käyttöjärjestelmä ja tietoverkko”-kurssin oppimateriaalin erästä sivua. Sivun on toteutettu HTML-kielellä ja siinä käydään läpi skannereita ja digitaalikameroita ja sivu liittyy Tietokonelaitteiston osat -osioon. Kuvassa 8, sivulla 47, on kyseisen sivun metadatakuvaus HTML-kielen META- ja LINK-tagien avulla toteutettuna.


Kuvan 8 kuvaus on lyhyt esimerkki niistä elementeistä, joita tyypillisesti käytetään kuvaamaan World Wide Web -materiaaleja. Kuvan tarkoituksena ei ole ottaa kantaa ovatko juuri kyseiset elementit kaikkein tärkeimmässä asemassa opetusmateriaalien kuvaamisessa, vaan tarkoitukseni on antaa lukijalle käsitys siitä minkälaista tietoa metadataelementteihin voidaan tallettaa.

TIETOTEKNIIKAN PERUSTEET

Joensuun yliopisto

VIIKKO 2

- ▶ Etusivu
- ▶ Keskusyksikkö
- ▶ Syöttölaitteet
- ▶ Tulostuslaitteet
- ▶ Apumuistit
- ▶ Modeemi
- ▶ Skanneri ja digitaalikamera
- ▶ Harjoitus
- ▶ Linkit

 s. 17-79

▶ Kurssin etusivu

▶ Viikko 1 ▶ Viikko 2 ▶ Viikko 3 ▶ Viikko 4 ▶ Viikko 5 ▶ Viikko 6

▶ Viikko 2 - Tietokonelaitteiston osat - Skanneri ja digitaalikamera

Skannerit ja digitaalikamerat

Kuvanlukija eli skanneri

Kuvanlukijalla saadaan luettua (skannattua) kuva digitaaliseen muotoon. Digitaalista kuvaa voidaan mikrossa muokata (parantaa, muuttaa, yhdistää) ja liittää ohjelmiin tai tallentaa tiedostoksi.

Kuvanlukijaryhmät:


- käsikuvanlukija (Hand Scanner)
- rullakuvanlukija (sivukuvanlukija, Page Scanner)
- tasokuvanlukija (Flatbed Scanner)
- rumpukuvanlukija (Drum Scanner)

Digitaalikamera


Digitaalikameralla voidaan ottaa kuvia aivan kuin tavallisella kameralla. Erona on se, että ei käytetä filmiä vaan kuva siirretään suoraan digitaaliseen muotoon kameralla olevaan muistiin. Kameramallista riippuen kuva voidaan siirtää levyllä tai kaapelin välityksellä mikeroon käsiteltäväksi. Kameroiden laatutaso vaihtelee hyvin paljon, kuten myös niiden hinnatkin. Edullisemmat kamerat ovat muutaman tonnin luokkaa, kun kalliimmat ylittävät satatuhatta markkaa.

Digitaalikameralla otetun kuvan lopulliseen laatuun vaikuttavat:

- kuvan tarkkuus eli kuinka monta pistettä kuvassa on. Joissakin kameroissa kuvan tarkkuus voidaan valita.
- kameran laatu eli miten hyvin värit toistuvat kuvassa
- pakkaustapa eli kuinka paljon tietoa häviää tiedon pakkauksen takia. Joissakin kameroissa pakkaustiheys on valittavissa.



HP ScanJet 6350C Series
(Kuva: Hewlett-Packard Oy)



Canon Digital Ixus
(Kuva: Canon Oy)

Kuva 7: Virtuaaliapprobatur-opintojen materiaalista.

Kuvan 8 metadatakuvaukseen en ole sovittanut kaikkia elementtejä, joita voitaisiin käyttää. Tämä johtuu siitä, että kyseisen esimerkin avulla haluan antaa pelkästään yleisellä tasolla käsityksen siitä, miten metadataelementtien toteutus onnistuu HTML-kielellä. Tässä vaiheessa en aio ruveta arvioimaan minkälaisia elementtejä esimerkin kaksi kaltaisessa kuvauksessa pitäisi olla, vaan jokainen metadatan toteuttaja voi valita omiin tarkoituksiinsa sopivia ele-

menttejä. Esimerkin kaksi metadatakuvaus voidaan ajatella sijaitsevan samassa HTML-tiedostossa kuin varsinainen kuvailtava sivu. Kuvauksen tekemiseen on hyödynnetty Kunzen (1999) esittämiä esimerkkejä, joissa on käytetty hyväksi elementtien tarkentimia, joita metadataan toteuttaja voi lisätä ilmenevien tarpeiden mukaan (Dublin Core Metadata Initiative 2000).

```
<html>
  <head>
    <title>Skanneri ja digitaalikamera</title>
    <link rel = "schema.DC"
      href="http://purl.org/DC/elements/1.0/">
    <meta name= "DC.Title"
      content = "Skanneri ja digitaalikamera">
    <meta name = "DC.Creator.Writer"
      content = "Jarkko Suhonen">
    <meta name = "DC.Creator.Writer"
      content = "Arto Haataja">
    <meta name = "DC.Creator.Illustrator">
      content = "Arto Piironen">
    <meta name = "DC.Subject"
      content = "skanneri ja digitaalikamera">
    <meta name = "DC.Date.Created"
      content = "2000-07-10">
    <meta name = "DC.Date.Available"
      content = "2000-08-30">
    <meta name = "DC.Type">
      content = "educational web-page">
    <meta name = "DC.Format"
      content = "text/html">
    <meta name = "DC.Language"
      content = "fi">
  </head>
  <body>...</body>
</html>
```

Kuva 8: Dublin Core metadataelementit toteutettuna HTML-kielillä.

Lyhyesti esitettynä kuva 8 pitää sisällään esimerkiksi tietoa sivun tekijöistä (DC.Creator), joita voidaan tarpeen mukaan luetella useita. Kuvauksessa on mukana myös dokumentin elämänsäkaareen liittyviä tärkeitä päivämääriä (DC.Date), joista saadaan selville milloin kyseinen sivu on ollut kehittelyn alla. Lisäksi löytyy yleistä tietoa sivun sisällöstä, käytetystä kielestä ja

sivun tiedostomuodosta (DC.Format). Liitteestä 1 löytyy tarkka kuvaus niistä 15:stä elementistä, jotka muodostava Dublin Core-standardin ja joita voidaan käyttää rakennettaessa metadatakuvauksia erilaisista tieto-olioista. Esimerkissä käytetään muun muassa elementin DC.Creator-kohdalla tarkentimia, joita ei löydy liitteen 1 esityksestä. Tarkentimien tarkoituksena on antaa tarkempaa informaatiota kuvailtavasta kohteesta (Dublin Core Metadata Initiative 2000). Kuvassa 8 tarkentimien avulla voidaan tarvittaessa lisätä informaatiota esimerkiksi kuvailtavan kohteen graafisen ulkoasun suunnittelijasta.

XML (eXtensible Markup Language)

XML-kuvailuformaatti on kehitetty SGML-formaatin pohjalle toimimaan erityisesti www-ympäristössä (Ahronheim 1998). Se perii tiettyjä SGML:n ominaisuuksia, mutta on selvästi yksinkertaisempi tavallisen käyttäjän kannalta. Bourdan ja Helierin (1999) mielestä XML näyttää lupaavimmalta keinolta metadatan toteuttamiseksi, koska se tarjoaa esimerkiksi HTML-formaattia tehokkaammat keinot sekä materiaalin että metadatan rakenteiden määrittämiseksi.

XML:n käyttökelpoisuutta puoltaa erityisesti se, että XML erottaa dokumentin sisällön ja ulkoasun toisistaan (Bourda ja Helier 1999). Tämä mahdollistaa sen, että XML-dokumentin ulkoasu voidaan muuntaa automaattisesti eri sovellusten tarpeiden mukaan. Koska metadatan toimivuus erilaisissa ympäristöissä on yksi sen tärkeimmistä ominaisuuksista, niin XML tarjoaa tätä varten erittäin tehokkaan toteutuksen.

RDF (Resource Description Framework)

RDF on W3C:n (World Wide Web Consortium) kehittämä formaatti, joka kuvaa www-dokumentin ominaisuudet ja sen mitä arvoja nämä ominaisuudet voivat saada (Bourda ja Helier 1999 sekä Brickley et al. 1999). RDF-tietomallin perustana on selvästi identifioitava resurssi, jota kuvataan erilaisilla ominaisuuksilla. Esimerkkeinä ominaisuuksista mainittakoon "koko", "nimi" ja "tekijä", joilla voi olla arvot "270 kb", "Oheislaitteet" ja "Virtuaaliapprobatuur-tiimi".

Vaikka suurin osa esimerkiksi Dublin Coren käyttäjistä toteuttaa metadatakuvauksensa HTML-kielellä, erityisesti RDF:n selvästi tehokkaammat ominaisuudet metadatan kuvausten

määrittelyssä puhuvat sen käytön puolesta (Brickley et al. 1999). Swick (1999) kuvaakin RDF:n käyttöä yksinkertaisena tapana tehdä tietokoneen ymmärtämiä lauseita tai ilmoituksia eli kuvauksia www-sivuista. RDF keskittyy metadataelementtien syntaksin ja rakenteen kuvaamiseen, eikä se ota kantaa elementin semantiikkaan kuten jossain määrin XML ottaa.

RDF-formaatin avulla voidaan esimerkiksi luoda omia kuvauksia, mikä mahdollistaa RDF:n käyttökelpoisuuden erityyppisten materiaalien kanssa. RDF:llä määriteltyä metadatakuvausta pystytään tarkentamaan ilmenevien tarpeiden mukaan, mikä on erittäin käyttökelpoinen ominaisuus hyvin nopeasti muuntuvassa World Wide Web-ympäristössä (Ahrnheim 1998). Esimerkki RDF:ää käyttävästä palvelusta on Netscape Navigatorin "What's Related", joka perustuu toisistaan etäällä olevien tietokantojen suhteiden määrittelyyn World Wide Webissä (Swick 1999). Palvelu tarjoaa listan www- sivuista, jotka ovat kiinnostavia luettavissa olevan sivun kannalta.

Kuvassa 9 on sivulla 46 olevan kuvan 7 esittämän HTML-sivun RDF-dokumentti. Alla olevassa esimerkissä on käytetty Dublin Core metadataelementtejä, joten se vastaa kuvan 8 kuvausta, joka on toteutettu HTML:llä. Koska tämän tutkimuksen puutteissa en ryhdy tarkemmin analysoimaan RDF-formaatin syntaksia tai semantiikkaa, niin esimerkki antakoon riittävän kuvan siitä, miten RDF-dokumentti rakentuu.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<rdf: RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.0"
  xmlns:dcq="http://purl.org/dc/qualifiers/1.0"
  <rdf:Description rdf:about="v2_6.htm">
    <dc:title>Skanneri ja digitaalikamera</dc:title>
    <dc:creator>
      <rdf:Bag>
        <rdf:li>Jarkko Suhonen</rdf:li>
        <rdf:li>Arto Haataja</rdf:li>
        <rdf:li>Arto Piironen</rdf:li>
      </rdf:Bag>
    <dc:subject>skanneri ja digitaalikamera</dc:subject>
    <dc:date>
      <rdf:Bag>
        <rdf:li>
```

```
<rdf:Description>
  <dcq:dateType>Created</dcq:dateType>
    <dcq:dateScheme>ISO8601</dcq:dateScheme>
  <rdf:value>2000-07-10</rdf:value>
</rdf:Description>
</rdf:li>
<rdf:li>
  <rdf:Description>
    <dcq:dateType>Available</dcq:dateType>
      <dcq:dateScheme>ISO8601</dcq:dateScheme>
    <rdf:value>2000-08-30</rdf:value>
  </rdf:Description>
</rdf:li>
</rdf:Bag>
</dc:date>
<dc:type> educational web-page</dc:type>
<dc:format>text/html</dc:format>
<dc:language>fi</dc:language>
</rdf:Description>
</rdf:RDF>
```

Kuva 9: Malli RDF-dokumentista.

Mitä keinoa sitten kannattaa käyttää metadataelementtien kuvauksen toteuttamiseksi? Koska HTML-kieli on vielä niinkin merkittävässä asemassa Wolrd Wide Webissä, sen käyttäminen metadataelementtien toteutukseen lienee luonnollisin ratkaisu. Lisäksi HTML:ää puoltaa sen helppokäyttöisyys ja syntaksin tutunomaisuus HTML:ää pitempään käyttäneille. XML ja RDF ovat ilmaisuvoimaisempia ja tehokkaampia, mutta samalla vaikeaselkoisempia ja hankalammin sovellettavissa olevia. Kuten kuvista 8 ja 9 voidaan varsin helposti päätellä, niin RDF-dokumentti on huomattavasti pitempi ja monimutkaisemman näköinen kuin vastaava HTML esitys. XML:ää ja RDF:ää voitaneen suositella käytettäväksi silloin, kun metadataelementtejä käsittelee pääosin tietokone, ei ihminen, jolloin niiden rakenteellisemmat ominaisuudet tulevat paremmin esille.

Tiettyjen kuvauskielten lisäksi metadataelementtien kuvaamiseen voi käyttää myös itse kehittelemiä periaatteita, jolloin voi rakentaa omiin tarkoituksiin parhaiten sopivan metadatan toteutuksen. Näistä edellä esitellyistä keinoista valitsisin HTML-kielen, sillä sen soveltaminen on helppoa ja sen käyttäminen World Wide Web ympäristöissä on varsin yleistä. Tosin suo-

siteltavaa on erottaa metadatatiedot tieto-olioista, koska esimerkiksi kuvien ja äänen metadataelementtejä ei voi lisätä tietoliikenteeseen itseensä. Kohdassa 4.3 esitetyn esimerkin tulen toteuttamaan täten HTML-kielen avulla.

3.4 Oppimisoliot

Miten oppimateriaaleista saadaan uudelleenkäytettäviä ja joustavia? Eräs mielenkiintoinen ratkaisu on oppimisolio- ja metadata-käsitteiden yhdistäminen rakenteeksi, joka teoriassa mahdollistaa uudelleenkäytettävän ja joustavan oppimateriaalin rakentamisen. Myös sopivien oppimateriaalien löytäminen ja toimiminen käyttäjän haluamassa ympäristössä ovat ongelmia, joihin oppimisolioiden toivotaan tuovan ratkaisuja (Robson 2000).

3.4.1 Määritelmä

Määrittelen *oppimisolion* (learning object) Bourdan ja Helierin (1999) sekä Currien ja Palacen (2000) tulkintojen pohjalta digitaalseksi tai ei-digitaalseksi kokonaisuudeksi, jota voidaan käyttää tai referoida minkä tahansa oppimistapahtuman aikana. Oppimistapahtuma voidaan ajatella prosessiksi, joka etenee oppimisolioiden muodostamaa polkua pitkin. Opetus voi olla perinteistä opetusta, etäopetusta, tietokoneavusteista opetusta tai verkkoympäristössä tapahtuvaa opetusta.

Oppimisolio ei siis ole pelkästään tietoverkkoihin tai teknologiaan kiinteästi yhteydessä oleva käsite, vaan siihen voidaan helposti liittää esimerkiksi perinteisiä oppimisvälineitä. Kuitenkin tutkimukseni luonteesta johtuen rajaan oppimisolion koskemaan pelkästään tietoverkossa olevaksi digitaalseksi olioksi. Tähän samaan päätyy IEEE-standardi LOM, joka edelleen rajaa oppimisolion käsitteen käytön koskemaan pelkästään teknologiaa hyödyntävää opiskelua (Markus 2000).

Oppimisolioiden tärkeimpiä ominaisuuksia on uudelleenkäytettävyys, joka mahdollistaa sen, että oppimateriaaleja tai oppimisympäristöjä ei aina tarvitse rakentaa täysin "tyhjästä". Oppimisolioita ovat edellä olevaan rajaukseen sisältyen muun muassa opetusohjelmat, aikataulut, oppimistavoitteet, luentomuistiinpanot, www-sivut, tehtävät, teksti, ääni ja simulaatiot (Bour-

da ja Helier 1999, Clark 1998). Uudelleenkäytettävyyden lisäksi oppimisoliot tarjoavat mahdollisuuden oppimateriaalin rakentamiseen erilaisten näkökulmien ja lähtökohtien pohjalta. Teoriassa oppimisolioiden avulla voidaan tehdä joustavia oppimateriaaleja, jotka pystyvät reagoimaan erilaisten oppilaiden tarpeisiin. Kohdassa 4.3 pyrin selvittämään, miten edellä kuvaillun mukaisen älykkään ominaisuuden toteutuksessa voidaan hyödyntää oppimisolioita.

Yhteenvetona oppimisoliosta voitaneen päätellä se, että oppimisolio on oppimiseen tai opiskeluun käytettävä kokonaisuus, joka voi sisältää itsessään myös muita oppimisolioita. Olioiden avulla pystytään rakentamaan oppimateriaaleja tai jopa kokonaisia oppimisympäristöjä, jotka tarvittaessa pystyvät mukautumaan erilaisten tarpeiden mukaan. Vaikka käytännön toteutukset oppimisolioiden avulla tehtyjen ympäristöjen kohdalla saattavatkin olla kaukana joustavista ja älykkäistä oppimateriaaleista, mielestäni oppimisolioiden idea antaa hyvän lähtökohdan kyseisten tavoitteiden saavuttamiseksi. Meisalo et al. (2000) suomentavat termin ”learning object” opetusolioksi, mutta mielestäni oppimisolio on vähintään yhtä hyvä suomenkielinen termi kuvaamaan kyseistä käsitettä.

3.4.2 Oppimisolioiden käyttö

Oppimisolioiden käytön avulla pyritään etsimään tehokkaampia keinoja oppimateriaalien kirjoittamiseen, ylläpitoon, jakamiseen ja etsimiseen. Oppimisolioiden avulla pystytään esimerkiksi identifioimaan tarkasti oppimateriaalin eri osat ja rakentamaan näiden osien välille monentyypisiä suhteita, jolloin voidaan rakentaa kohdan 3.1 mukaisia ontologioita (Bourda ja Helier 1999). Oppimisolio voidaan nähdä siis pienenä tietokokonaisuutena, joka joustavasti liitetään muokkautuvaan rakenteeseen. Muokkautuva rakenne voi olla esimerkiksi opiskelumuoduuli, dokumentti, tunti tai kurssi (Clark 1998).

On olemassa useita periaatteita, jotka käyttävät oppimisolioiden käsitettä oppimateriaalien rakentamiseksi. Meisalo et al. (2000) nimittävät oppimisolioiden käsitettä hyödyntävää suunnittelumallia oliomalliksi, joka korostaa oppimateriaalien rakentamisen tapahtuvan poimimalla joukko oppimisolioita, joilla on tietty oppimistavoite.

Eräs olioperustainen suunnittelumalli on Kassanken ja Suhlin (2000) yleinen kehys oppimateriaalien valmistamiseksi hypermediaympäristöön. Mallissa oppimisoliot samaistetaan legoi-

hin, joita voidaan käyttää rakennuspalikoina joustavan ja uudelleenkäytettävän oppimateriaalin tai oppimisympäristön toteuttamisessa. Myös Clark (1998) toteaa oppimisolioiden muodostavan isompia kokonaisuuksia, jotka voidaan tarvittaessa samalla tavoin identifioida olioksi ja yhdistää vuorostaan suurempaan kokonaisuuteen. Malli jakaantuu neljään osaan, jotka kaikki voidaan samaistaa tarvittaessa oppimisolioiksi:

1) *mediaelementti* on pienin mahdollinen kokonaisuus ympäristössä, esimerkiksi hyperteksti, animaatio, simulaatio, video, kuva ja Java-appletti.

2) *oppimiselementti* koostuu yhdestä tai useammasta mediaelementistä ja voidaan samaistaa tietyn kokonaisuuden tai aiheen osa-kokonaisuudeksi. Esimerkiksi World Wide Web-sivusto, joka sisältää animaatiota, kuvaa ja videoita.

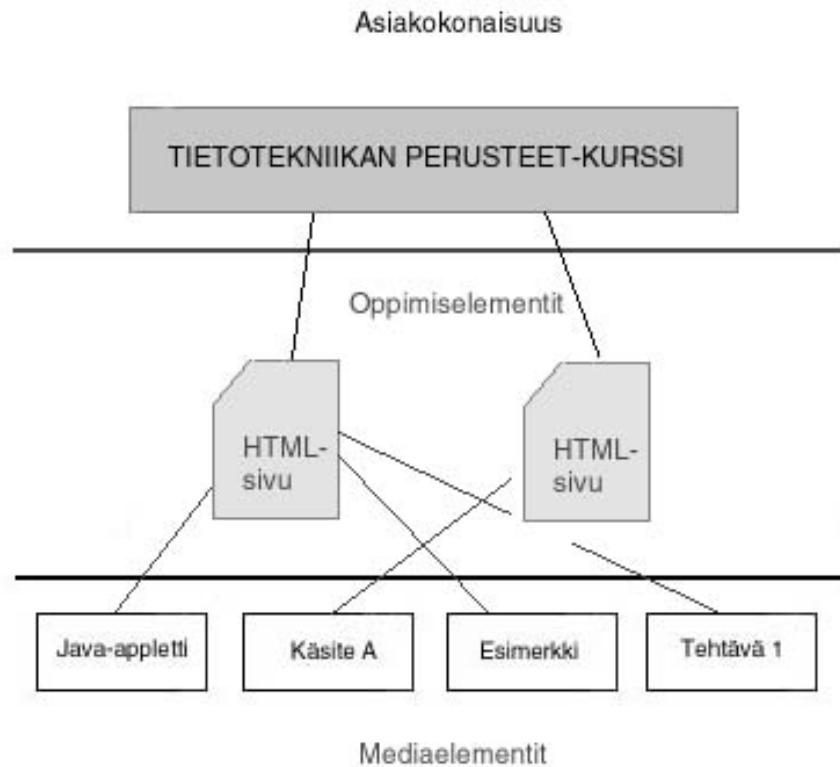
3) *asiakokonaisuus* muodostuu yhdestä tai useammasta oppimisoliosta, joka kuvaa tietyn oppikokonaisuuden. Esimerkkinä asiakokonaisuudesta voidaan mainita kurssi.

4) *metarakenne* määrittelee päälinjat siihen, kuinka asiakokonaisuuksien avulla rakennetaan tiettyjen oppimistavoitteiden kannalta relevantteja oppimateriaaleja. Metarakenne mahdollistaa yksilöllisten materiaalien kokoamisen, jotka ovat joustavia oppilaiden erilaisille tarpeille ja profiileille.

Kassanke ja Suhl (2000) painottavat nimenomaan metadatan merkitystä eri hierarkiatasoilla olevien elementtien välisten suhteiden kuvaamisessa. Metadata antaa lisäksi kuvauksen jokaisen olion sisällöstä, jolloin niitä voidaan käyttää joustavasti erilaisissa konteksteissa. Tämä esimerkki antaa mielestäni hyvän kuvan siitä miten oppimisolioiden käsite tukee älykkäiden ja joustavien oppimisympäristöjen rakentamista.

Kuvassa 10 on Kassanken ja Suhlin (2000) kehyksen mukaan rakennettu esimerkki oppimisolioiden käytöstä kolmella alimmalla tasolla, mistä nähdään eri elementtien muodostama kokonaisuus ja niiden väliset suhteet. Kuten kuvasta 10 voidaan päätellä oppimisoliot voivat esimerkiksi koulumaailmassa toimia seuraavan periaatteen mukaan: pienistä yksiköistä luodaan tunteja, tunnit vuorostaan muodostavat moduuleja, moduulit kurssseja ja kurssit opetus-

suunnitelmia. Teoriassa kurssien luojat pystyvät pelkästään tieto-olioita yhdistelemällä rakentamaan haluamiaan kokonaisuuksia, jotka luonnollisella tavalla tukevat muun muassa uudelleenkäytettävyyttä (Clark 1998).



Kuva 10: Kassanken ja Suhlin (2000) mallia kuvaava kaavio oppimisolioista.

3.4.3 Metadatan ja oppimisolioiden suhde

Kuten edellisessä kohdassa tuli esille, oppimisolioiden välille voidaan rakentaa erilaisia suhteita, joita voidaan kuvata esimerkiksi metadatan avulla. Markus (2000) käyttää termiä *opeuksellinen metadata* (educational metadata) sellaisten metadataelementtien yhteiseksi nimitykseksi, jotka pyrkivät kuvaamaan oppimisolioita. Suhteiden ja identiteetin kuvaamisen lisäksi sopivien oppimisolioiden löytäminen on hankalaa ja tähän metadatalle voidaan erityisesti vaikuttaa (Robson 2000). Metadatan avulla voidaan siis määritellä oppimisolion merkitystä ja suhdetta toisiin olioihin, joka helpottaa oppimisolioiden käyttöä tietyn oppimistavoitteen saavuttamiseksi.

IEEE:n Learning Object Metadata (LOM) on metadataesitys, joka on suunniteltu nimenomaan oppimisolioiden erityistarpeita varten. LOM onkin pohjana IMS-, ARIADNE- ja ADL-metadastandardeille, jotka sisältävät metadataelementtejä oppimisolioiden sisältöjen ja suhteiden kuvaamiseksi (Bourda ja Helier 1999, Currie ja Place 2000).

Vaikka oppimisoliot itsessään muodostavat rakennuspalikat verkostoituneen ympäristön rakentamiseksi, niin metadataa tarvitaan näiden olioiden yhteen liittämiseksi. Yhteistyön toteuttaminen olioiden välillä vaatii kattavan informaation olioiden ominaisuuksista ja käyttötarkoituksista. Tämän kuvailevan tiedon avulla oppimateriaalin tekijä voi liittää haluamansa oliot yhteen ja määritellä minkälaisessa suhteessa ne ovat keskenään (Markus 2000). Täten oppimisolioiden ja metadatan yhteispelillä voidaan rakentaa esimerkiksi toisistaan poikkeavia oppimispolkuja erilaisia tarpeita ajatellen.

Metadatan ja oppimisolioiden käytön suurimpia haasteita on saada olio ja sitä kuvaava metadata liittymään toisiinsa. Jos olioita ei ole pakattu tai linkitetty metadataan tarkoituksen mukaisin keinoin, niin menetetään tietoa olion käyttötarkoituksesta ja samalla edellä kuvatun kaltaisten rakenteiden muodostaminen käy erittäin vaikeaksi. Tämä tulee esille varsinkin monimutkaisissa materiaaleissa, joissa olioiden välisiä suhteita on käytännössä mahdotonta hahmottaa ilman metadatatietoa (Currie ja Place 2000).

Currie ja Place löytävät ongelmia myös metadataan linkittyneiden eri oppimisolioiden osalta, koska metadata voi sisältää linkin olioon, jota ei ole enää olemassa tai jonka paikka on muuttunut. Tämä ongelma on varsinkin tietoverkoissa erittäin ajankohtainen, sillä olioiden elinikä saattaa olla minimaalisen lyhyt. Tosin varsinaisen olion lisäksi myös metadatan paikka voi vaihtua, joka aiheuttaa vuorostaan hieman erityyppisiä ongelmia.

4. METADATA JA OPPIMISOLIOT ÄLYKKÄÄSSÄ OPPIMISYMPÄRISTÖSSÄ

Toisessa luvussa esittelin nykyisten oppimisteorioiden mukaan keskeiseen asemaan nousseen oppimisympäristön käsitteen ja erityisesti keskityin tietoverkoissa toimivan oppimisympäristön ominaisuuksiin. Kolmannessa luvussa tutustuimme yhteen tietoverkkojen mielenkiintoisimmista tiedon hakemiseen ja käyttöön liittyvistä sovelluksista eli metadataan ja oppimisoliioihin. Miten nämä kaksi periaatteessa toisistaan erillään olevaa asiaa voidaan yhdistää toimimaan yhdessä ja vielä älykkäällä tavalla? Vastausta lähdän etsimään tutustumalla niihin periaatteisiin, joiden pohjalta niin sanottuja älykkäitä oppimisympäristöjä toteutetaan. Lopuksi pyrin tuomaan esille sitä, miten metadataa ja oppimisoliota voisi hyödyntää älykkäiden oppimisympäristöjen rakentamisessa.

4.1 Älykäs oppimisympäristö

Kuten luvussa kaksi kävi esille, verkoissa tapahtuva opiskelu on tällä hetkellä erittäin suositua, mutta ongelmana on se, että verkko-opetuksessa käytettävät materiaalit ovat usein pelkkiä staattisia hypertekstejä (Brusilovsky 1999). Varsinkin vaikeana pidettyjen asioiden opiskelu ei välttämättä onnistu, jos kaikilla opiskelijoilla on käytössään yhtenäiset materiaalit. Tällaisten materiaalien ongelmana on se, että ne on suunniteltu aina tietyn tyyppisiä opiskelijoita varten, vaikka esimerkiksi opiskelijoiden taustat saattavat olla hyvinkin paljon toisistaan poikkeavia. Jotta verkko-opetuksessa käytettävissä materiaaleissa päästäisiin seuraavalle tasolle, niiden täytyy pystyä tarjoamaan tukea erilaisten oppilaiden oppimiselle. Tähän ongelmaan omalta osaltaan ratkaisun tuovat älykkäät ympäristöt, jotka pystyvät ottamaan huomioon erityyppisten opiskelijoiden vaatimuksia.

4.1.1 Mikä on älykäs oppimisympäristö?

Miten älykäs oppimisympäristö voidaan määritellä? Miten se eroaa "tyhmästä" oppimisympäristöstä? Kuten Chan et al. (1999) esittävät, on ideaalinen oppimisympäristö sellainen, jossa oppija on keskeisessä asemassa. Häntä ympäröi joukko opetuksellisia resursseja, joiden tarkoituksena on antaa virikkeitä ja aktiviteetteja oppimista varten. Tosin resurssien tarkoitukse-

na ei ole johdattaa opiskelijaa tarkasti laadittujen tiukkojen ohjeiden mukaan, vaan ohjata niissä tilanteissa, joissa oppilaille voidaan olettaa olevan eniten vaikeuksia.

Tietotekniikan tarjoamalla välineillä näiden resurssien käytöstä ja toiminnasta voidaan tehdä älykkään kaltaisia. Eräitä varteenotettavimmista vaihtoehdoista älykkään oppimisympäristön toteuttamiseksi ovat älykkäät tutorointijärjestelmät eli ITS (Intelligent Tutoring Systems), jotka voivat olla osana älykästä oppimisympäristöä. ITS:t käyttävät hyväkseen tietoa opetettava aiheesta, oppilaista, oppimisstrategioista ja oppimisympäristöistä tarjotakseen henkilökohtaista tutorointia oppilaille (Murray 1998). Taulukossa 6 on esitetty Brusilovskyn et al. (1998) esittämät toimenpiteet, joita tutor-opettaja voi tehdä auttaessaan oppilaita opetettavan aiheen opiskelussa. Heidän mielestään ITS-ohjelmiston täytyy tukea vähintään kolmea taulukossa esitellyistä ominaisuuksista, jotta se voidaan määritellä älykkääksi.

Taulukko 6: Tutor-opettajan seitsemän ominaisuutta (Brusilovsky et al. 1998).

Ominaisuudet:
1. Auttaa oppilaita hahmottamaan olennaisia asioita
2. Selittää oppilaille kuinka opetettavaa tietoa voidaan hyödyntää ongelmanratkaisussa
3. Antaa vinkkejä ongelmanratkaisumenetelmistä
4. Antaa vinkkejä ja korjauksia, jotta oppimistehtävät tulisivat ratkaistuksi
5. Tarjoaa relevantteja esimerkkejä kokemukseensa pohjautuen
6. Analysoi oppilaan virheet ja antaa palautetta
7. Ehdottaa seuraavan askeleen ottamista (miten kannattaa edetä opiskelussa)

Chan et al. (1999) puhuvat yksilöllisestä oppimisesta, kun edellä mainittujen resurssien avulla saavutetaan oppilaan omiin tarpeisiin reagoivia järjestelmiä. Eräs älykkään järjestelmän periaatteista on kerätä suuria määriä tietoja oppilaiden käyttäytymisestä oppimisympäristössä ja näiden tietojen perusteella tarjota tietyytyypisille oppilaille sopivia oppimisaktiviteetteja. Näillä perusteilla toteutettu oppimisympäristö voidaan luokitella älykkääksi oppimisympä-

ristöksi, koska se pystyy tarjoamaan ihmiskontaktia muistuttavan tavan oppimisen tukemiseksi.

Myös Niemirepo (1997) katsoo, että oppimisen sosiaalisia puolia ei ole riittävällä tavalla otettu huomioon verkko-opiskelussa. Älykkäillä järjestelmillä on täten mahdollisuus tuoda oppimisen sosiaalinen puoli paremmin esille. Sims (1999) korostaa tietokoneoppimisen tärkeimpinä tekijöinä sisällön ja oppijan sitoutumisen toisiinsa ja sitoutumisen aiheuttaman muutoksen oppijassa. Muutos voi olla tiedon oppimista, taidon osaamista tai tiedon hankkimista.

Edellisistä eri näkökannoista voidaan koota yhteen käsite "älykäs oppimisympäristö" seuraavasti: *älykkäästi toimiva oppimisympäristö* on tietokonepohjainen opetusjärjestelmä, jossa on erillisiä tietokantoja tai tiedonlähteitä opetuksen tueksi ja opetusstrategioiksi. (Ljubomir ja Vladan 1999). Näiden lisäksi järjestelmä pyrkii tekemään johtopäätöksiä oppilaan aiheen osaamisesta ja kehityksestä, jotta se pystyisi dynaamisesti muokkaamaan opetusta muuntuviin olosuhteiden mukaan.

4.1.2 Syyt mielenkiintoon älykkäitä oppimisympäristöjä kohtaan

Koulutuksen kustannukset lienee merkittävä tekijä kaikkialla maailmassa, esimerkiksi Yhdysvalloissa opetusministeriön ennusteiden mukaan opetus perusopetuksesta toisen asteen opetukseen saakka maksaa 232 miljardia dollaria vuodessa (Ljubomir ja Vladan 1999). Jokaisessa maassa ollaan kiinnostuneita tavoista, jotka laskevat koulutuksen hintaa. Luotetaan siihen, että uusien menetelmien kehittelyyn satsatut rahamäärät maksavat itsensä takaisin hyvin nopeasti.

Onkin ennustettu, että tietokoneen avulla tuetulla tai suoritettulla opetuksella on mahdollisuus vähentää opetusaikaa ja tuottaa paremmin oppineita ja harjaantuneita oppilaita. Tämä kuitenkin vaatii sen, että tietokonepohjaisten ympäristöjen on toimittava älykkäällä tavalla. Murray (1998) nostaa älykkään oppimisympäristön ominaisuuksista kaksi kaikkein tärkeimpään asemaan ajateltaessa nimenomaan oppilaiden oppimista. Ensinnäkin hänen mielestään oppilailla on mahdollisuus oppia paremmin, koska ympäristö mukautuu oppijan tarpeiden mukaisesti ja näin voidaan vähentää tietokoneiden mahdollisia negatiivisia vaikutuksia oppimisprosessiin.

Toisaalta oppimista parantaa se, että älykäs oppimisympäristö pystyy tarjoamaan palautetta oppilaiden tekemistä tehtävistä ja virheistä. Näin oppilaille annetaan paremmat mahdollisuudet rakentaa omaa tietämystään opetettavasta aiheesta virheistä ja vahvuuksista oppimalla.

Joidenkin tutkijoiden, kuten Bork (1997), näkemyksen mukaan pelkästään erittäin interaktiiviset, älykkäät tietokonepohjaiset oppimisympäristöt pystyvät tarjoamaan mielekkäitä oppimiskokemuksia. Siis itse asiassa, jos ajatuksenamme on rakentaa tietokonepohjaisia oppimisympäristöjä, niiden täytyy toimia älykkäällä ja joustavalla tavalla opiskelijoita kohtaan. Brusilovsky (1999) ennustaa älykkäiden oppimisympäristöjen olevan tulevaisuuden verkko-opetuksessa erittäin merkittävässä asemassa.

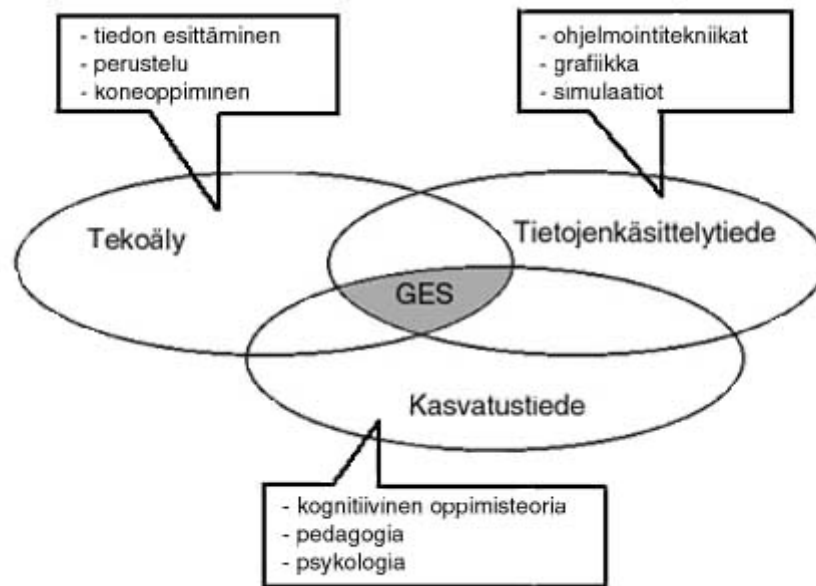
Verkko-oppiminen on vielä suhteellisen nuori ilmiö ja tällä hetkellä erittäin suosittua, jolloin ei-älykkäillä järjestelmillä on vielä kysyntää. Useat tutkimukset kuitenkin osoittavat, että älykkäillä järjestelmillä on merkitystä esimerkiksi periaatteessa itsenäisesti tapahtuvan verkko-opiskelun kannalta, joten tulevaisuudessa yhä enemmän mielenkiinto ja paine älykkäiden oppimisympäristöjen käyttöä kohtaan kasvaa. Brusilovsky (1999) näkee, että seuraavien viiden vuoden aikana tapahtuu huomattavaa kehitystä tällä rintamalla.

4.1.3 Älykkään oppimisympäristön ominaisuudet

Mitä mahdollisuuksia älykkäästi toimivalla oppimisympäristöllä teoriassa voisi olla? Älykkäästi toimiva ympäristö pystyy tarjoamaan melkein täydellisesti henkilökohtaisen oppimisprosessin jokaiselle oppijalle. Uutta opittua tietoa voidaan soveltaa omien kapasiteettien, odotusten ja aikaisemman tiedon perusteella, jolloin opiskelutilanteesta tulee henkilökohtainen, oppilaan ominaisuudet ihanteellisella tavalla huomioon ottava tapahtuma.

Sekä tietokoneohjelmien että tietokonepohjaisten oppimisympäristöjen kohdalla perimmäinen ongelma on se, mitä laadukas tietokoneohjelma voi tarjota opetuksellisessa mielessä. Ljubomir ja Vladan (1999) esittävät toimivan opetusohjelman (tai ympäristön) koostuvan kolmen eri tieteenalan kohtaamisen kautta. Nämä kolme alaa ovat kasvatustiede, tekoäly ja tietojenkäsittelytiede. Edellä mainittujen alojen kohdatessa voidaan puhua ratkaisuisista, jotka täyttävät hyvän opetusohjelman kriteerit (Good Educational System eli GES). Kuvassa 11 esitetään

kolmen eri tieteenalan risteyskohta, jossa kaikkien tieteenalojen erityisosaaminen on käytössä ja niiden avulla voidaan rakentaa GES-ominaisuudet täyttävä järjestelmä.



Kuva 11: Hyvän opetusohjelman ominaisuudet (Ljubomir ja Vladan 1999).

ITS:n käyttö älykkäässä oppimisympäristössä mahdollistaa aivan uusia tapoja tietokoneen käytölle opetuksessa, se voi esimerkiksi oikeasti toimia autenttisen oppimisprosessin kontrolloijana (Ljubomir ja Vladan 1999). Yksi nykypäivän tietokoneiden käyttöliittymästä puuttuva ominaisuus on aito kanssakäyminen ohjelmien ja käyttäjän kesken. Sims (1999) näkee interaktiivisuuden älykkäiden oppimisympäristöjen ominaisuutena, jonka kehittäminen tulee olemaan tärkeässä osassa tulevaisuudessa.

Interaktiivisuus

Bork (1997) tuo esille interaktiivisuuden eli kanssakäymisen tietokoneen ja käyttäjän välillä eräänä tärkeimmistä tulevaisuuden ympäristön ominaisuuksista. Interaktiivisuuden kautta päästään oppimateriaaleihin, jotka selvittävät opiskelijoiden tarpeet ja tarjoavat tarvittaessa apuaan sopivin keinoin. Ympäristöt pystyvät interaktiivisuuden avulla jatkuvasti tutkimaan oppilaan käyttäytymistä kartoittaen samalla kunkin oppilaan tarpeet ja vahvuudet, mikä mahdollistaa henkilökohtaisen tuen tarjoamisen. Itse termiä ”interaktiivisuus” käytetään kuva-

maan laaja-alaisesti kaikkia tapahtumia tietokoneen kanssa www-sivujen näppäilemisestä "ihmismäiseen" kanssakäymiseen (Sims 1999).

Bork (1997) puhuu erittäin interaktiivisista materiaaleista ja ympäristöistä silloin, kun ne pystyvät tarjoamaan henkilökohtaisia ongelmia ja aitoja oppimistilanteita. Esimerkiksi kahden henkilön reaalisessa elämässä tapahtuva jutustelu on erinomainen esimerkki interaktiivisesta oppimistilanteesta. Bork menee nykyisten tietokonelaitteistojen suorituskyvyn ääri rajoille väittäessään, että todellinen kanssakäyminen koneen kanssa tapahtuu pelkästään käyttämällä luonnollista kieltä. Näin äänen avulla tietokoneeseen saadaan ihmisläheisempi kontakti, jossa mahdolliset käyttöliittymästä aiheutuvat vaikeudet päästään sivuuttamaan. Näin ollen edellisessä kappaleessa kuvailtu pelkkien World Wide Web-sivujen linkkien ”naputtelu” ei liene varsinaisesti aitoa interaktiivisuutta.

Miten sitten määrittelemme nykyaikaan sopivan interaktiivisuuden, joka ei vielä ole Borkin (1997) kuvailemien periaatteiden tasolla eikä ole myöskään puuduttavaa hyperlinkkien seuraamista? Sims (1999) puhuu tietokoneohjelmien kohdalla interaktiivisuudesta silloin, kun käyttäjä kontrolloi opetuksen järjestelyä, rytmiä ja oppimateriaaleja. Nykyään ollaan vielä tilanteessa, jossa opetusjärjestelyjen suunnittelija ja toteuttaja päättävät esimerkiksi siitä, mitä käyttäjä katsoo tai lukee tietyssä tilanteessa. Interaktiivisuus mahdollistaa siis osaltaan seuraavaksi esiteltävän älykkään oppimisympäristön ominaisuuden eli joustavuuden.

Joustavuus

Kuten älykkään oppimisympäristön määritelmästä käy ilmi (kohta 4.1.1), eräs älykkään oppimisympäristön tunnuspiirteistä on sen pystyvyys dynaamisesti muuntautumaan muuttuneiden olosuhteiden mukaisesti. Tähän esimerkiksi ITS-ohjelmistot käyttävät tietokannoissa talletettua tietoa oppimisympäristössä olevista oppilaista ja heidän vahvuuksistaan (Murray 1998). Käytännössä tämä tarkoittaa sitä, että ympäristö voi tarjota yksityiskohtaisia keinoja tuoda opiskelijan saataville ihanteellinen oppimisaktiviteetti opiskelun eri vaiheissa. Joustavuuden ansiosta ympäristö pystyy ottamaan huomioon yksittäisten oppilaiden tarpeet tavalla, joka hyödyttää opiskelijaa parhaalla mahdollisella tavalla (Brusilovsky 1994).

Voinkin todeta, että nykykäsityksen mukaan edellä esitelty vahvasti interaktiivinen järjestelmä tarjoaa käyttäjälleen mahdollisimman suuren päätätävällän opetustapahtumista. Tähän

olisi siis pyrittävä suunniteltaessa oppimisympäristöjä, mutta siihen tavoitteeseen harvoin päästään. Vapaa päätäntävalta ei tosin ole kaikissa tapauksissa paras vaihtoehto, sillä tutkimukset osoittavat vapaan toiminnan ongelmallisuuden, kun kyseessä on noviisioppilaat. Kokeneemmat oppilaat osaavat käyttää vapauden antamia mahdollisuuksia, mutta aloittelijoille johdettu ja rajoitettu opiskelu saattaa sopia paremmin (Sims 1999). Tosin ongelmia voi syntyä siitä, etteivät opiskelijat – noviisit tai ekspertit – yksinkertaisesti hyväksy käytössä olevia opiskelumenetelmiä (Brusilovsky 1994).

Opetuksen rajoittamiseen liittyvä ongelma voidaan ratkaista esimerkiksi siten, että opiskelija voi milloin tahansa vaihtaa johdetusta tyylistä vapaampaan tai päinvastoin. Näin opiskelija voi kokeilla kumpi tyyli sopii paremmin hänelle ja kokemuksen karttuessa voidaan siirtyä vapaampiin muotoihin. Lisäksi käyttöliittymään liittyvät epäkohdat saattavat vaikeuttaa ympäristössä tehtävien suorittamista, joten käyttäjällä pitäisi olla mahdollisuudet muokata käyttöliittymää omien halujensa mukaisesti (Sims 1999). Käyttäjä voisi esimerkiksi valita, haluaako hän enemmän visuaalisuutta vai riittääkö hieman yksinkertaisempi esitystapa, joka sopii Brusilovskyn (1994) mukaan erityisesti kokemattomammille opiskelijoille.

4.1.5 Ympäristön rakentamisen ongelmat ja ratkaisuja ongelmiin

Kuten aikaisemminkin olen todennut, ovat älykkäästi ja joustavasti toimivat oppimisympäristöt vielä harvinaisia opetuksessa. Syitä tähän lienee useita, mutta esittelen seuraavaksi muutamia esimerkiksi Ljubomirin ja Vladanin (1999), Mizoguchin (1999), Frassonin et al. (1998) sekä Brusilovskyn (1999) esiin tuomia mielipiteitä älykkäiden oppimisympäristöjen toteuttamisen ongelmista.

Älykkään oppimisympäristön kehittäminen on erittäin kallis ja pitkäkestoinen prosessi. Jokainen järjestelmä joudutaan rakentamaan käytännössä alusta alkaen eli melkein tyhjältä pöydältä, mikä vaatii ison kehitystiimin ja paljon resursseja (Mizoguchi 1999). Vastaavasti uudelleenkäytettävyys ja uudet trendit ohjelmistotuotannossa ovat ottaneet ison roolin vähentäen kustannuksia. Lisäksi ITS:n toteuttamiseen käytettävät komponentit ja tiedot eivät ole jaettavissa saati sitten uudelleen käytettävissä, täten älykkäästi toimivien järjestelmien rakentamisen aloittaminen vaatii ylimääräisiä ponnisteluja.

Ljubomir ja Vladan (1999) näkevät ongelman myös siinä, että uusia järjestelmiä kehitellään käytännöllisesti katsoen ainoastaan tieteellisissä piireissä. Tämä aiheuttaa sen, että järjestelmiä hyödyntävien opettajien ja tutkijoiden välinen "kuilu" kasvaa. Tutkijat ovat kehitelleet jatkuvasti uusia järjestelmiä, mutta niiden käyttöönotto on ollut kehitystyötä hitaampaa. Esimerkiksi Brusilovskyn (1999) artikkelissaan esittelemistä älykkäistä järjestelmistä käytännöllisesti katsoen kaikki toimivat laboratorio-olosuhteissa, mutta niitä ei ole kertaakaan testattu todellisessa tilanteessa. Tämä kaikki aiheuttaa sen, että teoria ja käytäntö kulkevat täysin erilaisia polkuja ja tällä hetkellä useimmat etäopetukseen käytettävät järjestelmät eivät ole luonteeltaan älykkäitä (Brusilovsky 1999). Tosin tieteellisen tutkimuksen tulisikin näyttää suunta- viivoja tulevaisuudelle, mutta liiallinen erottautuminen todellisesta maailmasta ei liene pidemmän päälle kannattavaa.

Lisäksi Ljubomir ja Vladan (1999) kiinnittävät huomiota opettajien mahdollisuuksiin tehdä muutoksia käytettäviin järjestelmiin. Koska opetuksessa tarvittavat tiedot ja taidot voivat vaihdella jopa yhden maan sisällä, älykkään ympäristön käyttäjillä on oltava mahdollisuus muuttaa järjestelmän käyttäytymistä erilaisten tarpeiden mukaan. Ratkaisukeinona voisi olla opettajien saaminen mukaan järjestelmien kehittämiseen, jolloin he saisivat käytännön tasolla tietoa siitä, miten järjestelmää käytettäisiin kaikkein tehokkaimmin.

Edellä olevien näkökohtien lisäksi Frasson et al. (1998) haluavat painottaa älykkäiden oppimisympäristöjen rakentamisen vaikeutta, koska siihen tarvitaan aina hyvin paljon eri alojen ammattiosaamista (vertaa kuva 11). Älykkään oppimisympäristön toteuttamisessa ei riitä ainoastaan opettavan aiheen erityistietämys, vaan suunnittelijoilla ja tekijöillä täytyy olla varsin kattavat tiedot esimerkiksi erilaisista oppimisstrategioista, oppimisen psykologiasta, eli varsin syvälliset tiedot opettamisesta ja oppimisesta. Myöskään opetusmateriaalin luominen ei ole yksinkertaista, vaan senkin tekemiseen tarvitaan ammattitaitoa esimerkiksi käyttöliittymien alalta. Nämä kaikki edellä mainitut ja vielä useat muut tiedot ja taidot täytyy olla hallinnassa ennenkuin älykkäästi toimiva oppimisympäristö syntyy, joten sen rakentaminen vaatii paljon vaivannäköä.

Ratkaisuna älykkäiden ympäristöjen rakentamiseen Ljubomir ja Vladan (1999) esittelevät Edusoftin GET-BITS (Generic Tools for Building ITS)-mallin, jonka avulla pyritään helpottamaan ympäristöjen rakentamista. GET-BITS-mallin tavoitteena on kuvata käsitteellinen kehys, jolla voidaan esittää oppimisoliot, tapahtumat, vastuut, reaktiot ja suhteet, jotka tarvi-

taan tutoroinnin hoitamiseen. Toisena tavoitteena on toimivan käyttöliittymän rakentamisen nopeaan protoiluun ja helppoon opetussisältöjen luontiin, muokkaukseen, tuhoamiseen ja testaamiseen. GET-BITS-mallin periaate mahdollistaa omien tuntien rakentamisen ilman, että käyttäjän tarvitsee hallita ohjelmointia tai tekoälyn periaatteita.

Edellä esitelty GET-BITS malli on vain yksi monista periaatteista ja ohjelmista, jotka toimivat älykkäiden oppimisjärjestelmien tekijävälineenä. Frasson et al. (1998) luettelevat seuraavia tekijävälineitä, joilla pyritään helpottamaan älykkään ympäristön toteutusta: ID Expert, IDE, ISD Expert, GAIDA-projekti ja AGD. He itse esittelevät artikkelissaan rakentamansa CREAM (Curriculum Representation and Acquisition Model)-järjestelmän, jonka tavoitteena on esittää opetettava aihe mahdollisuuksien, opetuksen tavoitteiden ja pedagogisten resurssien avulla. Tätä tavoitetta varten käytetään esimerkiksi tehtäviä, demonstraatioita, ongelmia ja simulaatioita, jotka toimivat oppimisaktiviteetteina. Tässä esillä olevat järjestelmät käyttävät kohdassa 4.2 esiin tulevia älykkään oppimisympäristön toteuttamiseen tarkoitettuja periaatteita, mutta käyttäjän eli kurssin toteuttajan tehtävää on pyritty helpottamaan automatisoimalla tiettyjä toimenpiteitä ja vähentämällä esimerkiksi ohjelmoinnin tai tekoälyn tietämyksen merkitystä.

4.2 Älykkään oppimisympäristön toteuttaminen

Miten älykkäästi ja joustavasti toimiva oppimisympäristö voidaan rakentaa? Älykkään oppimisympäristön toteuttamiseen voidaan käyttää tekniikoita, jotka ovat osittain päällekkäisiä. Ympäristöt voidaan rakentaa käyttäen esimerkiksi yhtä tässä luvussa esiteltävistä periaatteista, mutta luultavimmin tehokkaimpaan tulokseen päästään, jos ne kaikki ovat jollain tavalla mukana ympäristössä. Esimerkiksi älykkäät agentit ja opiskelijamalli-rakenteet voivat toimia yhdessä siten, että älykkäät agentit käyttävät opiskelijamallia hyväkseen henkilökohtaisen palvelun tarjoamiseen.

4.2.1 Ontologiapohjainen arkkitehtuuri opetettavan tiedon esittämiseksi

Mizoguchi (1999) esittelee ITS:n komponenttien ja niiden välisten suhteiden rakentamista varten *ontologiapohjaisen arkkitehtuurin*, joilla pyritään ratkaisemaan opetettavan aineen ra-

kenteen pohjalta älykkään ympäristön toimintaa. Kuten kohdassa 3.1 määriteltiin, ontologia on kuvaus tietyn aihepiirin olioista ja niiden suhteista olemassa olevassa maailmassa. Ontologialla voidaan hahmottaa esimerkiksi tiettyyn oppimiskokonaisuuteen liittyvien erilaisten käsitteiden sisältöä ja suhteita. Murrayn (1998) mielestä älykkäissä oppimisympäristöissä on tärkeää se, että opetettava sisältö jaetaan modulaarisesti oppimisolioihin ja niitä voidaan siten käyttää eri tarkoituksiin. Opetettava aihe voidaan nähdä verkkona, josta selviää muun muassa oppimistavoitteet, aiheet ja niiden väliset suhteet.

Mizoguchi näkeekin ontologian ja siihen liittyvien käsitteiden merkityksen erittäin tärkeänä seikkana rakennettaessa oikeasti käyttäjän kanssa kommunikoivia älykkäitä oppimisympäristöjä. Ongelmana usein nykyisissäkin ITS-pohjaisissa ympäristöissä on se, että niillä ei ole eksplisiittistä esitystä opetettavan aiheen käsitteistä. Tämä johtaa siihen, etteivät ympäristöt ymmärrä yhdistää opetettavaa tietoa oikeisiin käsitteisiin, sillä on vuorostaan suora vaikutus siihen, miten järjestelmä keskustelee opiskelijan kanssa. Jos järjestelmällä ei ole oikeaa kuvaa opetettavan aiheen käsitteistä ja niiden välisistä suhteista, niin miten se pystyy kommunikoimaan käyttäjän kanssa käsitteiden tasolla. Vastauksena edellä olevaan on se, ettei kommunikoinnista tule aitoa.

Mizoguchi perää älykkäiden järjestelmien suunnittelijoilta teoriatietoisuutta eli opetettavan aineen teorian tuntemusta ontologian määrittelemiseksi. Tähän liittyy tiedon mallintaminen heurististen luokittelujen mukaan, joita on käytetty luotaessa muun muassa tietokantoja suurista asiakokonaisuuksista. Ideana on löytää aiheesta riippumaton malli, joka määrittelee aihepiirin olioiden roolit aiheeseen liittyvässä ongelmanratkaisussa, joka toimii oppimisen pohjana.

Mizoguchi (1999) jakaa ontologiat kolmeen tasoon, joiden tehtävänä on kuvata käsitteitä eri abstraktiotasoilla.

Taso 1. Joukko termejä, joiden tärkein tehtävä on artikuloida tutkittavaa maailmaa. Artikulointi tapahtuu määrittelemällä käsitteitä ja rakentamalla hierarkisia Is-a-suhteita niiden välille. Esimerkiksi: metadatakuvausten elementit. Tällä tasolla ei tehdä käsitteiden määrittelyjä.

Taso 2. Lisänä ensimmäiselle tasolle tulee käsitteiden formaaliset määrittelyt, joilla estetään väärinymmärrykset. Tarvittavat suhteet ja rajoitukset esitellään aksiomaattisesti. Suhteet ovat rikkaampia kuin ensimmäisellä tasolla ja määrittelyt ovat formaaleja, jotta tietokoneet voivat tulkita niitä.

Taso 3. Tämän tasoiset ontologiat ovat ajettavissa siinä suhteessa, että ontologia käyttää moduulien tuottamia abstrakteja koodeja, jotka ovat yhdistetty ontologiakäsitteisiin. Se voi vastata mallin ajonaikaiseen toimintaan liittyviin kysymyksiin. Tämän tasoisia järjestelmiä on harvassa.

Miten edellä olevia malleja voidaan käyttää rakennettaessa ontologiaan pohjautuvia järjestelmiä? Ensimmäisen tason mukaisia ontologioita voidaan käyttää sanastojen tapaan määrittelyssä esimerkiksi aihepiiriin kuuluvia käsitteiden nimiä. Toisen tason mukaiset ontologiat rakentuvat joukoista termejä ja niiden välisistä suhteista formaalien aksiomien avulla kuvattuna. Aksiomat toteutetaan deklaratiiivisesti ja niiden mukainen ontologia esittää oliot ja niiden väliset suhteet siten, että kyseinen ontologia voi toimia älykkyyden lähteenä oppimisympäristöissä (Mizoguchi 1999).

Ontologia voi toimia myös aiheen meta-mallina, joka on tietokoneella rakennettu abstraktio aihepiiristä. Ontologia tarjoaa käsitteet ja niiden väliset suhteet, jotka muodostavat usein meta-mallin eri osat. Aksiomat tarjoavat semanttisia rajoitteita käsitteiden sisällä, joten ontologia määrittelee rakennettavan mallin antamalla suuntaviivoja ja rajoitteita käsitteiden välillä (Mizoguchi 1999). Ontologian avulla voidaan rakentaa aiheen käsitteistä rakentuva hierarkkinen malli, joka toimii älykkyyden pohjana. Eri oppimisolioiden väliset suhteet voidaan rakentaa hyvinkin formaalilla ja abstraktilla tavalla, jolloin tietokone pystyy arvioimaan käsitteiden suhteita ja käyttämään näitä suhteita hyväkseen oppimisympäristössä. (Murray 1998). Näin saadaan esitettyä sekä itse tieto että siihen sisältyvää pedagogista informaatiota, joka on tärkeää esimerkiksi ongelmanratkaisun kannalta.

4.2.2 Opiskelijatiedot ja -malli

Yleisesti oppimisympäristöjen ja -materiaalien ongelmana on joustavuuden ja integraation puute. Joustavuudella tarkoitetaan sitä, että ympäristö mukautuu dynaamisesti opiskelijan

tarpeiden, kehityksen ja oppimisen mukaan. Integraatiolla tarkoitetaan puolestaan järjestelmän erilaisten komponenttien todellista yhteistyötä ja kommunikaatiota. Täydellisesti integroituneen järjestelmän komponenttien suorittamien toimenpiteiden vaikutusten pitäisi olla kaikkien muiden komponenttien saatavilla (Brusilovsky 1994). Esimerkiksi opiskelutehtävien tulokset voisivat olla järjestelmän kaikkien osien käytettävissä, jolloin oppilaan suoritukset vaikuttaisivat ympäristön käyttäytymiseen kyseistä henkilöä kohtaan.

Brusilovskyn (1994) mukaan *opiskelijamallin* keskeinen lähtökohta antaa mahdollisuudet ratkoa sekä joustavuuden että integraation ongelmaa. Mizoguchi (1999) kirjoittaa, että opiskelijamalli on järjestelmän älykkyyden lähde, jolla kuvataan järjestelmän tietämystä opiskelijoista ja heidän ominaisuuksistaan. Periaatteessa kaikki älykkään ympäristön komponentit voivat käyttää samaa opiskelijamallia muuntaakseen käyttäytymistään tietyn opiskelijan suhteen. Opiskelijamalliin voidaan tallentaa muun muassa opiskelijan kehittyminen opintojen aikana ja tarvittaessa älykkään ympäristön komponentit voivat päivittää opiskelijamallia opintojen edetessä. Tällöin opiskelijan kaikilla toiminnoilla on todellista merkitystä ja vaikutusta ympäristön toimintaan. Brusilovsky (1994) kuvaa edellä kuvattua tietojen muuntamisperiaatetta opiskelijamalli-silmukaksi (student modelling loop).

Myös Bork (1997) näkee oppilaiden tietojen tallentamisen yhdeksi tärkeimmistä keinoista rakennettaessa älykkäitä ympäristöjä. Hän laajentaa tietojen keräämisen kansainväliseksi toiminnaksi, jolloin järjestelmiin pystyttäisiin lisäämään käytännössä loputtomat määrät tietoa. Periaatteessa jokainen napin painallus tai liike voidaan tallentaa verkossa toimivassa oppimisympäristössä ja tallettaa palvelimelle, jolloin opiskelijan käyttäytymistä ympäristössä voidaan seurata hyvinkin tarkoin. Ajanmittaan oppilaista saadaan iso tietokanta erilaisia oppilasprofiiileja eli kuvauksia oppilaiden käyttäytymisestä ja oppimistuloksista pitkältä aikaväliltä. Näin järjestelmällä on paljon kokemusta erilaisista opiskelijoista ja niitä tietoja joko se tai suunnittelija voivat käyttää suunniteltaessa oppimisaktiviteetteja tuleville oppijoille (Chan et al. 1999).

Brusilovskyn (1994) kehittämässä ILE (Intelligent Learning Environment)-ympäristössä otetaan huomioon seuraavat opiskelijoiden ominaisuudet: henkilökohtaiset tekijät, kognitiiviset tekijät, opiskelustrategiat ja henkilökohtainen tietämys. Jokainen edellä mainittu elementti koodataan tiettyyn järjestelmän ymmärtämään muotoon, esimerkiksi opiskelustrategiatiedoilla voidaan hahmottaa tiettyyn tilanteeseen parhaiten sopivia oppimisstrategioita. Brusilovskyn

(1994) mielestä tärkeimmäksi ominaisuudeksi oppimisen kannalta nousee henkilökohtainen tietämys opetettavasta aiheesta, koska sen perusteella voidaan esimerkiksi aiheen jo osaaville opiskelijoille rakentaa mielekkäämpiä oppimiskokemuksia.

Seuraavaksi on lyhyt esittely Brusilovskyn (1994) käyttämästä tekniikasta ILE-ympäristössä. Ympäristössä on olemassa niin sanottu domain model -verkko, jonka solut vastaavat opetettavan aiheen elementtejä ja linkit vastaavat erilaisista suhteista elementtien välillä (pakollisuus, edeltävä). Opiskelijamallista löytyy overlay-mallin mukainen elementti, joka sisältää yhden laskurin jokaista aiheen elementtiä varten. Laskuri kuvaa opiskelijan ymmärtämystä kyseisestä elementistä. Laskuri voi saada siis esimerkiksi arvoja 1-5, jolloin 1=täydellisesti opittu ja 5=ei opittu lainkaan (tuntematon).

Edellisten lisäksi on käytössä kynnsarvo-tekniikka, jossa jokainen ILE-komponentti voi erottaa useita erilaisia tietotasojä jokaiselle tietoelementille eli overlay-mallin mukaisia tietoja. Yksinkertainen moduuli voi erottaa pelkästään kaksi tasoa: tunnettu tai tuntematon, kun kaikkein monimutkaisimmat moduulit voivat erottaa useita tasojä. Tietyn arvon siirtämisessä joukoksi tasojä jokainen moduuli voi käyttää kokonaislukuisia kynnsarvoja, jotka jakavat laskimen arvot vastaamaan moduulin tarvitsevia tietotasojä (vertaa edellinen kappale). Jokaisella moduulilla on oma joukko kynnsarvoja, jolloin nämä arvot voivat olla vaihdella erilaisten alojen ja opiskelijoiden mukaan (Brusilovsky 1994). Tämä mahdollistaa sen, että ympäristö voidaan muuttaa vastaamaan eri tasoisten kurssien ja oppilaiden tarpeita.

Esimerkiksi seuraavassa tilanteessa opiskelijamallista voisi olla hyötyä oppimisen tukena. Opiskelijan C on päätettävä opiskeleeko hän asian A vai asian B. Oppimisympäristö voi auttaa etsimällä tietokannasta toista opiskelijää, jolla on sama oppimishistoria kuin opiskelijalla C. Jos ympäristö päättelö, että asian B opiskelu oli hyödyllisempää toiselle opiskelijalle kuin asian A, niin se voi suositella opiskelijalle C asian B opiskelua. Jos tämä ajatus viedään pidemmälle, niin tulevaisuudessa on mahdollista toteuttaa opetussuunnitelmia, jotka perustuvat opiskelijan omaan mielenkiintoon, taustaan ja potentiaaliin (Chan et al. 1999).

4.2.3 Älykkäät agentit

Miten kohdassa 4.2.2 esitetyt opiskelijatiedot ja -mallit saadaan hyödynnettyä suunniteltaessa älykkäästi toimivia järjestelmiä? Eräs tapa on käyttää älykkäitä agenteja, jotka pystyvät prosessoimaan suuria tietomääriä ja toimimaan niiden pohjalta oppijan ohjaajana. Nduma ja Nwana (1996) määrittelevät *älykkään agentin* itsenäiseksi ohjelmaksi, joka voi kontrolloida omaa päätöksentekoaan ja käyttäytymistään tietyssä toimintaympäristössä eli ohjelmistossa. Agentilla on aina tarkasti määritelty tavoite, jota se pyrkii toteuttamaan kaikissa olosuhteissa. Agentit voivat esimerkiksi hoitaa palveluita, joita ympäristön käyttäjä tarvitsee.

Agenteilla on joukko ominaisuuksia, jotka ohjelmiston täytyy toteuttaa. Niitä ohjelmistoja, jotka eivät toteuta seuraavaksi esiteltäviä Nduman ja Nwanan (1996) määrittelemiä ominaispiirteitä, ei voida luokitella älykkäiksi agenteiksi. Ensinnäkin älykäs agentti on itsenäinen kokonaisuus eli se pystyy toimimaan ilman ihmisen panosta ja valvontaa sille asetettujen tavoitteiden mukaisesti. Sosiaalinen pystyvyys on toinen ominaisuus, joka yhdistetään usein nimenomaan älykkääksi piirteeksi. Älykkäällä agentilla on kyky olla inhimillisesti ja vakuuttavasti yhteistyössä muun muassa järjestelmää käyttävän ihmisen kanssa.

Kolmas merkittävä ominaisuus on reagointikyky, jota agentti toteuttaa tarkkailemalla ympäristöään ja reagoimalla erilaisiin tapahtumiin. Edelliseen liittyen älykäs agentti on myös aktiivinen eli tarvittaessa se pystyy tekemään aloitteen tietyissä asioissa. Viimeisenä ominaisuutena Nduma ja Nwana (1996) mainitsevat oppimiskyvyn, jonka ansioista agentti voi kehittää omaa tietämystään ja käyttäytymistään toimintaympäristön muutosten mukaisesti. Agentit voidaan jakaa toiminnallisuutensa mukaan erilaisiin agenttityyppeihin, joilla voidaan tarkemmin kuvata erilaisten agenttien toimintaa ja tarkoitusta. Taulukossa 7 on Nduman ja Nwanan (1996) esityksen mukainen jako erilaisista agenttityypeistä ja niiden toiminnasta. Jokaisesta agenttityypistä esitellään nimi ja lyhyt kuvaus niistä tehtävistä, joita kyseisen tyyppiset agentit pystyvät suorittamaan.

Taulukko 7: Nduman ja Nwanan (1996) esittämät agenttityypit.

Agenttityyppi:	Kuvaus:
Yhteistyöagentit	Tekevät laajaa yhteistyötä järjestelmän muiden tekijöiden, muun muassa toisten agenttien kanssa. Pystyvät oppimaan ja siten muuttamaan käyttäytymistään. Toimivat yleensä ympäristöissä, joissa paljon agenteja.
Käyttöliittymäagentit	Käyttöliittymäagentit tarjoavat apua ohjelmiston käyttäjälle. Käyttöliittymäagentit voivat myös oppia ja toimia täten käyttäjän henkilökohtaisen apurina.
Liikkuvat agentit	Liikkuvat agentit liikkuvat aktiivisesti World Wide Webissä käyttäen hyväkseen sen tarjoamia palveluja. Toimii yhteistyössä muun muassa erilaisten palvelimien kanssa.
Tietoagentit	Tietoagentit ovat aktiivisia, dynaamisia ja joustavia tiedon järjestäjiä. Voidaan käyttää esimerkiksi suurten tietomäärien käsittelyyn, jotta tietojen hyödyntäminen olisi helpompaa.
Ympäristönsä toimintaan reagoivat agentit	Tarkkailevat jatkuvasti ympäristöään ja tarvittaessa reagoivat nopeasti. Voivat toimia myös aktiivisina, jolloin niillä on kyky tehdä aloite tietyissä tilanteissa.
Hybridiagentit	Sisältävät vähintään kaksi edellä esitellyn agenttityypin ominaisuuksista.
Heterogeeniset agentit	Sisältävät vähintään kolmen eri agenttityypin tehtävät.

Edellä käytiin läpi yleisiä älykkäiden agenttien ominaisuuksia, mutta mitä erityispiirteitä opetustoimintaan tarkoitetut agentit sisältävät. Chan et al. (1999) määrittelevät opetuksessa käytettävät agentit älykkäiksi simuloiviksi agenteiksi, jotka ovat animoituja tai elävien kalta-

sia persoonallisuuksia. Nämä persoonallisuudet voivat olla kanssakäymisissä oppijan kanssa luonnollisella tavalla ja ne pystyvät tarjoamaan todentuntuisia oppimiskokemuksia. Agentit pystyvät ilmaisemaan tunteitaan ja mielialojaan, esittämään opittuja taitoja ja käyttämään ei-verbaalisia vihjeitä kiinnittääkseen oppijan huomion opiskeltavaan asiaan. Älykkäästi toimivat agentit voivat tuoda mukanaan muun muassa Niemirevon (1997) kaipaamaa sosiaalista kontekstia tietokoneella tapahtuvaan opiskeluun. Persoonallisesti toimivat agentit toimivat ympäristössä erilaisissa rooleissa ja ne pystyvät tarvittaessa muuntumaan muuttuneiden tilanteiden mukaan. Agentit voivat ottaa muun muassa seuraavia rooleja oppimisympäristössä: tutor, opiskelutoveri, kriitikko, joukkuekaveri ja opas (Chan et al. 1999).

Älykkäät agentit voivat suorittaa roolistaan riippuen hyvin erilaisia tehtäviä. Niemirepo (1997) esittelee erilaisia agenttityyppejä seuraavasti: viisaat agentit voivat etsiä tietoa elektronisista kirjastoista ja tarjoavat löytämäänsä tietoa opiskelijan käyttöön. Agentit voivat olla myös nimettömiä opiskelutovereita tai opettamisagentteja, jotka tarjoavat oppilaille tukea ja johdattelua sopivissa paikoissa. Agentit tietävät tarkoin miten ja mitä ohjaavat, joten oppimisympäristön varsinaiselta opettajalta jää aikaa enemmän yksittäisten, monimutkaisempien asioiden selvittämiseen.

Eräs agenttien tärkeimmistä piirteistä on mahdollisuus olla yhteistyössä eli kommunikoida sekä ihmisten että muiden älykkäiden ohjelmien kanssa. Agentti voi tehdä laajaa yhteistyötä ympäristön erilaisten osallistujien kanssa, olla aloitteen tekijänä ja toimia itsenäisesti tehden päätelmiä vallitsevasta tilanteesta (Niemirepo 1997). Älykkäästi toimivat ympäristöt voidaan rakentaa jopa siten, että jokainen järjestelmän komponentti on agentti.

Eräs tällaisten moniagenttisten ympäristön mielenkiintoisista kysymyksistä on agenttien välinen kommunikointi. Chen ja Mizoguchi (1999) toteavat, että agenttien välisen yhteydenpidon toteuttamiseen on kyllä kiinnitetty riittävästi huomiota, mutta vähemmälle on jäänyt kysymys itse viestien sisällöstä. Heidän mielestään ainoastaan opetettavasta aiheesta riippumattomat käsitteet ovat oikea ratkaisu agenttien välisen kommunikoinnin standardoimiseksi. Tämä on tietenkin toivottavaa, koska näin samoja agenteja voidaan käyttää hyödyksi erilaisten aiheiden opettamisessa. Myöskään pienet muutokset opetettavasta aiheesta eivät aiheuttaisi muutoksia agenttien toiminnassa, mikä olisi edessä, jos agentit keskustelisivat aina aiheesta riippuvilla käsitteillä. Niemirepo (1997) antaa moniagenttisesta ympäristöstä esimerkkinä GIA (Generic Institutional Architecture)-projektin, jonka tuloksena on syntynyt GIA-

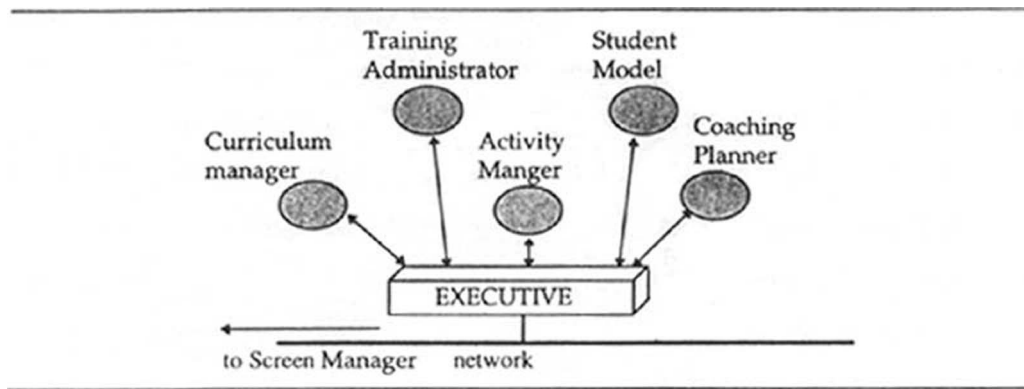
agenttipohjainen suunnittelumalli. Seuraava GIA-mallin esittely pohjautuu pitkälti Niemirevon tekstiin.

GIA-malli

GIA-mallin mukainen järjestelmä toimii koodattujen viestien eli KQML:n (Knowledge Query and Manipulation Language) avulla. KQML on sekä viestiformaatti että viestinkäsittelyprotokolla, jonka avulla agentit pystyvät tunnistamaan ja ottamaan yhteyttä sekä vaihtamaan tietoa toisten agenttien kanssa. Agenttien välistä kommunikaatiota on viime vuosina tutkittu erittäin paljon ja tunnetuimpia saavutuksia on juuri KQML-kieli (Chen ja Mizoguchi 1999). GIA-mallin suunnittelun pohjana toimii seuraava hypoteesi: ITS voidaan jakaa joukoksi itsenäisiä agenteja, jotka tekevät yhteistyötä ja vaihtavat tietoa käyttäen formaalia kieltä eli KQML:ää. Jakamiseen käytetään modulaarisuutta ja moduulien välillä on voimakasta yhteistyötä.

GIA-suunnittelun kriteereinä ovat järjestelmän liikuteltavuus, aiheriippumattomuus, laajennettavuus ja kehittyvät komponentit. Järjestelmän liikkuvuudella varmistetaan se, että GIA-mallin mukaiset järjestelmät ovat helposti siirrettävissä eri alustoilta toiselle. Aiheriippumattomuudella pyritään siihen, että ITS-suunnittelua voidaan hyödyntää mahdollisimman monilla alueilla. Laajennettavuus on erittäin tärkeää, koska sen ansiosta järjestelmään voidaan helposti liittää ohjelmistokomponentteja, esimerkiksi simulaatioita ja tiedonanalysointivälineitä. Viimeinen kriteeri on seurausta edellisistä, sillä järjestelmän komponenttien lisäämisen, muuntelun ja korvaamisen pitäisi olla mahdollisimman vaivatonta.

Kuten edellä olleista kappaleista käy selville, GIA-suunnittelumallin mukaisen järjestelmän jokainen komponentti on agentti, joka käyttää KQML:ää pitääkseen yhteyttä toisiin komponentteihin. GIA-malli jakautuu kahteen pääyksikköön, ytimeen (kernel) ja käyttöliittymään. Ytimen muodostamat agentit tarjoavat ITS:n tärkeimmät palvelut. Ytimen agenteja ovat: opiskelijarekisteri (Training Administrator), opetussuunnitelma ja tuntien suunnittelu (Curriculum Manager), päätösten avustaja (Coaching Planner) ja oppilasmalli/ongelmanratkaisu (Student Model). Käyttöliittymä toteutetaan joukolla käyttöliittymäagentteja. Agentit voivat olla visuaalisina esiintyminä näytössä, jolloin käyttäjä voi antaa niille toimintakäskyjä. Kuva 12 on kaavio ytimen toimintaperiaatteesta.



Kuva 12: GIA suunnittelumallin ydin (Niemi 1997).

4.2.4 Materiaalin esittäminen ja käyttöliittymän merkitys

Eräs mielenkiintoisimmista ja haasteellisimmista ongelmista kehitettäessä älykkäästi toimivia oppimisympäristöjä on se, miten samaan aiheeseen liittyvä oppimateriaalikokonaisuus esitetään oppilaalle henkilökohtaisella tavalla. Tutkimukset ovat osoittaneet, että esimerkiksi noviisiopiskelijat tarvitsevat enemmän selityksiä ja valaisevia esimerkkejä opiskelunsa tueksi kuin kokeneemmat opiskelijat (Brusilovsky 1994). Yksinkertaisesti ajatellen erilaisille opiskelijoille ei kannata tarjota samaa staattista oppimateriaalia, vaan materiaalin tulisi muuntua eri tilanteiden mukaan. Taulukon 6, sivulla 57 esittämistä tutor-opettajan ominaisuuksista joustavalla materiaalilla voidaan toteuttaa ydinasioiden esittäminen oppilaalle suotuisalla tavalla. Voidaankin ajatella, että ympäristö tarjoaa oppimateriaalin oppilaille siten, että oppimistavoitteiden mukainen polku oppimateriaalissa olisi aina mahdollisimman sopiva (Brusilovsky 1999).

Brusilovskyn (1999) mukaan opiskelijoille voidaan tarjota oppimateriaalia joko aktiivisella tai passiivisella tavalla. Aktiivisessa tavassa materiaalin tarjoamisessa on tietty oppimistavoite, johon päästään antamalla oppilaalle opetettavaan aiheeseen liittyvästä materiaalista osajoukko. Osajoukolla on määrätty oppimistavoite, jolloin osajoukkoja tarjotaan aina, kun asetettuun oppimistavoitteeseen on päästävää. Passiivisessa tyyliässä ei tarvita päämäärää eikä materiaaleja tarjota aktiivisesti, vaan materiaali annetaan opiskelijalle esimerkiksi silloin, kun hän ei pysty ratkomaan ympäristön tarjoamia oppimisongelmia. Tässä tapauksessa järjestelmä tarjoaa sen osajoukon oppimisolioita, joiden avulla oppilaan oletetaan ratkaisevan ongelman. Materiaali-

en erilainen tarjoaminen voidaan toteuttaa esimerkiksi CGI (Common Gateway Interface)-skripteillä, jotka mahdollistavat dynaamisten sivujen rakentamisen.

Materiaalin lisäksi myös opiskelijalle avautuva käyttöliittymä on oppimisympäristöissä usein samanlainen kaikille käyttäjille. Kuten edellä esitettyjen materiaalien tapauksessa, niin novii- si- kuin eksperttikäyttäjilläkkin on hyvin erilaiset vaatimukset ja tarpeet käyttöliittymän ominaisuuksista. Noviiikäyttäjät eivät halua kuvaruudulle mitään ylimääräistä ja esimerkiksi sivujen asettelun pitäisi olla mahdollisimman johdonmukaista. He antautuvat helpommin ympäristön ohjailtavaksi, kun vuorostaan kokeneet käyttäjät valitsevat mieluummin käytettävät aktiviteetit itse (Brusilovsky 1994).

Brusilovsky (1999) puhuu adaptiivisesta (adaptive) hypermediasta silloin, kun www-sivujen sisältöjä ja linkkejä muutetaan sekä oppilaan kehityksen mukaan että erilaisten oppilaiden kohdalla. Voidaan siis puhua adaptiivisesta esityksestä ja navigoinnista, joiden avulla pyritään tukemaan oppilasta tarjoamalla hänen kannaltaan sopivinta käyttöliittymää. Adaptiivisen navigaation toteutukseen Brusilovsky esittää kolme erilaista tapaa: suoran ohjauksen, joustavan linkkien selittämisen ja linkkien piilottamisen.

Suoran ohjauksen menetelmällä käyttäjälle kerrotaan, mitkä asiat tietyltä sivuilta johtavat parhaaseen tulokseen oppimisen kannalta. Linkkejä voidaan luoda dynaamisesti, jolloin eri tilanteita varten saadaan käyttöön erilaisia polkuja materiaalissa. Järjestelmä kertoo käyttäjälle, miten tavoitteeseen pääsee parhaiten ja ohjaa hänet sinne nopeasti. Opiskelija kyllä pääsee samaan paikkaan eri reittejäkin pitkin, mutta järjestelmä tarjoaa nopeinta tapaa. Parhaimman polun valitsemiseen käytetään esimerkiksi erilaisia heuristiikkoja, joiden avulla ympäristö voi arvioida sopivimman polun (Brusilovsky 1999).

Tietysti oppilaalla pitää on loppujen lopuksi valta päättää valitseeko hän järjestelmän tarjoaman polun vai käyttääkö kenties toista ratkaisua. Suoraa ohjausta hieman erilainen lähtökohta on joustava linkkien selittäminen ja piilottaminen (Brusilovsky 1999). Linkkien selityksiä muutetaan esimerkiksi oppilaan kehityksen mukaan. ELM-ART-, AST- ja InterBook-järjestelmät sisältävät ilmoitusjärjestelmän, joka antaa informaatiota jokaisen järjestelmän sivun pedagogisesta merkityksestä. Esimerkiksi, jos opiskelija ei ole valmis oppimaan kyseisen sivun sisältöä, niin ELM-ART- ja AST-järjestelmät lisäävät sivun alkuun tästä varoituk- sen. Näin oppilaalle annetaan vinkkejä siitä kannattaako hänen siinä vaiheessa opiskella ky-

seistä asiaa, vai jättää se myöhemmäksi. Piilottaminen tapahtuu siten, että käyttäjä ei pääse lainkaan käsiksi niihin sivuihin, joita hän ei ole järjestelmän mielestä valmis oppimaan. Edellisestä, joustavasta ilmoitustavasta poiketen käyttäjältä kielletään kokonaan eteneminen, jos hän ei ole siihen ympäristön mielestä valmis.

Sekä materiaalin tarjoamisessa käyttäjälle että adaptiivisessa käyttöliittymässä on tarkoitus muuntaa hypermediaoppimateriaaleja oppilaiden tavoitteiden, kehityksen sekä muiden opiskelijamallin tietojen perusteella. Ympäristössä, joka sisältää joustavan oppimateriaalin esityksen, sivut eivät ole pysyviä, vaan ne kootaan esimerkiksi oppimisolioista jokaista oppilasta tai oppilastyyppiä varten erikseen. Tämä mahdollistaa muun muassa sen, että noviisi- ja eksperttiopiskelijoille voidaan sekä opetustyyli että käyttöliittymä toteuttaa hyvin eri tavalla. Brusilovskyn (1999) mielestä tämä on erittäin toivottava piirre juuri World Wide Webissä, jossa "saman" sivun täytyy sopia hyvin erilaisille käyttäjille.

4.2.5 Ongelmanratkaisua tukevat välineet

Brusilovsky (1999) esittää kolme tekniikkaa ongelmanratkaisun tueksi älykkäällä tavalla. Ensimmäinen tekniikka on älykäs oppilaan ratkaisujen analysointi, jossa järjestelmä tekee päätelmiä oppilaan antamista ratkaisuista ja antaa niistä palautetta. Hieman erilaiseen ratkaisuun perustuu interaktiivinen ongelmanratkaisun tukeminen, jossa palautetta annetaan jatkuvasti, eikä pelkästään vastauksen antamisen jälkeen. Kolmas on esimerkkipohjainen ongelmanratkaisutuki, jossa oppilaat saavat esimerkkejä siitä, miten heidän ratkaisemiaan ongelmia on aikaisemmin ratkaistu. Nämä kaikki menetelmät pyrkivät auttamaan ongelmanratkaisua, mutta hieman eri tavoilla.

Älykkäässä oppilaan ratkaisujen analysointimenetelmässä järjestelmä käsittelee oppilaan lopullisia vastauksia. Järjestelmä selvittää, oliko annettu ratkaisu oikein vai väärin sekä mitä virheitä vastaus mahdollisesti sisältää. Tässä mallissa oppilas saa palautetta virheistään ja onnistumisistaan ja järjestelmä muuttaa opiskelijamallia jokaisen ympäristössä tehdyn ongelmanratkaisun jälkeen. Brusilovsky (1999) antaa esimerkkinä kyseistä menetelmää käyttävästä järjestelmästä PROUST-oppimisympäristön. Tämänkaltaiset mallit ovat yleisiä World Wide Web -ympäristössä, koska kanssakäyminen on passiivista ja se voidaan toteuttaa esimerkiksi CGI-kielellä.

Interaktiivinen ongelmanratkaisutuki-menetelmä on uudempi ja tehokkaampi kuin ratkaisujen analysointimenetelmä. Järjestelmä tarjoaa älykästä apua ongelmanratkaisun edetessä ja seuraa oppilaan tekemistä jatkuvasti. Se pyrkii aktiivisesti seuraamaan oppilasta ja yrittää ymmärtää tämän käyttäytymistä. Käyttäytymisen perusteella järjestelmä auttaa oppilasta ja tarvittaessa päivittää opiskelijamallia. Tätä mallia varten tarvitaan interaktiivinen järjestelmä, joka voidaan toteuttaa esimerkiksi Java-sovelluksilla. Brusilovsky (1999) mainitsee tätä mallia käyttävästä sovelluksesta esimerkkinä LISP-TUTOR:in Uusin menetelmä on esimerkkipohjainen ongelmaratkaisutuki-menetelmä, joka tarjoaa relevantteja, onnistuneita ongelmanratkaisukeinoja, jotka riippuvat oppilaan aikaisemmasta menestyksestä.

4.3 Metadata ja oppimisoliot älykkäässä oppimisympäristössä

Miten metadata ja oppimisoliot voivat sitten olla avuksi rakennettaessa kohdissa 4.1 ja 4.2 esitettyjä älykkäitä oppimisympäristöjä? Ensiksi esitän teoreettisen pohdinnan niistä mahdollisuuksista, joita oppimisolioilla ja metadataalla on toteutettaessa kohdan 4.2 mukaisia erilaisia tapoja toteuttaa älykkyyttä. Teoreettisen kehyksen jälkeen esittelen muutaman käytännön esimerkin, jotka osaltaan auttavat paremmin hahmottamaan teoriaa.

4.3.1 Oppimisolioiden merkitys älykkäässä ympäristössä

Kuten aikaisemmista kohdista on tullut esille, sopivien oppimisolioiden avulla voidaan rakentaa teoriassa hyvinkin monimutkaisia oppikokonaisuuksia. Miten sitten oppimisoliot kannatta toteuttaa? Oppimisoliokokonaisuuksien suunnittelu on tärkeä osa toteutusta, jotta oppimateriaali saataisiin jaettua sopiviin kokonaisuuksiin. Kohdassa 4.2.1 käsiteltiin ontologiapohjaisen suunnittelumallin etuja suunniteltaessa älykkäästi toimivia ympäristöjä. Sen mukaan oppimateriaalit rakennettaisiin mieluummin aiheesta riippumattoman ontologian perusteella, joka kuvaisi oppimateriaalin eri osat eli oliot ja niiden väliset suhteet formaalilla tavalla. Tällaisen ontologisen kuvauksen rakentaminen ei ole kovinkaan yksinkertainen prosessi, itse asiassa se vaatisi täysin oman tutkielmansa.

Varsinkin sellaisten aiheiden, joiden eri osien tai käsitteiden välillä ei ole monimutkaisia suhteita, oppimisoliot voidaan toteuttaa hieman erilaisten strategioiden mukaan. Lelouchen ja Moninin (1998) mukaan erilaisia tapoja mallintaa oppimateriaali ovat semanttiset ja käsitteelliset verkot, faktat, säännöt ja oliot. Heidän mielestään tärkeätä on tietää nimenomaan se, miten opetettavaan aiheeseen liittyvää tietoa tullaan käyttämään joko ympäristön tai opiskelijan toimesta, minkä perusteella voidaan määritellä esimerkiksi oppimisoliot ja niiden väliset suhteet. Esimerkiksi virtuaaliapprobatur-opintojen ”Tietokone, käyttöjärjestelmä ja tietoverkko”-kurssin oppimateriaali toteutettiin kurssilla käytetyn oppikirjan lukujen perusteella kahdeksaan eri osaan. Jokainen osa sisälsi aina yhden aihekokonaisuuden, esimerkiksi tietokonelaitteiston osat, joka puolestaan jakautui pienempiin osiin. Kuvasta 7 sivulta 46 löytyy esimerkki tietystä oppikokonaisuudesta eli oppimisoliosta, jonka avulla tutustutaan skannereiden ja digitaalikameroiden hienouksiin.

Näin muodostui joukko HTML-sivuja, jotka jaettiin kurssikirjan mukaisesti eri aihealueisiin. Aihealueet muodostivat viikkokokonaisuuksia ja viikkokokonaisuuksista tuli kokonainen kurssi. Emme katsoneet tarpeelliseksi mennä tämän tarkempaan jakoon, mutta periaatteessa oppimisolio voi olla vielä pienempi kokonaisuus, kuten määritelmä tai esimerkki. Luonnollisesti käyttämämme kuvat tai animaatiot ovat oppimisolioita, koska ne muodostavat itsessään kokonaisuuden, jolla on sisältö ja tarkoitus. Itse asiassa loimme tietämättämme oppimisolioita, jotka sisälsivät toisia oppimisolioita. Mutta ainoat suhteet oppimisolioiden välillä olivat PartOf-suhteita eli esimerkiksi kuva kuului tiettyä aihetta opettavaan kokonaisuuteen. Voidaan siis sanoa, että virtuaaliapprobatur-opintojen materiaali oli jaettu kurssikirjan antaman ohjenuoran perusteella oppimisoloihin, joiden välillä oli yksinkertaisia suhteita.

Älykkäästi toimivan oppimisympäristön kannalta oppimisoliot kannattaa muodostaa siten, että yksittäiset oppimisoliot sisältävät selvästi erottuvan yksittäisen kokonaisuuden, esimerkiksi määritelmän. Monimutkaisempia kokonaisuuksia voidaan sitten rakentaa yhdistelemällä pieniä oppimisolioita suuremmiksi kokonaisuuksiksi. Jokaisella oppimisoliolle olisi hyvä myös määritellä sen käyttötarkoitus eli mikä on oliion merkitys opetuksen kannalta (Lelouche ja Monin 1998).

Kun oppimisoliot on saatu määriteltyä ja niiden käyttötarkoitukset ovat selvillä, voidaan ryhtyä rakentamaan olioiden välille kuvauksia. Kuvauksien avulla voidaan päätellä, miten oppimisoliot suhtautuvat toisiinsa ja mikä on kuvauksen merkitys. Edellä kuvailussa esimerkissä

olioiden välillä oli vain PartOf-suhteita eli toinen olio kuului toiseen olioon, joista muodostui vähitellen aihekokonaisuuksia. Lelouche ja Monin (1998) käyttävät tästä menetelmästä nimitystä ”bottom-up”, jossa ensin määritellään kaikkien tärkeimmät ja perustavaa laatua olevat määritelmät yksittäisten oppimisolioiden avulla. Tämän jälkeen yksinkertaisista käsitteistä aletaan muodostamaan monimutkaisempia kokonaisuuksia, jolloin voidaan siirtyä esimerkiksi korkeampaan abstraktiotasoon.

Kuvauksia voi olla myös edellä kuvattua menetelmää monimutkaisempia, jolloin kuvauksen avulla voidaan kuvata esimerkiksi pakollisuutta, seuraavuutta, edeltävyyttä tai vapaaehtoisuutta. Kuvauksista voidaan rakentaa tietoverkko, joka kuvaa olioiden väliset suhteet ja niiden merkitykset eli kuvaukset (Lelouche ja Monin 1998). Tämä verkko voi sitten toimia pohjana rakennettaessa metadatakuvauksia oppimisolioista ja niiden välisistä suhteista.

4.3.2 Metadata oppimisolioiden kuvaamisessa älykkäissä ympäristössä

Kun oppimisoliot ja niiden väliset suhteet on määritelty, niin seuraava askel on rakentaa oppimisolioiden metadatakuvaukset. Minkälaisia elementtejä tällaisiin kuvauksiin voidaan ja pitää sisällyttää? Tietenkin sivulla 47 olevan kuvan 8 mukaiset yleiset tiedot, kuten oppimisolioiden tekijät ja tärkeät olion elinkaareen liittyvät päivämäärät, kuuluvat itsestään selvyytenä metadatakuvauksiin. Myös olioiden välisien suhteiden kuvaamiseen löytyy esimerkiksi Dublin Core -standardista elementtitarkentiminen, joten sitä voidaan käyttää kyseisen standardin (Dublin Core Metadata Initiative 1999) esittämällä tavalla kuvaamaan olioiden välisiä suhteita.

Koska Dublin Core-joukko on yleinen metadatastandardi, nimenomaan opetuksellisiin tarkoituksiin suunniteltuja elementtejä ei siitä löydy. Vastaavasti luvussa 3 esitetyt ARIADNE- ja IMS-standardit ovat suunniteltu opetuksellisia materiaaleja varten, mutta niiden käytössä tulee ongelmalliseksi ajoittainen monimutkaisuus ja pedagogisten elementtien todennäköinen sopimattomuus juuri älykkäiden ympäristöjen tarpeisiin. Ratkaisuna tähän ympäristön toteuttaja voi määritellä omiin tarkoituksiinsa sopivia elementtejä, joita voi lisätä esimerkiksi Dublin Coreen varsin joustavasti. Näillä omilla elementeillä oppimateriaalien tekijä voi lisätä metadatakuvauksiin haluamaansa informaatiota, jota voidaan käyttää hyväksi esimerkiksi toteutettaessa oppimateriaalin älykkäitä piirteitä.

Esimerkkinä pedagogisista tai älykkäiden toimijoiden tarvitsemista tiedosta voidaan mainita Kurhilan ja Sutisen (2000) esittelemä Ahmed-järjestelmä, jossa jokaisessa oppimisoliossa on tietty määrä parametrejä, joilla voidaan erottaa oppimisoliot toisistaan oppimisavaruudessa. Oppimisoliot voidaan siis kategorisoida tiettyjen ennalta määrättyjen arvojen mukaisesti. Tämän lisäksi oppilaita kuvataan samantyyppisillä parametreillä, jolloin oppilaan ”paikka” oppimisavaruudessa määräytyy kyseisten parametrien mukaan. Ympäristö tarjoaa oppilaille erilaisia aktiviteetteja eli oppimistapahtumia, joiden avulla oppilastietoihin talletettua informaatiota muutetaan oppimistulosten perusteella. Järjestelmässä toimivat älykkäät agentit eivät tiedä oppimisolioista mitään muuta kuin niiden parametrit ja ne pystyvät tarjoamaan opiskelijoille heidän tietojaan vastaava oppimisoliot opiskelun edetessä.

Metadatasesitykseen saadaan siis periaatteessa talletettua kaikki tarpeellinen pedagoginen tieto oppimisolioista ja niiden välisistä suhteista. Elementit voidaan rakentaa siten, että jokaisen oppimisolion kohdalla ei käytetä hyväksi kaikkia elementtejä, vaan tarpeen mukaan elementtejä voidaan jättää käyttämättä. Mahdollisia muutoksia varten metadatakuvauksia voidaan tarvittaessa päivittää esimerkiksi lisättäessä tiettyyn kokonaisuuteen uusia olioita tai päivittäessä uusia olioita vanhojen tilalle.

Metadata tarjoaa siis keinon tallettaa sitä tietoa, jota tarvitaan älykkään ympäristön toiminnan kannalta kuvaamaan oppimisolioiden ominaisuuksia ja käyttötarkoituksia. Älykkään oppimisympäristön eri osa-alueet voidaan toteuttaa siten, että ne hyödyntävät metadatatietoja omissa toiminnoissaan. Esimerkiksi kohdassa 4.2.3 esitellyt älykkäät agentit voisivat metadatakuvaukseen talletetun tiedon perusteella käsitellä oppimisolioita halutulla tavalla. Lisäksi metadataelementteihin pystytään liittämään samaa tietoa kuin opiskelijamalleissa, jolloin oppimisolion tietoja voidaan verrata opiskelijamallista löytyviin tietoihin.

4.3.3 Oppimisolioiden ja metadatan soveltaminen

Edellisten kohtien pohdinnan tueksi esitän esimerkin avulla oppimisolioiden ja metadatan soveltamista virtuaaliapprobatur-opinnoissa. Koska emme ole vielä käytännössä toteuttaneet älykkääksi luokiteltuja piirteitä, joudun kehittämään oppimisolioiden kuvaamiseen tarvittavat tiedot pitkälti keksittyjen tarpeiden mukaisesti. Eräs esimerkki oppimisolioihin liittyvistä tie-

doista löytyy edellä kuvatusta Ahmed-järjestelmästä, jossa tietyt oppimateriaalikonaisuudet arvotetaan joukolla parametrejä (Kurhila ja Sutinen 2000). Nämä parametreihin talletetut tiedot pystyttäisiin kuvaamaan myös metadatan avulla jokaisen oppimisolion kohdalla. Näin kyseinen metadatakuvaus olisi hyödynnettävissä esimerkiksi älykkäiden agenttien toimesta.

Koska HTML-formaatin käyttö on tutumpaa ja hieman yksinkertaisempaa, niin käytän sitä seuraavan esimerkin metadatakuvauksen toteuttamiseen. Kuva 13 perustuu luvussa kolme esitettyyn virtuaaliapprobatur-opintojen oppimateriaalisivun metadataesitykseen (kuva 8, sivu 47). Tätä lukua varten olen lisännyt kyseiseen kuvaukseen elementtejä, joita voivat muun muassa älykkäät agentit tai opiskelijamallit käyttää hyväkseen omassa toiminnassaan. Olen myös liittänyt kuvaukseen Dublin Core -standardin tarjoamat, olioiden välisiä suhteita kuvaavat elementit, joiden avulla saadaan tietoa muun muassa oppimisolioihin sisältyvistä muista oliosta. Kuvassa 10, sivulla 54 on esitetty miten oppimisolioiden avulla voidaan rakentaa suurempia kokonaisuuksia Suhlen ja Kassanken (2000) mallin mukaisesti.

```
<html>
  <head>
    <title>Skanneri ja digitaalikamera</title>
    <link rel = "schema.DC"
      href="http://purl.org/DC/elements/1.0/">
    <link rel = "schema.IC"
      href="http://cs.joensuu.fi/~appro/IC/">
    <meta name= "DC.Title"
      content = "Skanneri ja digitaalikamera">
    --- Tähän väliin kuuluvat esimerkin muut elementit ---
    <meta name = "DC.Language"
      content = "fi">
    <meta name = "DC.Source"
      content = "Tietotekniikan perusteet, sivut 17-79">
    <meta name = "DC.Relation.IsPartOf"
      content = "v2_index.htm">
    <meta name = "DC.Relation.References"
      content = "v2_1.htm">
    --- Tähän väliin voidaan lisätä kaikki ne sivut, joihin kuvailtavalla
    sivulta viitataan ---
    <meta name = "DC.Relation.References"
      content = "v2_linkit.htm">
    <meta name = "DC.Relation.HasPartOf"
      content = "\pics\small\6350c_mini.gif">
```

```
<meta name = "DC.Relation.HasPartOf"
      content = "\pics\small\digital_ixus.gif">
<meta name = "IC.Difficultlevel"
      content = "1">
<meta name = "IC.Graphics"
      content = "2">
<meta name = "IC.Strategy"
      content = "normal">
-- Tähän voidaan lisätä elementtejä tarpeen mukaan --
</head>
<body>...</body>
</html>
```

Kuva 13: Malli älykkyyden apuna käytettävästä metadatakuvauksesta.

Yllä oleva esimerkki on siis kuvassa 7, sivulla 46 olevan virtuaaliapprobatur-opintojen oppimateriaalisivun metadatakuvauksessa, johon on lisätty kuvan 8 kuvaukseen verrattuna muutamia elementtejä. Näiden elementtien tarkoituksena on antaa lukijalle käsitys siitä, miten metadatakuvauksia voidaan käyttää älykkäissä oppimisympäristöissä. Seuraavaksi tarkoitukseni on lyhyesti käydä läpi esimerkin 4 tärkeimmät kohdat, jotta lukija saisi mahdollisimman hyvän käsityksen elementtien merkityksestä ja käytöstä.

Koska Dublin Core -standardi on suunniteltu käytettäväksi mahdollisimman monen aiheen kanssa, kuvassa 13 on sekä Dublin Core -elementtejä että älykkyyden toteuttamiseen tarvittavia, tätä esimerkkiä varten kehiteltyjä elementtejä.

```
<link rel = "schema.IE" href="http://cs.joensuu.fi/~appro/IC/">
```

Yllä oleva rivi esimerkin alussa kertoo viitteen IC (Intelligent Content) määrittämiseen, jota käsitellään yksityiskohtaisemmin myöhemmin. Elementit voidaan suunnitella ja toteuttaa ilmenevien tarpeiden mukaisesti. Tätä esimerkkiä varten voidaan ajatella, että IC-elementtien kuvaus löytyy osoitteesta <http://cs.joensuu.fi/~appro/IC>. Sieltä IC-elementtien käyttäjät pystyisivät saamaan tarkempaa tietoa muun muassa siitä, miten elementtejä tulisi käsitellä.

Seuraavaksi kuvassa 13 on yleisiä Dublin Core -elementtejä, joilla ei ole älykkään ympäristön kannalta suurempaa merkitystä. Seuraava tämän esimerkin kannalta merkittävä Dublin Core -elementti on oppimisolioiden välisiä suhteita kuvaava DC.RelationIsPartOf-kuvaus.

```
<meta name = "DC.Relation.IsPartOf" content = "v2_index.htm">
```

DC.Relation.IsPartOf-elementti kuvaa sitä, että ”Skannerit ja digitaalikamera”-sivu kuuluu v2_index.htm-sivun esittämään ”Tietokonelaitteiston osat”-kokonaisuuteen. Tällä elementillä määritellään se, että ”Skannerit ja digitaalikamera”-sivu eli olio kuuluu suurempaan asiakokonaisuuteen, jolloin sen avulla voidaan tehdä oppimateriaalin rakenteeseen liittyviä päätelmiä.

```
<meta name = "DC.Relation.HasPartOf" content = "\pics\small\digital_ixus.gif">
```

Olioiden välisiä suhteita kuvaavat elementit loppuvat yllä olevaan riviin, joka puolestaan kuvaa sen, että ”Skanneri ja digitaalikamera”-sivuun kuuluu digital_ixus.gif-kuva. Verrattuna edelliseen kohtaan kyseinen elementti tuo siis esille sen, että kuvailtavaan olioon liittyy sekä loogisesti että fyysisesti toisia olioita.

Dublin Core -elementtien jälkeen tulee erityisesti älykkään ympäristön toimintaan liittyvät Intelligent Content -elementit. Tässä vaiheessa täytyy vielä korostaa sitä, että IC-elementit eivät ole todellisia, vaan niiden tarkoituksena on vain kuvata metadatan käytön mahdollisuuksia älykkään ympäristön toiminnassa. Selitän seuraavaksi tarkemmin kahden IC-elementin toimintaperiaatetta, joista voidaan päätellä myös mahdollisten muiden samantyyppisten elementtien käyttötarkoitusta ja merkitystä.

```
<meta name = "IC.Graphics" content = "2">
```

Yllä olevan elementin merkitys määritellään siten, että se kuvaa materiaalin grafiikkatasoa. Kyseistä elementtiä voidaan käyttää hyväksi esimerkiksi toteutettaessa joustavia oppimateriaaleja ja käyttöliittymiä. Elementin arvot voivat vaihdella esimerkiksi 1-4, jolloin 1 merkitsee graafisesti rikasta sivua ja 4 versiota, jossa on minimaalinen määrä grafiikkaa. Järjestelmä voi kyseisten arvojen perusteella näyttää joko opiskelijan haluaman version materiaalista tai ympäristön opiskelijamallista päättelämän, kullekin opiskelijalle sopivan version. Intelligent

Content -määrittelystä riippuu se, kuinka eri arvoja tulee käsitellä. IC.Graphics-elementti voi siis toimia kynnsarvona, joka määrää version sopivuudesta erilaisille opiskelijoille.

```
<meta name = "IC.Strategy" content = "information">
```

Tämän kaltaisen elementin avulla voidaan kertoa, minkä tyyppistä materiaalia kyseinen sivu oppimisen kannalta sisältää. Koska esimerkkinne sivu on staattinen eikä sisällä erityisiä komponentteja, se voidaan luokitella pelkästään informaatiota sisältäväksi. Tämä esimerkki ei parhaiten kuvaa kyseisen arvon merkitystä, koska aihepiiri on varsin yksinkertainen, jonka opiskeluun ei todennäköisesti tarvita mitään erityisiä keinoja. Mutta hieman monimutkaisempien asioiden opiskelua varten pystyttäisiin tekemään samasta aihealueista erilaisia versioita, joita voidaan tarjota dynaamisesti eritasoisille opiskelijoille. Muun muassa niille opiskelijoille, joille kyseinen aihe on täysin uutta tai jotka eivät ole sitä vielä sisäistäneet, esitettävä sivu voisi sisältää esimerkiksi ylimääräisiä esimerkkejä.

Kuva 13 ei ollut todellinen käytössä oleva metadatakuvaus, mutta mielestäni se antaa kuvan siitä, miten metadataelementtejä voidaan hyödyntää älykkäissä ympäristöissä. Metadataelementtien arvot olisivat siten kohdassa 4.2 esiteltyjen älykkyyttä toteuttavien toimijoiden käytössä. Esimerkiksi samaa metadataelementtijoukkoa voivat käyttää hyväkseen sekä älykkäät agentit että muut järjestelmän komponentit. Näin jokaisen olioon voidaan liittää joukko arvoja, joiden avulla järjestelmä pystyisi päättämään niiden käyttötarkoitusta ja merkitystä oppilaiden kannalta katsottuna.

5. TULEVAISUUDEN NÄKYMÄT

Edellisessä luvussa käsittelin sitä, miten metadataa ja oppimisolioita voidaan käyttää älykkäasti toimivassa oppimisympäristössä. Miten tästä pitäisi edetä? Mitkä voisivat olla tulevaisuudessa oppimisolioiden sovellukset? Miten metadata liittyy näihin sovelluksiin? Haluan näin tutkielman lopulla pohtia näitä kysymyksiä niiden mielikuvien perusteella, jotka ovat oppimisolioista ja metadatasta muodostuneet tämän tutkimusprosessin aikana.

5.1 Metadataelementtien sisällön määrittely

Kuten kohdan 4.3 esimerkistä käy selville, en ole tämän tutkielman parissa sen tarkemmin määritellyt minkälaista tietoa ”älykkäisiin” elementteihin pitäisi tallentaa. Eräänä mielenkiintoisena jatkotutkimuksen aiheena voisikin olla millaisia metadataelementtejä oppimisolioihin pitäisi liittää, jotta älykkään oppimisympäristön eri toimijat pystyisivät käyttämään niitä omissa toiminnoissaan mahdollisimman tehokkaasti. Tietysti tämä vaatisi sen, että aluksi kehitettäisiin ne periaatteet ja sovellukset, joilla älykäs ympäristö toteutetaan. Sen jälkeen oppimisolioihin voitaisi ruveta suunnittelemaan tarpeeseen vastaavia metadataelementtejä, jotka olisivat kaikkien oppimisympäristön komponenttien käytössä. Olisiko mahdollista kehittää esimerkiksi aiheesta riippumattomia elementtejä, jotka liitettäisiin oppimisolioihin pelkästään älykkäiden ominaisuuksien toteutusta varten?

Myös opettavien aihealueiden erityispiirteet asettavat omia vaatimuksiaan metadataelementtien sisältöön, vaikka periaatteessa aiheesta riippumattomien metadatakuvausten toteuttaminen olisi suotavaa. On kuitenkin ilmeistä, että ympäristön tehokkuuden kannalta liian yleiset kuvaukset eivät ole riittävän tehokkaita kaikkien aiheiden kannalta. Eräänä jatkotutkimuksen aiheena voisi olla esimerkiksi se, mitä erityispiirteitä voitaisiin lisätä ohjelmoinnin opetukseen liittyviin metadataesityksiin. Vaikka elementit kannattaa rakentaa älykkään ympäristön osatekijöiden tarpeiden mukaisesti, kuitenkin tehokkaan toimimisen kannalta myös tiettyyn aihekokonaisuuteen liittyvien metadataelementtien suunnittelu ja toteutus on tarkoituksenmukaista tietyissä tilanteissa.

Vuorostaan yksittäisten metadataelementtien luomisen automatisoiminen olisi myös erittäin mielenkiintoinen tutkimusprosessi. Kunze (1999) esittää Pearl-ohjelmointikielellä toteutetun automaattisen metadataelementtien luomiseen tarkoitetun ohjelman, joka pystyy kuvailtavaksi tarkoitettujen dokumenttien sisällön perusteella muodostamaan niihin liittyviä metadatakuvauskuksia. Etenkin suurten oppimateriaalien kohdalla ainakin osittain automaattinen metadatakuvausten toteuttaminen olisi paikallaan. Ongelmaksi muodostuvat erityisesti älykkääseen toimintaan suunnitellut elementit, joiden automaattinen luominen olisi peruselementteihin verrattuna huomattavasti vaikeampi prosessi.

Yhteenvetona edellisistä kappaleista voidaan todeta se, että tärkeää kaikenlaisten metadatakuvausten kannalta katsottuna on se, minkä strategioiden ja periaatteiden perusteella metadataelementtien sisällöt määritellään. Yleensä kannattaisi panostaa yksinkertaisesti ja lyhyesti toteutettuihin metadatajoukkoihin, mutta älykkään ympäristön tarpeet saattavat vaatia hyvinkin monimutkaisten ja pitkäköjien metadataesitysten toteuttamisen. Metadataelementtien toteuttajat joutuvat tasapainottelemaan näiden kahden yhtä tärkeän ominaisuuden välillä, jotta molempien näkökannat tulisivat mahdollisimman hyvin täytetyksi. Yksinkertaisten ja yksittäisten sivujen metadatakuvausten toteuttaminen onnistuu vielä varsin pienellä työpanoksella, mutta hiemankin monimutkaisten ja pitkien kuvausten luomiseen tarvitaan huomattavasti suurempi työmäärä. Tämän työmäärän optimointi on metadatakuvausten käytön yleistymisen kannalta erittäin tärkeää, joten siihen olisi syytä paneutua kunnolla.

5.2 Metadata oppimisympäristön eri tekijöiden kannalta katsottuna

Olen pitkälti tässä tutkimuksessa keskittynyt siihen, miten oppimateriaalien ja itse oppimisympäristöjen suunnittelijat ja toteuttajat pystyisivät hyödyntämään metadataa ja oppimisolioita. Miten muut oppimisympäristöön kuuluvat vaikuttajat pystyisivät hyödyntämään samaa metadatakuvausta kuin itse oppimisympäristö omissa toimissaan? Miten ylipäätään saadaan samaan kuvaukseen lisättyä eri osapuolien hyödynnettävissä olevia asioita? Myös nämä kysymykset ovat avainasemassa, jos metadataa ruvetaan hyödyntämään isossa mittakaavassa.

Esimerkiksi opetusta tarjoavan organisaation kannalta useasti käytettävät ja tarkasti määritellyt oppimisoliot ovat elintärkeitä, koska oppimisympäristöjen toteuttaminen on varsin kallista

toimintaa. Metadatan avulla kaikki tärkeimmät oppimisolioihin liittyvät tiedot saadaan kätevästi esitettyä niille tahoille, joiden tarkoituksena on käyttää hyväkseen olion tarjoamia palveluita. Täten oppimateriaaleja pystyttäisiin rakentamaan olioiden avulla ilman, että samat henkilöt vastaisivat sekä olioiden että oppimateriaalien toteuttamisesta. Tarkat metadatakuvaukset pystyvät tarjoamaan riittävästi tietoa oppimisolion käytöstä, jolloin esimerkiksi uudelleenkäytettävyydessä päästään aivan uudelle tasolle.

Oppimisympäristöstä löytyy myös muita toimijoita, joiden voidaan ajatella hyötyvän metadatakuvauksista. Esimerkiksi oppilaille voitaisiin tarjota metadatakuvausten perusteella tietoa minkälainen merkitys erilaisilla oppimisolioiden muodostamilla kokonaisuuksilla on juuri hänen oppimisensa kannalta. Näin oppilaat voisivat metadatatietojen perusteella päätellä tiettyjen asiakokonaisuuksien sopivuudesta omien lähtökohtien perusteella. Jos oppilas esimerkiksi toteaa, että oppikokonaisuus on hänen kannaltaan merkityksetöntä, se voitaisiin ohittaa ja siirtyä eteenpäin. Sama periaate pätee myös oppimisympäristön ohjaajan kohdalla, koska hän pystyisi rakentamaan oppikokonaisuuksia erilaisia oppimistavoitteita varten metadatakuvausten ja oppimisolioiden avulla. Tämä tulee esille sellaisissa ympäristöissä, joissa järjestelmä ei hoida kyseistä toimintaa.

Ongelmaksi tulee lähinnä se, miten kaikkien eri tekijöiden tarvitsevat tiedot voidaan liittää yhteen ilman, että metadatakuvauksista tulee liian raskaita ja tehottomia. Ei kuitenkaan ole toivottavaa, että osa metadataelementeistä olisi vain tietyn osa-alueen käytettävissä ja muille osille kyseisestä tiedosta ei olisi suurtakaan hyötyä.

5.3 Tulevaisuuden oppimateriaalit

Metadatan ja oppimisolioiden käytöstä tulevaisuudessa ei ole vielä tarkkaa kuvaa, mutta niiden ominaisuuksien maksimaalinen hyödyntäminen avaa visioita tulevaisuuden oppimisympäristöistä ja niissä käytettävistä materiaaleista. Tulevaisuuden oppimateriaalien voidaan ajatella koostuvan pelkästään oppiavaruudesta sijaitsevista oppimisolioista (Kuruhila ja Sutinen 2000). Oppimisolioita joko tarjotaan oppilaille älykkään ympäristön toimesta tai oppilas valitsee itse olioiden muodostaman kokonaisuuden opiskeltavaksi.

Esimerkiksi virtuaaliapprobatur-opintojen muodostama kokonaisuus voitaisiin joskus tulevaisuudessa toteuttaa metadatan ja oppimisolioiden avulla siten, että opetettavia asioita ei erotettaisi yksittäisiksi kursseiksi. Määriteltäisiin vain oppimisolioiden avulla muodostettuja asiakokonaisuuksia, joita oppilaat voisivat oman osaamisen ja mielenkiinnon mukaan opiskella. Ympäristö pystyisi päättämään esimerkiksi opiskelijamallin pohjalta, minkälaista tietoa oppilaille tulisi tarjota erilaisissa tilanteissa. Näin oppilaille pystyttäisiin tarjoamaan oppimispolkuja sen mukaan, mitä tietoja heiltä puuttuu esimerkiksi virtuaaliapprobaturkokonaisuudesta, jolloin oppilaiden tarpeet tulisivat parhaiten tyydytettyä.

Ihanteellinen tulevaisuuden näkymä olisi sellainen, että oppilaalla on tiedossa tavoite, johon hän pyrkii. Näin hän itse pystyy arvioimaan tarjottavan tiedon merkitystä oman oppimisensa kannalta ja valitsemaan oppimisavaruudesta sopivat oppimisoliot. Pitkälle toteutettuna kaikki oppilaitoksissa tarjottava tieto voitaisiin tarjota tällä keinoin, jolloin opiskelu etenisi oppilaan omien tavoitteiden mukaisesti kohden tiettyä päämäärää eli esimerkiksi opintokokonaisuuksia. Tämä tosin on varsin kaukainen visio verrattuna tämänhetkiseen tilanteeseen, mutta mielestäni oppimisolioiden ja älykkäiden ympäristöjen ominaisuudet mahdollistavat edellä esitetyn kaltaisen kehityksen joskus hamassa tulevaisuudessa.

6. YHTEENVETO

Olemme tutustuneet siihen, minkälaisia käyttömahdollisuuksia metadatalta ja oppimisolioilla on älykkäissä oppimisympäristöissä. Viidennessä luvussa pyrin esiteltyjen mahdollisuuksien pohjalta hahmottamaan tulevaisuuden oppimateriaalien ja oppimisympäristöjen rakennetta sekä toimintaperiaatetta. Tämän tutkielman puitteissa voinen todeta, että metadatan ja oppimisolioiden ominaisuudet mahdollistavat niiden käytön myös älykkäästi toimivien oppimisympäristöjen rakentamisessa. Vaikka älykkäästi toimivan oppimisympäristön rakentaminen on aikaa vievä ja kallis projekti, metadata ja oppimisoliot pystyvät osaltaan vähentämään ympäristöjen tekemiseen tarvittavien resurssien määrää.

Mitä sitten olisin voinut tehdä toisin? Ensimmäisenä mieleen tulee neljännessä luvussa esitelty metadatan ja oppimisolioiden soveltaminen, johon olisi voinut kehittää oikeasti toimivia esimerkkejä. Käyttämäni ratkaisu antaa lukijalle jonkinlaisen kuvan siitä miten metadataelementtejä voidaan soveltaa, mikä oli mielestäni riittävää tämän tutkielman kannalta. Hieman kritiikkiä voidaan varmaan esittää myös neljännen luvun lähteitä kohtaan, joita olisi voinut olla hieman enemmänkin. Peter Brusilovskyn artikkelit ovat neljännessä luvussa ehkä liiankin keskeisessä asemassa, joten useampien näkökulmien esiin tuominen olisi ollut aiheellista.

Näiden kohtien lisäksi ongelmalliseksi tutkielman tieteellisyyden kannalta nousevat esiin varsin useat konferenssijulkaisut. Puolustukseksi voinen sanoa sen, että varsin monet konferenssijulkaisut olivat suhteellisen uusia, jolloin niistä voidaan katsoa löytyvän varsin ajankohtaista tietoa tutkielmassa käsiteltävistä asioista. Huolimatta edellä luetelluista puutteista, olen varsin tyytyväinen lopputulokseen. Etenkin kolmanteen lukuun voinen olla erittäin tyytyväinen, koska mielestäni luku muodosti kokonaisuudessaan hyvin yhtenäisen kokonaisuuden.

Lähdeluettelo

Ahronheim, J.R.: Descriptive metadata: Emerging standards. *Journal of Academic Librarianship*, **24**(5):395-404, 1998.

Alpert, E., Bland, C.G., Carswell, L., Citzmar, D., DeWitt, J., Dumitru, M., Fahraeus, E.R., Lawhead, P.B. & Scott, K.: The web and distance learning: What is appropriate and what is not. *SIGCUE Outlook*, **25**(4):27-37, 1997.

Anderson, T. & Wason, T.: IMS learning resource metadata information model: final specification version 1.1. IMS Global Learning Consortium, INC., 2000a. Internet WWW-sivu, URL: <http://www.imsproject.org/metadata/mdintov1p1.html> (21.10.2000).

Anderson, T. & Wason, T.: IMS metadata best practice and implementation guide: final specification, version 1.1. IMS Global Learning Consortium, INC., 2000b. Internet WWW-sivu, URL: <http://www.imsproject.org/metadata/mdbestv1p1.html> (21.10.2000).

ARIADNE. Educational metadata recommendation, Versio 3.0, 1999. Internet WWW-sivu, URL: <http://ariadne.unil.ch/Metadata> (21.10.2000).

Auer, A. & Pohjonen, J.: Kohti uusia oppimisympäristöjä. Teoksessa Collan, J., Kari, J., Karjalainen, M.(toim.) & Pohjonen, J. *Teknologia koulutuksessa*. WSOY, Juva, 1995.

Bork, A.: The Future of computers and learning. *THE Journal*, **11**(24):69-78, 1997.

Bostock, S.J.: Design web-based instruction for active learning. Teoksessa Khan, B.H. *Web-based instruction*. Educational Technology Publications, New Jersey, USA, 225-230, 1997.

Bourda, Y. & Helier, M.: Applying IEEE learning object metadata for publishing teaching programs. Proceeding of ED-MEDIA, World conference on educational multimedia, hypermedia & telecommunication, June 19-26, Seattle, Washington, USA, 1999.

Brickley, D., Miller, E. & Miller, P. (toim.): Guidance on expressing the Dublin Core within the Resource Description Framework (RDF). Dublin Core Metadata Initiative, 1999. Internet WWW-sivu,

URL: <http://www.ukoln.ac.uk/metadata/resources/dc/datamodel/WD-dc-rdf-19990701.html> (22.10.2000).

Brown, A.: Designing for learning: What are the essential features of an effective online course? *Australian Journal of Educational Technology*, **13**(2):115-126, 1997.

Brusilovsky, P.: Student model centered architecture for intelligent learning environments. Teoksessa: *Proceedings of Fourth international conference on user modelling, 15-19 August, Hyannis, MA, USA*, 31-36, 1994.

Brusilovsky, P.: Adaptive and intelligent technologies for web-based education. Teoksessa Peylo, C. & Rollinger, C. (toim.). *Künstliche intelligence, Special issue on intelligent systems and teleteaching, Vol.4*, 19-25, 1999.

Brusilovsky, P., Medvedeva, O. & Ritter, S.: Creating more versatile intelligent environments with a component-based architecture. Teoksessa: Goettl, B.P., Half, H.M., Redfield, L.L. & Shutle, V.J. *Lecture notes in computer science, vol. 1452*. Springer-Verlag, Berlin, 554-569, 1998.

Carswell, L.: The "Virtual University": Toward an Internet paradigm. *Proceedings of the 6th annual conference on the teaching of computing / 3rd annual conference on integrating technology into computer science education on changing the delivery of computer science education*, August 18-21, Dublin, Ireland, 46-50, 1998.

Chan, T.-W., Hue, C.-W. & Tseng, O.J.L.: Space of active social learning models. Teoksessa: Cumming, G., Gomez, L. & Okamoto, T. (toim.). *Advanced research in computers and communication in education*. IOS Press, Amsterdam, 53-60, 1999.

Chen, W. & Mizoguchi, R.: Communication content ontology for learner model agent in multi-agent architecture. Teoksessa: Cumming, G., Gomez, L. & Okamoto, T. (toim.).

Advanced research in computers and communication in education. IOS Press, Amsterdam, 95-102, 1999.

Clark, R.: Recycling knowledge with learning objects. *Training & Development*, **52**(10):60-61, 1998.

Currie, D. & Place, G.: Learning objects containers: A suggested method for transporting metadata with a learning object. *Proceedings of ED-MEDIA, World conference on educational multimedia, hypermedia & telecommunication*, June 28-July 1, Montreal, Canada, 2000.

Dominique, J., Motta, E. & Schumm, S.B.: Representing scholarly claims in Internet digital libraries: A knowledge modelling approach. *Proceeding of EDLD: Third European conference on resume and analyses technology for digital libraries*, September 22–24, Paris, France, 1999.

Dublin Core Metadata Initiative. Dublin Core metadata element set, version 1.1: reference description, 1999. Internet WWW-sivu, URL: <http://purl.org/dc/documents/rec-dces-19990702.htm> (18.10.2000).

Dublin Core Metadata Initiative. Dublin Core qualifiers, 2000. Internet WWW-sivu, URL: <http://purl.org/dc/documents/rec/dcmes-qualifiers-20000711.htm> (30.10.2000).

Dybdahl, A., Sutinen, E. & Tarhio, J.: On animation features of Excel. *Proceedings of the 6th annual conference on the teaching of computing / 3rd annual conference on integrating technology into computer science education on changing the delivery of computer science education*, August 18-21, Dublin, Ireland, 77-80, 1998.

Facer, K., Furlong, J., Furlong, R. & Sutherland, R.: A new environment for education? The computer in the home. *Computers & Education*, **34**(1):195-212, 2000.

Farquhar, J.D. & Jones, M.G.: User interface design for web-based instruction. Teoksessa Khan, B.H. *Web-based instruction*. Educational Technology Publications, New Jersey, USA, 239-244, 1997.

Feldman, S. & Milstead, J.: Metadata. *Online*, **23**(1):24-30, 1999a.

Feldman, S. & Milstead, J.: Metadata projects & standards. *Online*, **23**(1):32-39, 1999b.

Frasson, C., Gauthier, G. & Nkambou, R.: A new approach to ITS-curriculum and course authoring: the authoring environment. *Computers & Education*, **31**(1):105-130, 1998.

Granger, M.J. & Passerini, M.: A developmental model for distance learning using the Internet. *Computers & Education*, **34**(1):1-25, 2000.

Greening, T.: WWW support of student learning: A case study. *Australian Journal of Educational Technology*, **14**(1):49-59, 1998.

Griffin, S. & Wason, T.: The year of metadata. *Educom Review*, **32**(6):56-58, 1997.

Hara, N. & Kling, R.: Students' distress with a web-based distance education course. CSI-working paper, 2000. Internet WWW-sivu, URL:
http://www.slic.indiana.edu/CSI/wp00_01.html (15.11.2000).

Heery, R.: Review of metadata formats. *Program*, **30**(4):345-373, 1996.

Hoffman, B. & Ritchie, D.C.: Incorporating instructional design principles with the World Wide Web. Teoksessa Khan, B.H. *Web-based instruction*. Educational Technology Publications, New Jersey, USA, 135-138, 1997.

Karjalainen, A.: Structured hypercourseware for distance learning. *Proceedings of Enable -97 conference*, May 28-30, Espoo, Finland, 1997.

Kassanke, S & Suhl, L.: OR-World – using objects in a hypermedia learning environment. *Proceedings of ED-MEDIA, World conference on educational multimedia, hypermedia & telecommunication*, June 28-July 1, Montreal, Canada, 2000.

Kunze, J.: Encoding Dublin Core metadata in HTML. Dublic Core Metadata Initiative Memo, 1999. Internet WWW-sivu, URL:

http://www.mirror.ac.uk/sites/ftp.isi.edu/in_notes/rfc2731.txt (30.10.2000).

Kurhila, J. & Sutinen, E.: From intelligent learning systems to intelligent learning material. *Proceedings of ED-MEDIA, World conference on educational multimedia, hypermedia & telecommunication*, June 28-July 1, Montreal, Canada, 2000.

Lelouche, R. & Monin, J.-F.: Introduction of pedagogical concepts in domain modelling for an intelligent tutoring system. *Journal of Engineering Education*, **23**(2):255-272, 1998.

Levi, A.: Learning technology is not only "pushing buttons". *International Journal of Instructional Media*, **21**(3):255-260, 1994.

Lifländer, V-P.: *Yhteistoiminnallinen projektioppiminen verkossa*. Oy Edita Ab, Helsinki, 1999.

Ljubomir, J. & Vladan, D.: The role of the teachers in intelligent tutoring and authoring tools, the GET-BITS model of EduSoft. Teoksessa: Cumming, G., Gomez, L. & Okamoto, T. (toim.). *Advanced research in computers and communication in education*. IOS Press, Amsterdam, 768-775, 1999.

Lloyd, A.D.: Pedagogy vs. competition in higher education distance learning. *Educational Technology & Society*, **3**(2):23-35, 2000.

Markus, B.: Educational metadata. *Proceedings of Quo Vadis-International Conference, FIG Working Week*, May 21-26, Praha, Tsekki, 2000.

Maureen, T.: Constructivism, instructional design and technology. *Educational Technology & Society*, **3**(2):50-60, 2000.

Meisalo, V., Sutinen, E. & Tarhio, J.: *Modernit oppimisympäristöt*. Tietosanomat Oy, Juva, 2000.

Messing, J.: The price a distance education student pays when using electronic learning resources. *Proceedings of ED-MEDIA, World conference on educational multimedia, hypermedia & telecommunication*, June 26-July 1, Montreal, Canada, 2000.

Miller, G.E.: Distance education and the emerging learning environment. *Journal of Academic Librarianship*, **23**(4):319-321, 1997.

Mizoguchi, R.: Ontology-awareness for intelligent instructional systems. Teoksessa: Cumming, G., Gomez, L. & Okamoto, T. (toim.). *Advanced research in computers and communication in education*. IOS Press, Amsterdam, 45-54, 1999.

Murray, T.: Authoring knowledge based tutors: Tools for content, instructional strategy, student model and interface design. *Journal of the Learning Sciences*, **7**(1):5-64, 1998.

Nduma, D. & Nwana, H.: Research and development challenges for agent-based systems. *IEEE/BCS Software Engineering Journal*, 1996.

Niemirepo, T.: Designing agent-based learning environments. Teoksessa: Väliharju, T. & MLE-group (toim.). *Digital media as a learning environment*. Tampereen yliopisto, Tampere, 51-66, 1997.

Norris, P. & Russel, S.: *Artificial intelligence: A modern approach*. Prentice-Hall, USA, 1995.

Pantzar, E.: WWW in the media world: Basic principles for WWW-based learning environment design. Teoksessa Väliharju, T. & MLE-group (toim.). *Digital media as a learning environment*. Tampereen yliopisto, Tampere, 67-83, 1997.

Perkins, D.N.: Technology meets constructivism: Do they make a marriage? *Educational Technology*, **31**(5):18-23, 1991.

Robson, R.: Report on learning technology standards. *Proceeding of ED-MEDIA, World conference on educational multimedia, hypermedia & telecommunication*, June 28-July 1, Montreal, Canada, 2000.

Sims, R.: Interactivity and narrative: strategies for effective learning. *Proceedings of ED-MEDIA, World conference on educational multimedia, hypermedia & telecommunication*, June 19-26, Seattle, Washington, USA, 1999.

Swick, R.R.: RDF: Weaving the web of discovery. *Networker*, **3**(2):21-25, 1999.

Tomak, I.: Virtual Environments in Education. Teoksessa Cunning, G., Gomez, L. & Okamoto, T. (toim.). *Advanced research in computers and communications in education*. IOS Press, Amsterdam, 3-10, 1999.

Weibel, S.: The Foundations of resource description. *DLib Magazine*, July 1995. Internet WWW-sivu, URL:
http://mirrored.ukoln.ac.uk/lis_journals/dlib/dlib/dlib/july95/07weibel.html (17.10.2000).

Wilson, B.G.: Metaphors for instruction: Why we talk about learning environments. *Educational Technology*, **35**(5):25-30, 1995.

Liite1: Dublin Core Metadata Element Set, Version1.1: Reference Description (Dublin Core Metadata Initiative 1999)

Seuraavat 15 elementtiä muodostavat Dublin Core metadatastandardin. Olen lisännyt jokaiseen elementtiin esimerkkitapauksen, jota ei ollut sisällytettyä alkuperäiseen dokumenttiin. Jokainen elementti pitää sisällään kymmenen attribuuttia: nimi, tunniste, versio, yhteisö, kieli, määritelmä, pakollisuus, tietotyyppi, määrä ja kommentti. Attribuutit ovat ISO/IEC 11179-standardin mukaisia. En esittele attribuutteja tarkemmin, vaan jokaisessa kohdassa on lyhyt sanallisen selvitys elementin merkityksestä ja käytöstä.

Elementti: Title (otsikko)

Tekijän tai julkaisijan tieto-oliolle antama nimi. Tyypillisesti nimi on se, jolla dokumentti tunnetaan virallisesti. Esimerkki: Tietotekniikan perusteet.

Elementti: Creator (tekijä)

Taho, joka ensisijaisesti on vastuussa tieto-olion sisällöstä. Tekijä on yleensä henkilö, organisaatio tai palvelu. Tarkenteina voidaan käyttää muun muassa Writer- ja Illustrator-tarkentimia. Esimerkki: Tietojenkäsittelytieteen laitos.

Elementti: Subject (aihe)

Kuvaus siitä aihealueesta, jota tieto-olio käsittelee. Tyypillisesti Subject-elementti sisältää avainsanoja, fraaseja tai luokittelukoodeja, jotka kuvaava tieto-olion aihepiiriä. Subject-elementissä suositellaan kontrolloitujen sanastojen tai luokitusten käyttöä, jotta tietyyntyyppiseen aiheeseen kuuluvat oliot voitaisiin liittää toisiinsa. Esimerkki: Java ohjelmointi.

Elementti: Description (kuvaus)

Tieto-olion sisällön kuvaus vapaamuotoisena tekstinä. Description-elementti voi sisältää tiivistelmän, sisällysluettelon, viittauksen toisiin esityksiin tai vapaalla tekstillä olevan kuvauksen olion sisällöstä. Kuvausta ei ole rajoitettu mihinkään edellä mainittuihin.

Elementti: Publisher (julkaisija)

Organisaatio, joka on julkaissut tieto-olion sen nykyisessä muodossa ja on vastuussa sen jake-
lusta. Julkaisija voi olla henkilö, organisaatio, yhteisö tai palvelu. Esimerkiksi: Joensuun yli-
pisto.

Elementti: Contributor (muu tekijä)

Henkilö tai organisaatio, joka on varsinaisten tekijöiden lisäksi antanut merkittävän panoksen
tieto-olin tekemiseen. Muun tekijän vastuu tieto-olion sisällöstä on toissijainen.

Elementti: Date (päivämäärä)

Päivämäärä, joka liittyy tieto-olion elinkaareen tapahtumaan. Tyypillisesti Date-elementtiin
talletetaan olion luonti tai julkaisu päivämäärä. Tallennusmuotona käytetään ISO 8601 -stan-
dardin mukaista esitystä: VVVV-KK-PP. Esimerkki: 2000-02-18.

Elementti: Type (laji)

Tieto-olion luokiteltu lajityyppi, joka voi olla esimerkiksi kotisivu, romaani, runo, kuva tai a-
nimaatio.

Elementti: Format (tiedostomuoto)

Tieto-olion tiedostomuoto eli formaatti, joka voi olla fyysinen tai digitaalinen. Format-
elementti voi sisältää kuvauksen media-tyypistä (MIME-tyyppi) tai olion laajuudesta. Format-
elementti voi sisältää myös kuvauksen niistä laitteisto- ja ohjelmistoympäristöistä, joita olion
käyttäminen edellyttää.

Elementti: Identifier (tunniste)

Tunniste, joka yksikäsitteisesti yksilöi tieto-olion. Identifier-elementti voi olla esimerkiksi
URI-, URL-, DOI- tai ISBN-tunnus.

Elementti: Source (lähde)

Sisältää ne tieto-oliot, joista kyseessä olevan olion tiedot ovat peräisin.

Elementti: Language (kieli)

Tieto-olion sisällössä käytettävä kieli. Elementtiä voidaan toistaa, jos sisältö on monikielinen. Esimerkki käytöstä: "en"-englanti, "fr"-ranska.

Elementti: Relation (suhde)

Viittaus niihin olioihin, jotka ovat jollakin tavalla yhteydessä kuvailtavaan olion. Tarkenteina voidaan käyttää seuraavia tarkentimia suhteen laadun ilmaisemiseksi: IsPartOf, HasPart, IsVersionOf, IsFormatOf, References, IsReferencedBy, IsBasedOn, IsBasisOn tai HasVersion.

Elementti: Coverage (kate)

Tieto-olion ajallinen kate tai maantieteellinen sijainti. Maantieteellinen kate voi tarkoittaa fyysistä aluetta, paikannimeä tai koordinaatteja. Ajallinen kate viittaa siihen, miten kauan olion on käytettävissä.

Elementti: Rights (oikeudet)

Kenellä on oikeudet kyseiseen tieto-olioon. Voi sisältää joko tietoa tekijänoikeudellisista asioista tai linkin paikkaan, josta tietoa löytyy. Esimerkiksi: Public Domain.

Liite2: IMS Core metadataelementit (Anderson ja Wason 2000b)

Seuraava taulukko on osittain yhtenäinen IEEE:n Learning Object Metadata -mallin kanssa, mutta mukaan on otettu pelkästään Andersonin ja Wasonin (2000b) esityksen mukaiset IMS Core metadataelementit. Taulukkoa pitää lukea siten, että aina kun kohdassa ei ole enää alakohtia esimerkiksi 1.2, niin siinä ”solmussa” sijaitsee arvo eli esim. general.title = ”Teoksen nimi”. Vuorostaan kohtaan 1.3 (catalogentry) ei tule arvoa, vaan sen alielementtiin catalogue tulee arvo, esimerkiksi general.catalogentry.catalogue = ”ISBN”.

N:o	Nimi	Selitys	Monimuotoisuus	Tyyppi	Perustelu
1	general	Sisältöriippumattomat ominaisuudet	-	-	-
1.1	identifier	Resurssin yksikäsitteinen tunniste	yksittäinen arvo	Varattu	Vastaa Dublin Coren DC.Identifier elementtiä
1.2	title	Resurssille annettu nimi	yksittäinen arvo	LangStringType (1024 merkkiä)	Vastaa Dublin Coren DC.Title elementtiä
1.3	catalogentry	Resurssin luokitus	järjestämätön lista (8 alkioa)	-	-
1.3.1	catalogue	Luokittelujärjestelmä	yksittäinen arvo	String (1024 merkkiä)	Esim.: ISBN, ARIADNE
1.3.2	entry	Luokitustunnus	yksittäinen arvo	String (1024 merkkiä)	Esim.: 2-7342-0318
1.4	language	Tyypillisen resurssia hyödyntävän henkilön käyttämä kieli	järjestämätön lista (8 alkioa)	String (128 merkkiä)	Käytetään ISO639 ja ISO3166 standardien mukaista merkinäytätapaa. Esim. "en", "en-GB"
1.5	description	Resurssin sisällön kuvaus	järjestämätön lista (8 alkioa)	LangStringType (1024 merkkiä)	-
2	lifecycle	Ominaisuudet, jotka liittyvät resurssin elämänsykliin	-	-	-
2.1	version	Resurssin versionumero	yksittäinen arvo	LangStringType (64 merkkiä)	Esim. 3.0,1.2.alpha
2.3	contribute	Henkilöt tai organisaatiot, jotka osallistuneet resurssin tekemiseen	järjestämätön lista (32 alkioa)	-	-
2.3.1	role	Osallistumisen muoto	yksittäinen arvo	LangStringType (128 merkkiä)	Esim. Author, Publisher, Unknown, Terminator

2.3.2	entity	Osallistuneet organisaatiot	järjestetty lista (tärkeimmät ensin)	String (1024 char)	-
2.3.3	date	Päivämäärä	yksittäinen arvo	DateType	Esim. 2000-11-01
3	metametadata	Tietoa metadataelementeistä	-	-	-
3.1	identifier	Metadatan yksikäsitteinen tunniste	yksittäinen arvo	Varattu	-
3.4	metadatascheme	Käytetyn metadatarakenteen nimi (standardi)	järjestämätön lista (8 alkioita)	String (32 merkkiä)	LOM-1.0
3.5	language	Metadataelementtien kuvaukseen käytetty kieli	yksittäinen arvo	String (128 merkkiä)	-
4	technical	Resurssin tekniset ominaisuudet	-	-	-
4.1	format	Resurssin formaatti eli talletusmuoto	järjestämätön lista (8 alkioita)	String (512 merkkiä)	MIME-tyyppi Esim. video/mpeg
4.3	location	Resurssin sijainti	järjestetty lista (8 alkioita, tärkeimmät ensin)	String (1024 merkkiä)	URI. Esim: http://host/id
6	rights	Resurssin käyttöehdot	-	-	-
6.1	cost	Vaatiiko resurssin käyttö maksua	yksittäinen arvo	String (8 merkkiä)	Varattu sanasto: {yes,no}
6.2	copyright and other restriction	Rajoittavatko resurssin käyttöä copyright tai muut rajoitukset	yksittäinen arvo	String (8 merkkiä)	Varattu sanasto: {yes,no}
6.3	description	Kommentteja käyttöoikeuksista	yksittäinen arvo	String (8 merkkiä)	-
9	classification	Luokittelun mukaiset ominaisuudet	-	-	-
9.1	purpose	Luokittelun tarkoitus	yksittäinen arvo	LangStringType (128 merkkiä)	Esimerkiksi Educational Objective
9.3	description	Tekstipohjainen kuvaus luokituksesta	yksittäinen arvo	LangStringType (2048 merkkiä)	-
9.4	keywords	Avainsanat, jotka kuvaavat ominaisuuksia	järjestetty lista (8 alkioita)	LangStringType (2048 merkkiä)	-