



TAMPEREEN TEKNILLINEN YLIOPISTO

Pasi Häkkinen

# **Videopohjaisen dynaamisen multi- mediaoppimateriaalin tuottaminen**

Diplomityö

AIHE HYVÄKSYTTY  
SÄHKÖTEKNIIKAN  
OSASTONEUVOSTON KOKOUKSESSA  
4. KESÄKUUTA 2003

Tarkastajat: Prof. Seppo Pohjolainen  
Prof. Markku Kivikoski

# Alkusanat

Tämä työ on tehty Tampereen teknillisen yliopiston Hypermedialaboratoriossa. Työn tarkastajina ovat toimineet prof. Seppo Pohjolainen Matematiikan laitokselta ja prof. Markku Kivikoski Elektroniikan laitokselta. Haluan kiittää molempia heidän vaivannäöstään ja työtä koskevista kommentteista. Haluan myös kiittää koko Hypermedialaboratorion henkilökuntaa hienosta työskentelyilmapiiristä. Erityisesti haluan kiittää Pekka Rantaa osallistumismahdollisuudesta mielenkiintoiseen Metsäkonesimulaatio-opetuksen tuotteistaminen-projektiin ja runsaista ideoista itse ohjelmistokehitykseen liittyen.

Esitän myös kiitokseni vaimolleni Tarjalle ja tyttärilleni Anniinalle ja Janinalle heidän tuestaan ja kärsivällisyydestään sekä työtovereilleni Jukka Huhtamäelle ja Lauri Pohjaselle vertaistuesta työn loppuun saattamisessa.

Tampereella 8. tammikuuta 2005

Pasi Häkkinen

Kuminakatu 4 A 1

33710 Tampere

puh. 03 316 0310

s-posti: [pasi.hakkinen@iki.fi](mailto:pasi.hakkinen@iki.fi)

# Sisältö

<i>Tiivistelmä</i> .....	5
<i>Abstract</i> .....	7
<i>Lyhenteet</i> .....	9
<b>1. Johdanto</b> .....	<b>11</b>
<b>2. Multimedia</b> .....	<b>14</b>
<b>2.1. Multimedian historia</b> .....	<b>14</b>
<b>2.2. Multimediasisältöjen käyttökohteita opetuksessa ja opiskelussa</b> .....	<b>17</b>
<b>2.3. Multimediaoppimateriaalin tuotantoprosessi ja sen erityispiirteet</b> .....	<b>22</b>
Ideointi .....	23
Suunnittelu ja käsikirjoitus .....	25
Toteutus .....	27
Jakelu .....	29
Muuta huomioitavaa .....	31
<b>2.4. Video verkkomultimediassa</b> .....	<b>32</b>
<b>3. Sisällön metatiedon tallentaminen ja hyödyntäminen</b> .....	<b>37</b>
<b>3.1. Metatietostandardeja multimedian kuvailuun</b> .....	<b>39</b>
Dublin Core .....	40
LOM.....	40
MARC 21 .....	40
MPEG-7 ja MPEG-21 .....	41
<b>4. Staattinen vs. dynaaminen multimediaesityksen koostaminen</b> .....	<b>43</b>
<b>4.1. Matemaattinen mallinnus -verkostohankkeen luentotallenteet</b> .....	<b>45</b>
Luentotallenteiden tuotantoprosessi.....	46
Pohdintaa toteutuksesta.....	49
<b>5. Dynaamisen multimedian toteutustekniikat</b> .....	<b>53</b>
<b>5.1. SMIL-kuvauskieli</b> .....	<b>54</b>

---

5.2.	Muut verkkomultimedian koostamisformaatit .....	60
5.3.	Soitto-ohjelmat .....	61
6.	<i>Metsäkonesimulaatio-opiskelun verkkoympäristö: SIMUBA</i> .....	64
6.1.	<b>Käyttöperiaate</b> .....	65
	Videoleikkeen siirto .....	66
	Videoleikkeiden kommentointi .....	67
	Opettajan siirtämien videoleikkeiden hyödyntäminen.....	69
6.2.	<b>Tekninen toteutus</b> .....	73
	Palvelinpään toteutus .....	75
	Asiakaspään toteutus.....	77
6.3.	<b>Toteutuksen arviointi</b> .....	77
7.	<i>Yhteenveto ja johtopäätökset</i> .....	82
	<i>Lähdeluettelo</i> .....	85
	<i>Liitteet</i> .....	91
	Liite 1: SMIL-ohjelmakoodiesimerkki .....	91
	Liite 2: SIMUBAn palvelinalusta ja ohjelmaversiot.....	93
	Liite 3: Matemaattisen mallinnuksen luentotalenteiden palvelinalusta ja ohjelmaversiot	94

# Tiivistelmä

## TAMPEREEN TEKNILLINEN YLIOPISTO

Sähkötekniikan koulutusohjelma

Matematiikan laitos, Hypermedialaboratorio

**Häkkinen, Pasi:** Videopohjaisen dynaamisen multimediaoppimateriaalin tuottaminen

Diplomityö, 94 s.

Tarkastajat: Prof. Seppo Pohjolainen

Prof. Markku Kivikoski

Rahoittaja: Tampereen teknillinen yliopisto/Hypermedialaboratorio sekä

Metsäkonesimulaatio-opetuksen tuotteistaminen -projekti

Sähkötekniikan osasto

Tammikuu 2005

Tässä diplomityössä selvitettiin oppimateriaaliksi tarkoitettujen multimediasisältöjen tuottamista dynaamisesti painopisteen ollessa videopohjaisissa sisällöissä. Lisäksi työssä toteutettiin verkko-oppimisympäristö, jonka päätoiminnallisuuksina ovat verkkovideoiden hallinta ja julkaiseminen kirjastonäkymien avulla sekä videoleikkeiden kommentointi ajastettujen tekstileikkeiden avulla.

Videon ja sen ohella myös äänen käyttö ovat lisääntyneet voimakkaasti verkossa oppimateriaalina pakkaustekniikoiden ja tietoverkkojen kehittymisen myötä. Toisin kuin WWW-palvelimilla olevissa teksti- ja kuvasisällöissä, videota sisältävissä multimediaesityksissä ei käytetä juuri lainkaan hyväksi sisällön räätälöintiä katselijan mukaan eikä tietokantoja datan säilytyspaikkana, vaan verkkomultimedia tuotetaan staattisiksi tiedostoiksi sisällöntuottajan työasemalle, josta ne siirretään WWW- tai mediapalvelimille käyttäjien katseltaviksi.

Dynaamisuudella saavutetaan katselijakohtaisen räätälöitävyyden mukanaan tuomat edut, mutta myös muita hyötyjä on saavutettavissa. Medialeikkeiden uudelleenkäytön lisäksi multimediasisältöjen julkaisemista voidaan helpottaa selainkäyttöliittymän kautta käytet-

tävillä hallintatyövälineillä ja itse sisällön tuottamistakin voidaan yksinkertaistaa, kuten tässä työssä toteutettu videoleikkeiden kommentointityöväline on osoittanut.

Dynaamisen verkkomultimedian tuottamiseen on olemassa useita eri tekniikoita, joista tässä työssä hyödynnettiin menestyksekkäästi W3C:n määrittelemää multimedian kuvauskieltä SMIL:iä. Dynaamisuus tuo mukanaan myös tiettyjä vaatimuksia, joista tärkeimpiä ovat multimediasisältöjen rakenteistaminen ja metatiedon kerääminen mediaelementeistä tuotannon kaikissa vaiheissa käsikirjoittamisesta tuotantoon ja julkaisuun saakka.

# Abstract

## TAMPERE UNIVERSITY OF TECHNOLOGY

Degree program in electrical engineering

Department of Mathematics, Hypermedia Laboratory

**Häkkinen, Pasi:** Producing video-based dynamic multimedia learning material

Master of Science Thesis, 94 p.

Examiners: Prof. Seppo Pohjolainen

Prof. Markku Kivikoski

Funding: Tampere University of Technology/Hypermedia Laboratory

Department of electrical engineering

January 2005

In this work production of video-based dynamic multimedia learning material is studied. In addition, an Internet-based learning environment using dynamic multimedia content was developed. The main functionalities of the learning environment are streaming video clip management, publishing through library views and video clip commenting with time sensitive comments.

Usage of video and audio has grown rapidly in online learning material with more efficient compression algorithms and faster networks. User adaptation and database driven content are commonly used in text and image-based web content, but this is not the case with online multimedia. Great majority of multimedia contents are produced as static files with different multimedia creation tools and transferred to web or media servers.

In addition to benefits gained from viewer adaptation, dynamic presentation creation enables content reusability and makes possible to develop and use content management systems with web browser user interface. The actual multimedia production can also be made easier with simple web applications and presentation templates like we have seen with web page generators. Internet greeting cards combining user selectable animation, music and text contents are just the tip of the ice berg of forthcoming in this area.

There are several techniques to produce dynamic multimedia content, both standardized and manufacture specific. In this work, SMIL multimedia mark-up language, specification from W3C, was chosen and successfully deployed as the base technology.

Creation of dynamic multimedia content has some requirements, of which the most important ones are content structuring both in semantic and media type levels and formalized usage of metadata. Importance of metadata collecting should be also noticed during manuscript, shooting and editing processes.



---

# Lyhenteet

3GPP	3rd Generation Partnership Project kolmannen sukupolven matkapuhelinverkkojen standardointiorganisaatio
ASCII	American Standard Code for Information Interchange yleisin tekstitiedostoformaatti
ASP	Active Server Pages dynaamisten WWW-sivustojen toteutustekniikka
CD-ROM	Compact Disc optinen tallennusmedia
CSC	opetusministeriön hallinnoima tieteen tietotekniikan keskus Suomessa
DC	Dublin Core metatietostandardi
DHTML	Dynamic HTML Dynaaminen HTML
DV	Digital Video digitaalinen video
DVD	Digital Versatile Disc optinen tallennusmedia
GIF	Graphics Interchange Format yleinen kuvaformaatti
GUI	Graphical User Interface graafinen käyttöliittymä
HDTV	High Definition Television teräväpiirtotelevisio
HTML	Hypertext Markup Language WWW-sivujen merkkaukieli
HTTP	Hypertext Transfer Protocol WWW-sivujen yleisin siirtoprotokolla
IEEE	Institute of Electrical and Electronics Engineers Elektroniikka-alan standardointijärjestö
JSP	Java Servlet Pages Java-pohjainen dynaamisten WWW-sivustojen toteutustekniikka
LOM	Learning Object Metadata oppimateriaalien metatietostandardi
MIT	Massachusetts Institute of Technology tunnettu amerikkalainen tekniikan alan yliopisto

---

MPEG	Moving Pictures Expert Group video- ja ääniformaattistandardi
PC	Personal Computer tietokone
PDA	Personal Digital Assistant kämmentietokone
PHP	PHP: Hypertext Preprocessor dynaamisten WWW-sivustojen toteutustekniikka
PNG	Portable Network Graphics kuvatiedostoformaatti
QoS	Quality of Service palvelun laatu
SFS	Suomen Standardisoimisliitto
SIMUBA	videoleikkeiden kommentointiin perustuva WWW-pohjainen oppimisympäristö
SMIL	Synchronized Multimedia Integration Language verkkomultimedian merkkauskieli
SVG	Scalable Vector Graphics kuvatiedostoformaatti
TTY	Tampereen teknillinen yliopisto
URL	Uniform Resource Locator tiedoston tai muun resurssin yksilöivä Internet-osoite
W3C	World Wide Web Consortium WWW-suosituksia määrittelevä organisaatio
WWW	World Wide Web Web
XHTML	eXtensible HyperText Markup Language WWW-sivujen merkkauskielen uusin versio
XML	eXtensible Markup Language laajennettava tekstipohjainen tiedon tallennus- ja siirtoformaatti

# 1. Johdanto

Multimediaksi voidaan lukea hyvin erilaisia sisältöjä, mutta tässä työssä käsitellään lähes yksinomaan tietoverkkojen välityksellä siirrettävää multimediaa. Tässä ympäristössä sisällön dynaamisuudesta saavutetaan suurimmat hyödyt. Toinen työssä käytetty rajaus liittyy käsiteltävien oppimateriaalien sisältämiin mediatyyppeihin: Tässä työssä liikkuva kuva on synonyymi videokuvalle, puhtaasti tietokoneella tuotettu animaatio ja sen itsensä mahdollistama dynaamisuus rajautuu pois.

## **Multimedia ja hypermedia:**

Käsitteitä hyperteksti, hypermedia ja multimedia käytetään hyvin vaihtelevasti tarkoittaessa eri mediaelementteistä (teksti, kuva, ääni, videokuva, animaatio jne) koostettuja esityksiä. Rajallisimman määrityksensä mukaisesti hyperteksti on dokumentti, joka koostuu tekstisolmuista ja niiden välisistä linkeistä. Hypermedian rakenne on samanlainen, mutta solmut voivat sisältää tekstin lisäksi myös muita tiedon esitysmuotoja - ääntä, kuvia ja liikkuvaa kuvaa. Multimedia tarkoittaa pelkästään näiden tiedon eri esitysmuotojen yhdistelmää, johon ei liity hypertekstin navigoitavaa rakennetta. Lisäksi multimedia rajataan yleensä tietokoneen välityksellä katseltaviin esityksiin. Siten esimerkiksi televisio ja elokuva eivät ole multimediaa, vaikka ne yhdistävätkin tehokkaasti eri mediatyyppejä.

Käytännössä käsitteiden erottelu ei ole näin helppoa. Puhdasta hypertekstiä käytetään hyvin harvoin, sillä tekstin joukkoon sijoitetaan yleensä kuvia ja yhä useammin myös muita mediaelementtejä. Tämän vuoksi puhutaan usein hypertekstistä, vaikka se sisältäisikin muita mediatyyppejä kuin tekstiä. Rajanveto hypermedian ja multimedian välillä on myös häilyvä, sillä multimediaesityksissäkin käytetään navigointia mm. siirtymiseen esityksen kohdasta toiseen. Tässä työssä rajanveto on tehty siten, että lineaarinen esitys, vaikka se sisältäisikin linkkejä esityksen kohdasta toiseen, on multimediaa, ja epälineaarinen esitys, jossa ei ole selvää alkua ja loppua, on hypermediaa. Tämän jaottelun perusteella luennosta tehty tallenne on multimediaa ja eri luentotallenteista koottu WWW-sivusto vapaalla navigoinnilla on hypermediaa.

**Dynaaminen multimediaesitys:**

Multimedia on sisältönsä puolesta yleensä dynaamista, pitäähän multimediaesitys sisällään lähes aina liikkuvaa kuvaa ja/tai ääntä. Tässä työssä dynaamisuudella viitataan kuitenkin siihen, kuinka multimedia- tai hypermediaesitys koostetaan katselijalle esitystilannetta varten eli miten ja missä vaiheessa eri mediaelementit sijoitellaan toisiinsa nähden sekä asemoinnin osalta (layout) että ajallisesti. Dynaaminen multimediamateriaali koostetaan vasta esitysvaiheessa, jolloin esityksen sisältöä, rakennetta ja esitystapaa voidaan muuttaa sekä katselijakohtaisesti että ajallisesti päivitysten muodossa. Toimintaidea on sama kuin WWW-julkaisujärjestelmissä, joissa asiasisältö eli tekstit ja kuvat, on erotettu WWW-sivun ulkoasusta ja jokaisen WWW-sivun sisältö voidaan räätälöidä katselijakohtaisesti.

Länsimainen yhteiskunta on hyvin video-orientoitunut. Lapset oppivat jo hyvin varhain katselemaan televisiota ja elokuvia, televisio on ohittamassa painetun median tärkeimpänä uutisinformaation lähteenä ja videokameran käyttö perhetapahtumien ja lomamatkojen tallentajana on arkipäiväistä. Video on siten mediana erittäin tuttu ja läheinen useimmille meistä ja sen vuoksi videon käyttö myös oppimateriaalina on perusteltua. Toisaalta televisio on opettanut meidät varsin vaativiksi videomateriaalin (teknisen) laadun suhteen, mikä nostaa kynnystä aloittaa oman materiaalin tuottaminen pienillä resursseilla. Tähän ongelmaan tarjoaa ratkaisun digitaalitekniikka, joka mahdollistaa jopa TV-tuotannon videokamera-tietokone -yhdistelmällä, kunhan näitä välineitä osataan käyttää oikein.

Jos sisällön suunnittelu ja itse videokuvaaminen jätetään huomiotta, niin suurin tekninen ongelma, joka liittyy videomateriaalin käyttöön perinteisen tekstipohjaisen materiaalin rinnalla tai sen korvaajana, on liikkuvan kuvan tallentamiseen tarvittava suuri datamäärä. Videomateriaali tarvitsee pakattunakin suuren tallennus- ja siirtokapasiteetin, jota ei ole kaikkialla käytettävissä. Tämän vuoksi kaiken informaation esittäminen videona ei ole järkevää, vaan osa tietosisällöstä tulee välittää vähiten kapasiteettia tarvitsevien mediatyyppien, tekstin ja kuvan avulla päätelaitteisiin, joissa se on mahdollista. Parhaassa tapauksessa kaikki eri mediaelementtityypit voidaan yhdistää yhdeksi, käyttäjän kannalta saumattomaksi esitykseksi, joka tarjoaa käyttäjän tarvitseman informaation helposti riippumatta siitä, missä tai millä päätelaitteella hän sitä katselee ja käyttää. Tähän tavoitteeseen päästään ainoastaan dynaamisten multimediasisältöjen avulla, joissa katselukokemus

voidaan räätälöidä katselijan, käytössä olevan tietoverkon siirtokaistan ja päätelaitteen mukaan.

Dynaamisia multimediasisältöjä hyödyntäviä palveluita ei juurikaan ole käytössä. Yhtenä suomalaisena esimerkkinä voidaan nostaa esiin Yleisradion Opinportti. YLE Opetusohjelmat [Opinportti 2004] tarjoaa Suomen suurimman opetusalan video- ja audioleikepankin Klaffin, jossa voi katsella ja kuunnella lähinnä Yleisradion opetusohjelmien tarjonnasta poimittuja medialeikkeitä. Leikkeitä on metatietoineen tallennettu yli 2000 kappaletta ja ne ovat vapaasti käytettävissä yksityiseen ja koulukäyttöön. Opinportin Koostamotyökalulla voi rakentaa omaa oppimateriaalia, johon voi liittää Klaffista löytyvää video- ja audioaineistoa. Toimintaperiaatteeltaan työkalu on template-pohjainen eli valmiiseen sivupohjaan voidaan tuottaa tekstiä WWW-selainkäyttöliittymästä HTML-editorilla, luoda kuvia omalla kuvaeditorilla tai tuoda valmiita kuvia sekä liittää yksi audio- tai videoleike leikepankista.

Tässä työssä käydään läpi seikkoja, joita tulee huomioida suunniteltaessa ja toteutettaessa dynaamisia multimedia- tai hypermediatuotteita sekä esitellään erilaisia teknisiä ratkaisuvaihtoehtoja dynaamisuuden aikaansaamiseksi. Luvussa 2 kerrotaan multimedian historiasta ja multimediasisältöjen käyttökohteista opetuksessa sekä esitellään multimediaoppimateriaalin tuotantoprosessi ja sen erityispiirteet. Luvussa 3 perustellaan medialeikkeiden kuvailemisen tärkeys metatiedon avulla sekä esitellään tärkeimmät multimedian kuvailuun käytettävät metatietostandardit. Luvussa 4 verrataan perinteistä staattisen multimediaesityksen koostamista dynaamiseen koostamistapaan ja esitellään esimerkkinä staattisen sisällön tuottamisesta Matemaattinen mallinnus -verkostohankkeen luentotallenteiden tuotantoprosessi. Luvussa 5 esitellään toteutustekniikoita eri verkkomultimediaformaattien ja soitto-ohjelmien avulla. Esimerkki dynaamisesta multimediatoteutuksesta käsitellään luvussa 6, jossa esitellään Metsäkonesimulaatio-opetuksen tuotteistaminen –projektissa tuotettu videoleikkeiden arviointi- ja julkaisutyöväline. Luvussa 7 kerrotaan tämän työn yhteenvedo ja johtopäätökset.

## 2. Multimedia

Multimediaa on käytetty viihdetarkoituksiin koko sen historian ajan. Yhdistelemällä useaa eri mediaa ja hyödyntämällä useaa eri aistia on pyritty tuottamaan kokonaisvaltaisempia elämyksiä multimediatuotteen käyttäjälle. Tätä elämyksellisyyttä on hyödynnetty myös oppimisessa. Ihminen on luontaisesti visuaalinen ja auditiiivinen olento ja hän oppii asioita vauvasta lähtien näkemällä, kuulemalla ja koskettamalla. Haju- ja makuaistin avulla voi myös oppia, mutta multimedialla ei olla vielä päästy hyödyntämään niitä täysimääräisesti.

Multimediaa käytetään viihde- ja koulutussovellusten lisäksi tiedon välittämiseen, suunnittelun apuvälineenä mallintamiseen, arkkitehtuurin ja maankäytön suunnitteluun, infokioskeissa, markkinoinnissa ja käyttöoppaissa. Tässä työssä painopiste on koulutuskäytön puolella, mutta koska työn näkökulma on tekninen, samat ideat toimivat myös muissa konteksteissa.

### 2.1. Multimedian historia

Multimedia ja tietokonepelit ovat kulkeneet käsi kädessä koko tietokoneen historian ajan. Tietokoneen ja ihmisen välinen vuorovaikutus ja käyttäjän osallistuminen tietokoneen toimintaan ovat kiinnostaneet ohjelmoijia jo 50-luvulta lähtien. Ensimmäinen videopeliksi luokiteltava peli "Tennis for two" on vuodelta 1958 ja sitä pelattiin oskilloskoopin näytöllä. Fyysikko Willy Higinbothan kehitti sen Brookhavenin ydintutkimuslaitoksessa New Yorkissa viihdyttääkseen kuiviin esittelyihin pitkästyneitä vierailijoita. [The Dot Eaters 2004]

Ensimmäinen oikea interaktiivinen tietokonepeli kehitettiin MIT:ssä vuonna 1961 DEC PDP-1 mainframe-tietokoneelle ja se oli nimeltään Spacewar! Peli levisi laajalle huomioon pelialustan yleisyyden, ja sille tehtiin jopa oma peliohjain. [The Dot Eaters 2004]

Ensimmäisten tietokoneiden vaatimattomat tallennus-, näyttö- ja ääniominaisuudet rajoittivat eri medioiden käyttöä pitkään. Vasta 1970-luvun loppupuolella videolevy-tekniikka mahdollisti riittävän nopeat hakutoiminnot videomateriaalin käyttöön. Yksi kunnianhimo-

simmista hankkeista tuolloin oli Yhdysvaltain puolustusministeriön rahoittama ja MIT:in kehittämä Spatial Data Management System (SDMS), jonka tavoitteena oli suurten tietomassojen hallittavuuden parantaminen monimediaalisen ihmisen ja koneen välisen vuorovaikutuksen avulla. Perusajatuksena järjestelmässä oli ihmiselle luontainen tilapohjaisuus informaation esittämisessä, kontrolloinnissa ja navigoinnissa. Näön ja kuulon lisäksi tuntoaisti oli merkittävässä asemassa. Järjestelmä piti sisällään mm. erikoisvarustellun mediahuoneen, neljä minitietokonetta, lentäjän tuolin, videoprojektorin ja kaksi kosketusnäyttöä. [Donelson 1978]

Videolevy-tekniikkaa hyödynnettiin myös vuonna 1980 Aspen-projektissa, jossa luotiin videopohjainen virtuaalijelu Aspenin kaupungin kaduilla. Ideana oli luoda kaupungista interaktiivinen videokartta, josta pystyi selaamaan eri reittivaihtoehtoja kosketusnäytön avulla. [Aspen Moviemap 2004]

Yleiskäyttöiset, graafisilla käyttöliittymillä varustetut PC-tietokoneet avasivat multimediaman digitaalisen maailman suuremmalle yleisölle. Xerox Alto vuodelta 1973 oli ensimmäinen graafisella käyttöliittymällä ja hiirellä varustettu henkilökohtainen tietokone. Alto jäi tutkimuskäyttöön, sillä Xerox ei koskaan kaupallistanut sitä. Alton seuraajasta, Xerox Star 8010 vuodelta 1981, ei myöskään tullut kaupallista menestystä, sillä IBM PC ehti jo vallata PC-markkinat avoimella arkkitehtuurillaan. Altossa ja Starissa esitellyt innovatiiviset ideat ja teknologiat jäivät kuitenkin elämään ja niitä hyödynsivät myöhemmin niin Atari, Apple kuin Microsoftkin omissa tuotteissaan. [Digibarn 2004] [Redant 2001]

David Backer esitteli 1980-luvun alussa ensimmäisen sähköisen kirjan, jossa käyttöliittymä oli kirjan tapainen. Ohjaus tapahtui kosketusnäytöltä ja laitteessa oli puheselostus. Kirjan sisältö oli interaktiivinen ja siihen oli integroitu ja synkronoitu ääniä, tekstiä, kuvia ja videoita. [Backer 1988]

Tietokoneet alkoivat levitä koteihin ja kouluihin 1980-luvun loppupuolella. Commodore 64 yleistyi Suomessa kotien ensimmäisenä tietokoneena 80-luvun puolivälissä ja sen jälkeen tulivat graafisella käyttöliittymällä ja jo kohtuullisen hyvillä ääniominaisuuksilla varustetut tietokoneet: Apple Macintosh, Atari ST ja Commodore Amiga. Apple esitteli vuonna 1987 HyperCardin, josta tuli nopeasti eniten käytetty hypertekstijärjestelmä ja sen

päälle kehitettiin useita muita sovelluksia. IBM PC jatkoi tuolloin yleistymistään, mutta multimediakoneeksi sitä ei voinut vielä kutsua.

Multimedia teki läpimurtonsa 1990-luvun alussa, kun tavallisten koti- ja toimistotietokoneiden ominaisuudet alkoivat riittää videomateriaalin esittämiseen näytöllä. Multimedia-CD-ROM:it toivat multimedian jokaisen tietokoneen käyttäjän ulottuville, kun Applen tietokoneiden lisäksi PC:t alettiin varustaa CD-ROM-aseamalla ja ääniominaisuuksilla.

Multimedian tuotanto-ohjelmat eli multimediatekijät tulivat markkinoille samoihin aikoihin ja ne madalsivat kynnystä tuottaa omia multimediatuotteita esimerkiksi koulutus-käyttöön. Tuon ajan tunnetuimmat ja käytetyimmät ohjelmistot olivat Asymetrix Toolbook ja Macromedia Director.

Internet oli ottanut ensiaskeleensa jo 1970-luvun puolella, mutta vasta WWW:n keksiminen 1991 ja graafisten WWW-selainten esiinmarssi muutamaa vuotta myöhemmin räjäyttivät verkon käytön eksponentiaaliseen kasvuun. Internetistä tuli yhtä kuin WWW, ja aikaisemmin tekstivoittoiset WWW-sivut muuttuivat visuaalispainotteisiksi kuvien käytön myötä.

Ääntä ja liikkuvaa kuvaa ryhdyttiin välittämään Internetissä pian WWW:n käyttöönoton jälkeen. Multimediaan oli jo totuttu CD-ROM:ien kautta ja vastaavaa sisältöä yritettiin tuottaa pian myös verkkoon. Multimediasisältöjen käyttöä rajoittivat kuitenkin alhaiset siirtonopeudet ja aluksi jouduttiinkin tyytymään huomattavasti multimedia-CD-ROM:ejä köyhempään mediasisältöön.

Aluksi lyhyitä medialeikkeitä siirrettiin verkossa normaalien tiedostojen tapaan, jolloin esimerkiksi videoleikettä pystyi katsomaan vasta sen jälkeen, kun koko videotiedosto oli latautunut ja talletettu omalle tietokoneelle. Tilanne muuttui 1990-luvun puolivälin jälkeen, kun ensimmäiset streaming- eli suoratoistoformaattit ja -ohjelmistot tulivat käyttöön. Näiden etuna oli se, että tiedoston toisto tilaajan tietokoneella saattoi alkaa heti, kun riittävä määrä dataa oli vastaanotettu. Mediavirtaa ladattiin samalla, kun sitä toistettiin. Kohdeyrymänä olivat tuolloin lähinnä modeemikäyttäjät, koska nopeampia yhteyksiä oli hyvin harvalla käytettävissään. [Häkkinen et al. 1997]



Ensimmäiseksi uusi tekniikka otettiin käyttöön äänen siirtämisessä. Tunnetuin ensimmäisistä audiostream-formaateista oli RealAudio, joka nopeasti valtasi äänistreaming-markkinat lukuisilta muilta yrittäjiltä. Pian äänen jälkeen myös videota varten kehitettiin omia suoratoistoformaatteja, joiden avulla pystyttiin välittämään suhteellisen heikkotasoisia 'postimerkki'-videota. Tyypillistä tälle streaming-tekniikan pioneeriajalle olivat lukuisat eri kilpailevat formaatit ja yhtä lukuisat soitto-ohjelmat, koska jokainen eri formaatti vaati oman lisäohjelman asennuksen tietokoneeseen.

Ääni- ja videomateriaalien käytön yleistymisen myötä palvelinpuolella otettiin käyttöön ensimmäiset mediapalvelimet tavallisten WWW-palvelinten rinnalle. Video- ja äänimateriaalin jakamiseen suunniteltujen mediapalvelimien avulla pystytään palvelemaan suuriakin yhtäaikaista katselijamääriä ja niiden avulla suorat verkkolähetykset eli ns. live-lähetykset ovat mahdollisia.

Streaming-markkinat alkoivat nopeasti keskittyä 1990-luvun lopulla. Valta-asemaan nousi Real Networks, joka on edelleen markkinajohtaja streaming-markkinoilla. Microsoft myöhästyi hiukan omassa aloituksessaan, mutta se on tasaisesti kasvattanut markkinaosuuttaan vuoden 2000 jälkeen ja on tällä hetkellä tasoissa Realin kanssa. Kolmas suuri toimija on Apple Quicktime-formaatillaan, jonka vahva historia CD-ROM-multimediassa on kantanut myös streaming-aikakauteen. Näiden kolmen kaupallinen multimedia-arkkitehtuurin lisäksi verkkomultimediassa käytetään jonkin verran MPEG-standardien [MPEG 2003] mukaisia formaatteja, joista verkkomultimediassa odotetaan eniten MPEG-4:ltä, sekä Macromedian Flash- ja Director-tuoteperheitä.

## **2.2. Multimediasisältöjen käyttökohteita opetuksessa ja opiskelussa**

Multimediaa käytetään opetuksessa ennen kaikkea havainnollistamaan opiskeltavia asioita ja ilmiöitä aktivoimalla ja käyttämällä useita eri aisteja samanaikaisesti. Useimmille ihmisille näköaisti on tärkein kanava vastaanottaa informaatiota ympäristöstään, mutta myös kuulo-, tunto- ja makuaistit ovat tärkeitä lähteitä, joissakin tilanteissa jopa näköä tärkeäm-

piä. Multimediasovelluksissa pyritään lähinnä hyödyntämään näköä ja kuuloa, tuntoaistin käyttö on vasta tulollaan haptisten järjestelmien, esimerkiksi datakäsineiden, kehittymisen myötä. [Clarke 2001]

CD-ROM-pohjaista multimediaa on käytetty opetuksessa jo kahdenkymmenen vuoden ajan, mutta tämän työn kannalta mielenkiintoisemman verkkomultimedian historia ulottuu paljon lähemmäs. Ensimmäiset suoratoistoratkaisut otettiin yliopistoissa käyttöön 90-luvun lopussa, jolloin silloisen tekniikan rajoissa tuotettiin ensimmäisiä ääni- ja videopohjaisia oppimateriaaleja verkkoon.

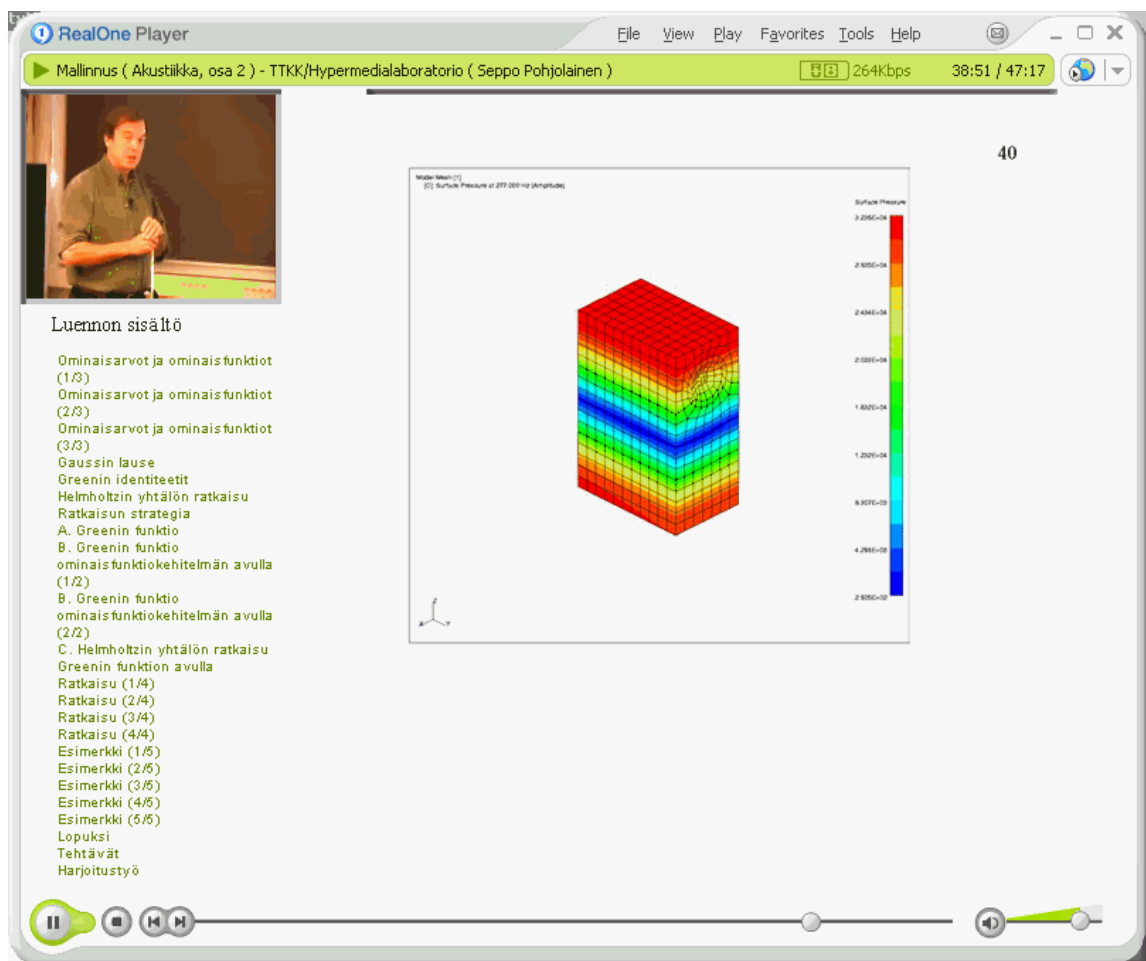
Verkon kautta seurattavaa ääni- ja videomateriaalia käytetään pääasiassa kolmessa erityyppisessä tapauksessa. Lähiopetuksen ja erityisesti etäopetuksen tukimateriaalina ja havainnollistamisvälineenä verkkovideo tarjoaa selkeän lisäarvon pelkkään teksti- ja kuvapohjaiseen materiaaliin verrattuna. Esimerkiksi kielten ja musiikin opiskelussa ääni- ja videopohjaisen sisällön edut ovat selkeästi todettavissa. Toisessa käyttötapatyyppissä koko kurssin tai muun opintojakson toteutus perustuu suoratoistoratkaisulle, jolloin pääosa opetusmateriaalista on mediatallenteena verkossa. Tätä käytetään erityisesti yliopistojen välisissä verkostohankkeissa. Näiden kahden käyttötyyppin välissä ovat tapaukset, joissa normaalit lähiopetusluennot tallennetaan ja tarjotaan opiskelijoille kertausmateriaaliksi ja/tai korvaamaan lähiopetustilanteet tilanteissa, joissa opiskelija ei pääse seuraamaan luento- aikatauluongelmien tai pitkien välimatkojen takia. Pääpaino näyttää tällä hetkellä olevan kahdessa viimeksi mainitussa mediatallenteiden käyttötavassa [Sariola et al. 2003].

Etäopetuksessa on käytetty luentotallenteita aina videonauhurin keksimisestä ja yleistymisestä lähtien. Videokasetilla on pystytty välittämään luennoijan ääni ja kuva sekä videokameralla kuvattua oheismateriaalia sellaisille henkilöille, joilla ei ole ollut mahdollisuutta osallistua luentoaiheen lähiopetusjaksoille.

Tietoverkot tarjoavat uuden siirtomedian ääni- ja videotallenteiden välittämiseen. Tietokoneen käyttö tallenteiden esitysvälineenä tuo vanhaa TV-nauhuri -yhdistelmää enemmän vaihtoehtoja luentotallenteen koostamiseen. Yksinkertaisin tapa on muuttaa videotallenne suoraan suoratoistoformaattiin yhdeksi videoleikkeeksi, mutta järkevämpää on erottaa esitysmateriaali videokameralla kuvatusta videosta ja esittää se alkuperäisessä formaatis-

saan, esimerkiksi kuvina ja tekstinä, videokuvan rinnalla. Tällöin pystytään takaamaan esimerkiksi luentokalvoille paras mahdollinen kuvanlaatu ja tallenteeseen voidaan rakentaa sisäinen navigointi. Videokuva voidaan myös tällaisessa tapauksessa jättää kokonaan pois, jos sen tuoma lisäarvo pelkkään ääneen verrattuna on pieni.

Seuraavassa kuvassa on esitetty tyypillinen käyttöliittymä luentotallenteesta eli miltä se näyttää luennon seuraajan näkökulmasta. Pääosan näytön alasta vie oikeassa reunassa näkyvä luentomateriaalikalvo. Vasemmassa yläkulmassa pyörii videokuva luennoijasta ja vasemmassa alareunassa on listattuna luentokalvojen otsikot. Luentokalvot on synkronoitu videotallenteeseen eli ne vaihtuvat luennoijan puheen tahtiin. Kalvojen otsikot toimivat linkkeinä kohtiin, joissa luennoija alkaa käsitellä kyseistä kalvoa. Näiden linkkien avulla katselija voi liikkua luennossa haluamaansa kohtaan täysin vapaasti.



Kuva 1 Esimerkki SMIL-kuvauskielellä toteutetusta luentotallenteesta Matemaattisen mallinnuksen verkkokurssilla, joka 'luennoidaan' ainoastaan luentotallenteiden avulla.

Verkon kautta katseltavissa olevan luennon tai muun tallenteen voi seurata itselle parhaiten sopivana ajankohtana ja haluamassaan paikassa. Tämä on hyvin suuri etu henkilöille, jotka eivät pystyisi osallistumaan lähiopetukseen työn, muiden opintojen tai pitkien välimatkojen takia. Lähiopetuksen korvaamisen lisäksi luentotallenteet soveltuvat hyvin opitun asian kertaamiseen. Luennoista on helppo poimia ne kohdat, jotka vaativat kertausta ja itselle selvät osiot voi jättää katselematta ja kuuntelematta. Jo ensimmäiselläkin katselukerralla epäselvien tai vaikeiden kohtien uudelleenkatseleminen on selkeä etu perinteiseen luento- verrattuna, tosin tarkentavien kysymysten esittäminen luennoijalle täytyy verkkoympäristössä järjestää erikseen toisin kuin live-tilanteessa.

Tekniikalla ja sen toimivuudella on luonnollisesti suuri vaikutus luento- ja muiden mediatalletojen onnistuneeseen hyödyntämiseen. Tallenteiden seuraaminen verkon välityksellä asettaa vaatimuksia niin päätelaitteiden kuin verkonkin ominaisuuksille. Kiinteillä verkkoyhteyksillä toimittaessa yhteysongelmat ja kaistan riittämättömyys ovat selkeitä vikatilanteita lukuun ottamatta suhteellisen harvinaisia, mutta kohdalle osuessaan ne estävät opetusmateriaalin käytön. Laajakaistaliittymien yleistymisen koodissa ja opiskelijasuuntojen kiinteät verkot eivät poista kuitenkaan sitä tosiasiaa, että siirtokaista on rajallinen resurssi: modeemikäyttäjiä tulee olemaan tulevaisuudessakin ja uudet mobiiliyhteydet tarjoavat kiinteitä yhteyksiä heikomman palvelutason. Tämän vuoksi sisällöntuotannossa ja tallenteiden jakelussa tulee huomioida skaalautuvuus tarjoamalla esimerkiksi videotallenteesta pelkkä audio-versio. Verkkokaistan rajallisuus voi tulla esiin myös palvelinpuolella, jos esimerkiksi reaaliaikaiselle lähetykselle tulee satoja yhtäaikaista seuraajaa eikä tähän ole varauduttu hajautetulla sisällönjakelulla.

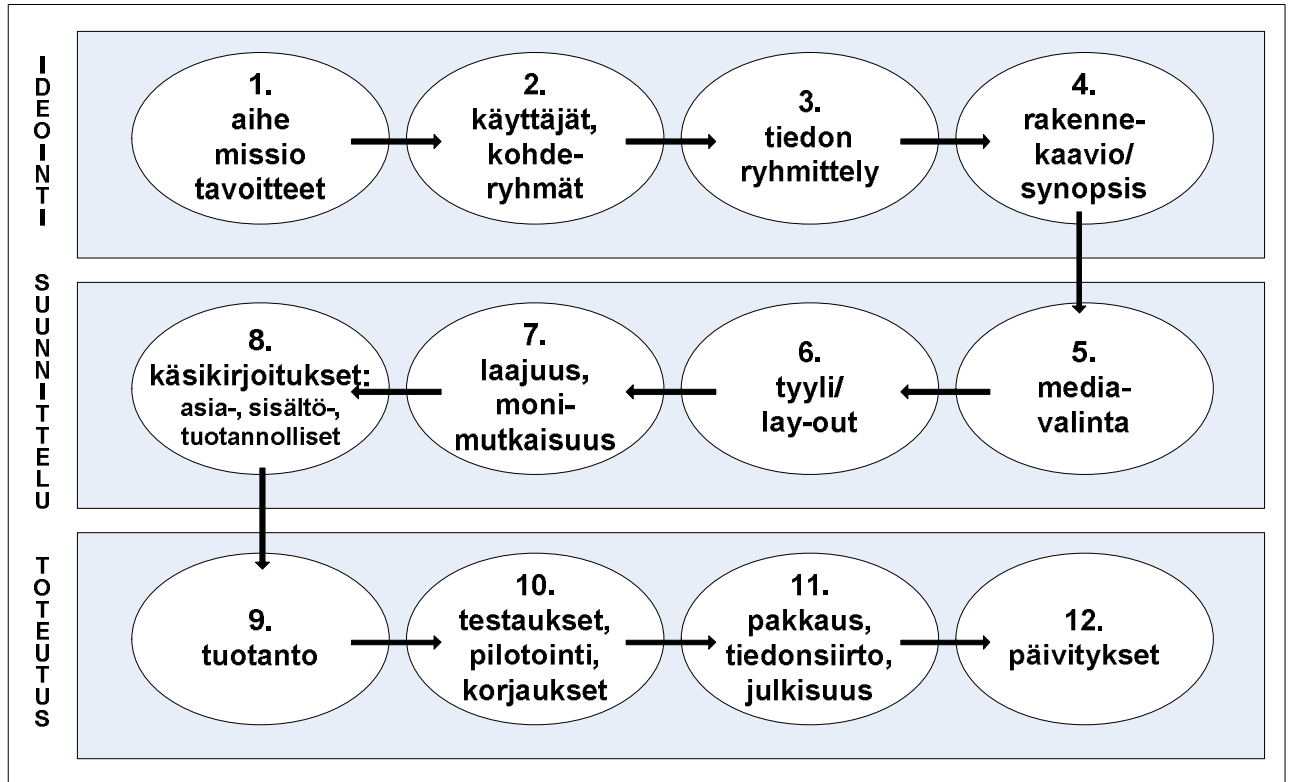
Teknisiä ongelmia liittyy myös ohjelmistoihin, joilla verkkotalletoita katsellaan. Jokainen suoratoistoformaatti vaatii käytännössä oman soitto-ohjelman, jonka täytyy olla usein viimeisintä versiota. Soitto-ohjelmien pitäminen ajan tasalla, jos siitä yleensä on olemassa versio halutulle käyttöjärjestelmälle, vaatii erityisesti keskitetyn ylläpidon ympäristöissä lisätyötä. Kotitietokoneille ohjelmistojen päivitys ei yleensä ole ongelma, mutta yliopistojen ja varsinkin yritysten tietokoneilla omatoimisuudelle ei tarjota mahdollisuuksia, vaan ohjelmistotarjonnassa ja päivityksissä ollaan ylläpidon armoilla.

Nykyinen tekniikka mahdollistaa reaaliaikaiset lähetykset opetustilanteista ja ensimmäisiä toteutuksia on jo olemassa. Suoria lähetyksiä voidaan käyttää videoneuvotteluna toteutettavan opetuksen rinnalla tai jopa videoneuvottelun korvaavana tekniikkana tilanteissa, joissa vuorovaikutus osallistujien välillä on vähäistä ja kysymysten esittäminen ja muu kommentointi voidaan toteuttaa tekstipohjaisesti esimerkiksi WWW:n kautta. Suoratoistopohjaisen suoran lähetysten etuina videoneuvotteluun verrattuna ovat yksinkertaisemmat laite- ja verkkoyhteysvaatimukset vastaanottopäässä, yksinkertainen täysin WWW-pohjainen käyttöliittymä ja opetuksen seuraamismahdollisuus suoraan opiskelijan omalta työasemalta. Jätettäessä videokuva pois lähetyksestä luennon seuraamiseen reaaliaikaisesti riittää hyvinkin vaatimaton verkkoyhteys, kun esitysmateriaali jaetaan etäpisteisiin äänen tueksi.

## **2.3. Multimediaoppimateriaalin tuotantoprosessi ja sen erityispiirteet**

Lukkari [Lukkari 2004] jakaa digitaalisen sisältötuotantoprojektin kolmeen vaiheeseen: ideointi, suunnittelu ja toteutus. Tätä vaiheistusta voidaan käyttää hyvin multimediatuotannossa kunhan muistetaan, että jokainen tuotantoprosessi on erilainen ja prosessin kulku riippuu sisällöstä. Jos tuotettava oppimateriaali on hyvin videopainotteista, esimerkiksi video-DVD, tuotantoprosessille voi hyvin hakea mallia perinteisestä AV-tuotannosta, ja toisaalta interaktioon perustuvan WWW-sovelluksen tuottaminen on selkeästi ohjelmistotekniikkaa.

Kuva 2 esittää yhden tavan jäsentää tuotantoprosessi eri vaiheisiin. Vaiheistus voidaan tehdä myös muulla tavoin, mutta kaikissa tuotantoprojekteissa on syytä tunnistaa ja erottaa toisistaan päävaiheet: ideointi, suunnittelu ja toteutus. Seuraavassa on käyty läpi lyhyesti multimediasisällön tuotantoprosessin eri vaiheet ja niissä huomioitavia asioita.



Kuva 2 Sisältötuotantoprojektin vaiheet [Lukkari 2004]

## Ideointi

Oppimateriaaliksi tarkoitetun multimediatuotteen tuotantoprosessi alkaa ideoinnilla aivan samalla tavalla kuin minkä tahansa muunkin multimediasisällön tapauksessa. Vaiheesta voidaan käyttää myös nimeä määrittely, koska tässä vaiheessa on syytä saada selkeä näkemys siitä, kenelle, mitä ja millä rajaehdoin oppimateriaali tuotetaan. Sisällön osalta jo alkuvaiheessa on yleensä suhteellisen hyvä kuva siitä, mistä aiheesta tai aiheista tuote tul- laan tekemään ja tämä tarkentuu myöhemmin käsikirjoitusvaiheessa. Haasteellisempi tehtävä on määrittellä, kenelle tuote tehdään, miten hoidetaan jakelu ja mitä resursseja tarvitaan eri tuotannon vaiheissa.

Kohderyhmän määrittely on syytä tehdä huolellisesti heti alkuvaiheessa, koska kohdeylei- sö vaikuttaa sekä sisällön esitystapaan että valittavissa oleviin multimedian toteutustekni- koihin. Oppimateriaalin tapauksessa kohderyhmä on yleensä löydettävissä ja rajattavissa järkevästi, koska harva materiaali on tarkoitettu kaikille esikoululaisista vaareihin ja

mummoihin. Esitystapaan liittyen kohdeyleisö vaikuttaa sisällön rakenteen, navigointiratkaisujen ja näkymien suunnitteluun sekä mediatyyppien valintaan.

Esitystapaa konkreettisemmin kohderyhmä sanelee sen, mikä tai mitkä ovat sisällön jakelutiet eli miten digitaalinen tieto tai teos välitetään kohderyhmilleen. Jakelutiet eivät ole enää toisiaan poissulkevia eivätkä välttämättä edes irrallisia toisistaan, sillä nykyisin voidaan yhdistää samaan tuotteeseen sekä online-jakelu (verkko) että offline-jakelu (optinen tallennusmedia). Jakeluteillä on kuitenkin edelleen selviä eroavaisuuksia ja niiden sopivuus pitää arvioida tilannekohtaisesti. Kohderyhmä määrittelee, riittääkö yksi loppuformaatti – DVD, CD-ROM tai WWW-sivusto – vai pitääkö loppuformaatteja olla useita. Jos jakeluformaatteja on useita, täytyykö niiden vastata yksi yhteen toisiaan vai voivatko ne erota toisistaan paljonkin sisältöpainotuksiltaan? Ihannetapauksessa lopputuote toimisi ilman muokkauksia suoraan sekä online- että offline-jakelussa, mutta tämä tilanne saavutetaan harvoin erityisesti silloin, kun tuotetaan rakenteeltaan dynaamista sisältöä.

Multimediatuotteen tekeminen vaatii hyvin monialaista osaamista eli tuotannon eri vaiheissa tarvitaan usean eri henkilön työpanosta alkaen sisällön asiantuntijoista käsikirjoittajiin, graafikoihin, kuvaajiin ja ohjelmoijiin. Jos tuotantoprosessille on jo määritelty rajaehdot resurssien osalta eli on jo olemassa tieto, keiden henkilöiden työpanos on käytettävissä ja mitä lopputuote saa maksaa, resurssit pitää huomioida jo ideointivaiheessa. Jos taas koko tuotantoprojektin budjetti laaditaan vasta lopputuotteen määrittelyn jälkeen, ideointivaiheen lopussa tulisi olla ainakin oikeassa mittasuhteessa oleva arvio siitä, mitä lopputuotteen tekeminen vaatii ajallisesti ja rahallisesti.

Ideointivaiheen lopputuloksena syntyy synopsis eli lyhyt tiivistelmä tuotettavan oppimateriaalin ideasta: mitä ja kenelle. Synopsiskeen tiivistetään tuotteen punainen lanka tai tarina, joka saa tuotteen käyttäjän kiinnostumaan juonesta, "jäämään koukkuun" ja oppimaan juoneen liitetyn materiaalin avulla. Lisäksi tässä vaiheessa on hyvä hahmotella rakennekaavio siitä, kuinka oppimateriaaliin valittu tietosisältö ryhmitellään lopputuotteessa, koska sen pohjalta sisällön asiantuntija tai tuotteen tilaaja pystyy kommentoimaan ja hyväksymään rakenteen ja sitä voidaan käyttää suoraan suunnitteluvaiheen pohjana.



## **Suunnittelu ja käsikirjoitus**

Kun määrittely on olemassa, voidaan aloittaa varsinaisen tuotteen suunnittelu. Suunnittelun tuloksena syntyy käsikirjoitus, jonka tarkkuuteen on syytä kiinnittää huomiota, jotta toteutusvaiheessa ei kulu aikaa ylimääräiseen arvuutteluun siitä, mitä pitäisi tehdä. Keskeisiä kysymyksiä tässä vaiheessa ovat: Mitä mediaelementtejä käytetään? Miltä oppimateriaali tai muu lopputuote näyttää ja millaisia taittomalleja käytetään eli tyylilaji? Miten laaja ja monimutkainen tuotteen rakenteesta tehdään eli kuinka monta navigointitasoa tuotteeseen toteutetaan? [Lukkari 2004]

Suunnittelussa tulee olla mukana niin sisällön, tekniikan kuin pedagogiikan asiantuntijoita, koska näiden kaikkien osa-alueiden osaaminen löytyy erittäin harvoin samasta henkilöstä. Sisällön asiantuntijuus on itsestäänselvyys tuotettaessa opiskelumateriaalia, mutta kahta muutakaan ei pidä aliarvioida. Oli loppuformaatti mikä tahansa, jokaisella toteutustekniikalla on omat mahdollisuutensa ja rajoituksensa, joiden tiedostaminen ja tunteminen auttavat toisaalta saamaan kaiken hyödyn irti valituista tekniikoista ja toisaalta hylkäämään epärealistiset tai liian kalliit ideat jo suunnittelupöydällä. Pedagogisia taitoja tarvitaan suunniteltaessa sitä, kuinka tieto kannattaa opetuksen ja oppimisen kannalta esittää. Lisäksi on osattava ottaa kantaa sisältöjen rajauksiin, tuotteen käyttötapoihin, mahdollisiin sisältöön liittyviin harjoituksiin sekä arviointiin.

Suunnittelun vaatima työmäärä riippuu hyvin paljon toteutettavasta multimediatuotteesta. Yksinkertaisimmillaan riittää yksi tai kaksi palaveria silloin, kun toimitaan jonkin valmiin pohjaratkaisun päällä ja tuotantomalli on jo valmiina olemassa. Tällainen tilanne on esimerkiksi luentotallenteiden teossa, joissa yksittäisten luentojen tallentamisessa toistetaan pääsääntöisesti samaa kaavaa ja käsikirjoituksena toimii luennoijan laatima rakenne omasta esityksestään. Erona normaaliin luentoon valmistautumiseen on ainoastaan se, että luennoijan kanssa täytyy sopia ennen luentoja, mitä esitysmateriaaleja hän haluaa ja pystyy käyttämään luennoillaan ja vaatiiko tallenteen tekeminen muutoksia luennointityyliin. Näistä asioista pystytään sopimaan parissa palaverissa, jolloin suunnittelu-aika jää hyvin lyhyeksi.

Toista ääripäätä edustaa itsenäinen ja kertaluonteinen verkkomultimediateos, jossa käytetään rikasta mediasisältöä ja suurin osa sisällöstä tuotetaan alusta lähtien projektin aikana.

Kun valmista pohjaa ei ole, kaikki pitää tehdä lähtien tyhjältä pöydältä. Erityisesti video- ja animaatiotuotanto vaativat tarkat käsikirjoitukset, koska korjaukset ja muutokset ovat jälkikäteen hankalia tehdä. Muutenkaan tässä vaiheessa ei kannata hakea aikasäästöjä, koska huonosti tehty suunnittelu kostautuu moninkertaisesti tuotantovaiheessa.

Käsikirjoitusvaiheessa tarkennetaan ehkä jo ideointivaiheessa luotua rakennekaaviota ja päätetään navigointiratkaisuista eli asetetaan eri asiasisällöt oikeisiin suhteisiin keskenään. Rakenteen kuvaus, niin tärkeä kuin se onkin, ei kuitenkaan vielä riitä tuotantovaihetta varten, vaan tarvitaan konkreettinen ruutusuunnitelma: mitä kullakin itsenäisellä ruudulla tai näyttönäkymässä on ja tapahtuu. Tässä vaiheessa on huomioitava mahdolliset eri loppuformaattit. Jos loppuformaattit eroavat sisällöltään tai toiminnallisuuksiltaan toisistaan, yhtäläisyydet ja eroavaisuudet on kirjattava selvästi, jotta tuotannossa pystytään välttämään samojen sisältöjen tuottaminen tarpeettomasti kahteen kertaan ja osataan varautua eroavaisuuksien vaatimaan työmäärään.

Käsikirjoitusta tehtäessä on usein ongelmana se, kuinka paljon mitään mediatyyppiä käytetään eri tuotteen osissa. Animaatio ja liikkuva kuva ovat tehokkaita huomion kiinnittäjiä ja luovat tehokkaasti elämyksiä, mutta niiden tuottaminen on kallista. Poikkeustapauksia lukuun ottamatta oppimateriaalituotantoprojektien resurssit ovat rajalliset, jolloin kalliiden mediaelementtien käyttö pitää harkita ja kohdentaa tarkoin. Luukkonen käyttää kolmea eri ilmaisutasoa selittämään, kuinka vaativaksi ilmaisu eri kohdissa lopputuotetta mitoitetaan: fiktiotaso, faktataso ja detaljitaso [Luukkonen 2000].

Fiktiivinen ilmaisu pitää sisällään kaikkein vaativimmat ratkaisut ja se on tarkoitettu käytettäväksi katselijan kiinnostuksen herättäjänä ja elämysten luoja. Keinoina käytetään video- ja animaatiosekvenssejä, nopeita kuvaleikkauksia ja rikasta äänimaisemaa. Fiktio-tason tehtävä on sama kuin elokuvan esittelyvideon: se kertoo olennaisen elokuvasta, mutta jättää kaiken niin auki, että katsojan on pakko nähdä koko filmi.

Faktatasolla tehdään mediaelementeistä sellaista kokonaisvalintaa, jonka seurauksena ruudun tai WWW-sivun asiasisältö välittyy selkeästi katselijalle, ei tunnelma tai tehokeinot. Selkeys saavutetaan määrämuotoisuudella, loogisella rakenteella, sommittelulla ja jäsen-tyneellä asioiden esitystavalla. Katselijan huomio kiinnitetään sisältöön, ei ulkoisiin seik-

koihin. Selkeä rakenne helpottaa myös ylläpito-, päivitys- ja muutuskustannusten pitämistä kurissa ja mahdollistaa dynaamisen sisällön käytön esityksessä fiktiotasoa paremmin.

Detaljitaso on kaikkein lähimpänä tietokantaa jo ilmaisultaankin. Se on multimediatuotteessa yleensä lisäominaisuutena tarjoten esimerkiksi sanaston tai yksityiskohtaista informaatiota asiasisällön osista. Tieto tarjotaan katselijalle luettelomaisesti ja täysin määrämuotoisena. Hypertekstuaalisuus mahdollistaa älykkäästi käytetyn detaljitason, joka lisää tuotteen sisällöllistä rikkautta ja monipuolisuutta huomattavasti.

Suunnitteluvaiheen lopputuloksina syntyy yksi tai useampi käsikirjoitus riippuen lopputuotteen laajuudesta ja tarvittavan uustuotannon määrästä. Asiakäsikirjoitus sisältää sisältömateriaalin yksityiskohdat, eri mediaelementtien käytön ja pedagogiset ratkaisut. Asiakäsikirjoitus kannattaa laatia ryhmässä, johon kuuluu käsikirjoittajan lisäksi ainakin pedagogi, sisällön asiantuntija ja visuaalisen ilmaisun asiantuntija. Laajoissa projektityyppisissä tuotannoissa tarvitaan lisäksi tuotantokäsikirjoitus, joka määrittelee toteuttajat, toteutustavat, aikataulut, kustannukset sekä projektin seurannan ja hyväksymismenettelyt. Samassa yhteydessä on syytä laatia ohjeistus metatiedon keräämiseen tuotannon aikana eri mediaelementeistä. Metatiedon tärkeys korostuu tuotettaessa dynaamista sisältöä, tästä enemmän luvussa 3.

## **Toteutus**

Toteutusvaiheessa käynnistyy varsinainen tuotanto suunnitteluvaiheessa luotujen käsikirjoitusten ja suunnitelmien pohjalta. Multimediatuotannosta pätee erityisen hyvin sanonta ”hyvin suunniteltu on puoliksi tehty”. Kun graafikoilla, kuvaajilla, äänittäjillä ja ohjelmoijilla on tarkka, mutta ei kuitenkaan toteutusta liikaa sitova, käsikirjoitus toteutuksen pohjana, itse toteutus sujuu suoraviivaisesti ilman arvuutteluja siitä, mitä oikeasti pitäisi tehdä. Tuotantovaiheessakin tulee eteen yllätyksiä ja ongelmia erityisesti laajoissa tuotannoissa, mutta ne ovat ratkaistavissa, jos peruskonsepti on kunnossa ja tuotantotiimillä on käsikirjoittajien kanssa yhteinen näkemys siitä, mitä ollaan tarkalleen ottaen tekemässä.

Animaatio ja videokuva ovat suhteellisesti eniten tuotantoaikaa vievät mediaelementit multimediatuotteessa ja siksi niihin liittyvät muutokset tulevat kaikkein kalleimmiksi tehdä toteutusvaiheessa. Parhaassa tilanteessa iteraatiokierroksia ei tarvita yhtään, mutta jos

suunnitteluvaiheessa ei ole jostakin syystä pystytty tekemään tarkkaa suunnitelmaa toteutuksesta, iteraatiokierrosten lukumäärä tulee saada pidettyä mahdollisimman pienenä ja prototyyppien vaatima työmäärä rajallisena, jottei koko tuotantoprojektin aikataulu ja budjetti karkaa käsistä.

Toteutusvaiheessa tuotetaan yksittäiset mediaelementit, yhdistetään ne ja tuotetaan ohjelmisto, joka huolehtii multimediatuotteen interaktiivisista ominaisuuksista. Tarvittavan ohjelmointityön määrä riippuu paljon tuotteen luonteesta: perinteisen multimediatuotteen voi tehdä multimediatehityksellä hyvin pienellä ohjelmointityöllä, kun taas tietokantapohjaisen, dynaamiseen sisältöön perustuvan tuotteen toteutus vaatii runsaasti ohjelmistokehitystä. Tämä tulee huomioida sekä aikataulutuksessa että resurssoinnissa.

Toteutusvaiheen dokumentointi ja sen systemaattisuus korostuu tuotettaessa dynaamiseen sisältöön perustuvaa multimediaoppimateriaalia. Erityisesti video- ja äänituotannossa tarkan otoskohtaisen kirjanpidon tärkeys on nähty jo pitkään. Uutena vaatimuksena on tullut tämän otoksiin ja medialeikkeisiin sidotun kuvailevan ja luokittelevan tiedon käsittely sellaisessa digitaalisessa muodossa, että sitä voidaan hyödyntää ilman manuaalisia toimenpiteitä aina loppusovellukseen saakka. Kun tämä tieto kulkee medialeikkeen mukana koko tuotantoprosessin läpi, multimediatuotteessa tarvittavien leikekohtaisten tietojen syöttö esimerkiksi tietokantaan tapahtuu merkittävästi pienemmällä vaivalla kuin tilanteessa, jossa jonkun pitää katsella leikkeet läpi ja kerätä käsin samat tiedot.

Tuotantovaiheessa syntyy niin kuvista kuin ääni- ja videoleikkeistä master-versiot eli versiot, jotka ovat laadultaan parhaat ja sopivat jatkomuokkaukseen. Näitä versiota ei voida käyttää loppuformaateina, koska ne eivät välttämättä ole sellaisessa tiedostomuodossa, jonka loppukäyttäjä pystyisi avaamaan, ja ne vievät liikaa tallennuskapasiteettia. Näistä master-versioista muunnetaan ja pakataan jakeluformaatit, jotka soveltuvat erinomaisesti katselukäyttöön, mutta joiden muokattavuus on erittäin heikko usein aivan tarkoituksella. Master-versioiden arkistointi on syytä huolehtia jo tuotannon aikana, jotta ne eivät vain häviä projektin päätyttyä. Ulkopuolisella sisällöntuottajalla teetetyistä materiaaleista osataan yleensä vaatia alkuperäisversiot varsin hyvin, mutta master-versiot ovat aivan yhtä tärkeitä omatuotannossakin, oli se kuinka vaatimatonta tahansa. Kuviin tai leikkeisiin voidaan joutua myöhemmin tekemään muutoksia, niitä voidaan käyttää jossakin toisessa tuo-

tannossa ja erityisesti verkkomultimedian tapauksessa äänen ja videon pakkausmenetelmät kehittyvät hurjaa vauhtia, jolloin jossakin vaiheessa voi tulla eteen loppuformaatin vaihto laadultaan parempaan. Näissä tilanteissa helposti löytyvät ja hyvin dokumentoidut master-versiot, olivat ne tiedostoina tai nauhoina, ovat arvokkaita.

## **Jakelu**

Multimediatuote jaetaan loppukäyttäjälle joko optisen median (CD-ROM tai DVD) tai verkon välityksellä. Myös yhdistelmä on mahdollista eli raskaimmat ja muuttumattomat sisältöosat käytetään paikallisesti ja päivitettävä osuus ladataan tietoverkon kautta. Tätä on käytetty esimerkiksi multimediatietosanakirjoissa.

Optisen median käyttö jakeluvälineenä tuo mukaan tiettyjä etuja. Vaatimukset asiakaspään laitteistolle ovat pienemmät kuin verkkosovelluksessa, koska tietoverkkoyhteyttä ei tarvita ja medialeikkeet voidaan tallettaa heikommalla pakkauksella, jolloin niiden katseluun riittää hiukan vanhempikin tai tehottomampi tietokone. Toteutustekniikoiksi voidaan myös valita sellaiset, jotka ovat varmatoimisempia ja vakaampia kuin tukeutuminen koko ajan päivittyvien WWW-selainten ja niiden laajennusten ominaisuuksiin. Optinen media on myös luotettavampi, yhden palvelimen tai kriittisen verkkoyhteyden ongelmat eivät välity kaikille multimediatuotteen käyttäjille.

Dynaaminen ja usein päivittyvä multimediasisältö pitää jakaa verkon välityksellä, jotta sisältöjen vaihtaminen olisi mahdollista. Toinen verkkomultimediasovellusten kiistaton etu on se, että kun eri päätelaitteet on otettu huomioon toteutuksessa, sama sisältö on katseltavissa sekä työasemalta kotoa että kämmentietokoneelta tai älypuhelimella mobiilisti.

Jaettaessa multimediasisältöjä verkon välityksellä tulee huomioida sisällön resurssitarpeet, jotka ovat huomattavasti suuremmat kuin tavallisen WWW-palvelun tapauksessa. Yksinkertaisin mitata ja ratkaista on levykapasiteetin tarve palvelimelle tai palvelimilla, joiden välityksellä mediaelementit jaetaan. Huomattavasti kriittisempi ja vaikeammin mitoitettavissa on siirtokaistan määrä, joka varataan palvelinpäähän. Jos sisällön katseleminen sujuvasti vaatii noin 250kbit/s siirtokaistan yhtä katselijaa kohden, niin jo 20 yhtäaikaista katselijaa vaatii laskennallisesti 5Mbit/s siirtokaistaa palvelimella, käytännössä jonkin verran enemmän johtuen siirtotien aiheuttamasta purskeisuudesta eli siitä, että dataa siirtyy välillä

enemmän ja välillä vähemmän kuin sisällön nimellisoikeus ilmoittaa. Yhtäaikaisten katse-  
lijoiden maksimimäärää on hyvin vaikea arvioida ja varautuminen pahimpaan tilanteeseen  
on usein taloudellisesti mahdotonta, koska ylimääräinen siirtokaista ja mahdollisesti mak-  
sulliset käyttämättömät käyttäjälisenssit mediapalvelimelle tulevat kalliiksi. Jos yhtäai-  
kaisten käyttäjien määrä vaihtelee paljon, kannattaa harkita mediapalvelinpalvelujen os-  
tamista operaattorilta tai muulta palveluntarjoajalta ja sijoittaa raskaat mediaelementit pal-  
veluntarjoajan palvelimelle, jos valitut mediaformaattit sen sallivat. Vähemmän siirtokais-  
taa kuluttavat teksti-, kuva- ja äänileikkeet voidaan pitää omalla palvelimella vähän hi-  
taammankin yhteyden päässä.

Dynaamisen sisällön käyttö ei sulje täysin pois tuotteen jakelua optisena medianä. Täysin  
vastaavan toiminnallisuuden toteuttaminen ilman verkon käyttöä on hankalaa, koska kaik-  
ki palvelinohjelmistot tulisi asentaa paikallisesti, mutta staattisen version luominen tuot-  
teesta on mahdollista. Tällöin esimerkiksi DVD-levylle talletetaan kertaos sen hetkisestä  
sisällöstä ja rakenteesta varsinaisella tuotantopalvelimella ilman interaktiivisia toiminnal-  
lisuuksia. Sisällön adaptiivisuutta käyttäjän valintoihin voidaan toki simuloida tiettyyn  
rajaan asti tallentamalla kaikki eri variaatiot mahdollisista näkymistä, mutta jos adaptiivi-  
sia komponentteja on paljon, eri variaatioiden lukumäärä kasvaa eksponentiaalisesti eikä  
niiden kaikkien tallentaminen ole käytännössä mahdollista.

Staattisen version luominen dynamisesta sisällöstä ja rakenteesta vaatii oman toimintalo-  
giikkansa toteuttamisen verkkomultimediasovellukseen. Teksti- ja kuvasisällöt saadaan  
talletettua offline-katselumuotoon sopivilla työkaluohjelmistoilla, jotka käyvät hakurobo-  
tin tavoin kaiken sisällön läpi, tallettavat tiedostot levyille ja muokkaavat linkit vastaamaan  
levylle talletettua hakemistorakennetta. Ääni- ja videoleikkeet tuottavat kuitenkin ongel-  
mia näille työkaluille, sillä mediaelementeihin ei viitata suoraan HTML-sivulta vaan  
yleensä ns. metatiedoston avulla. Metatiedosto pitää sisällään osoitteen varsinaiseen me-  
dialeikkeen sijaintipaikkaan, joka voi olla mediapalvelin jonkun muun siirtoprotokollan  
kuin WWW-sivujen siirtämiseen käytetyn HTTP-protokollan takana. Vaikka työkalu  
osaisikin etsiä metatiedostoista medialeikkeiden oikean sijaintipaikan, niiden lataaminen  
ja tallentaminen levyille on vaikeaa, koska niitä ei ole tarkoitettu tallennettavaksi levyille ja  
itse asiassa suoratoistotuotteiden valmistajat pyrkivät estämään tallentamisen kaikin kei-  
noin.

Jakelukanavan valinta vaikuttaa koko tuotantoprosessin aikataulutukseen. Verkkopohjainen multimediatuote voidaan siirtää päivässä tai parissa kehitysympäristöstä tuotantoympäristöön lopullista testiä varten, jonka jälkeen tuote on kohdeyleisön käytettävissä välittömästi. Optisen median tapauksessa puhutaan kertaluokkaa suuremmista ajoista, sillä levyjen painattaminen ja toimittaminen asiakkaille voi kestää viikkoja.

## **Muuta huomioitavaa**

Tehtäessä täysin uudentyyppistä materiaalia tuotantoryhmällä, jonka kaikilla jäsenillä ei ole aikaisempaa kokemusta animaatio- tai videotuotannosta, projektiin on syytä varata resursseja kevyen prototyypin tekemiseen. Prototyypin avulla kaikki osalliset saavat yhteisesti hyväksytyyn ja ymmärrettyyn kuvan siitä, kuinka tuotantoprosessi etenee, mitä vaaditaan missäkin vaiheessa ja mitä lopputuloksia on odotettavissa. Samalla löytyvät tuotantoprosessin loppupuolelta mahdolliset sudenkuopat, joita ei osattu nähdä ennalta ja jotka kaataisivat koko projektin, jos sama olisi tullut vastaan lopullista sisältöä tuotettaessa.

Tuotteen kieliversiointi saattaa tuntua kaukaiselta asialta suunniteltaessa ja toteutettaessa multimediatuotetta ensimmäistä kertaa, jolloin päällimmäisenä tavoitteen on saada toimiva tuote edes yhdellä kielellä. Jos tarve toisen kieliversioon on ennakoitavissa ja vaikka sitä ei päätettäisikään ensivaiheessa toteuttaa, kieliversiointiin voidaan ja pitääkin varautua. Kuvamateriaalissa kielisidonnaiset komponentit kannattaa pitää erillään muista graafisista komponenteista, jotta kielisidonnaiset komponentit olisivat helposti vaihdettavissa. Videomateriaalin osalla varautuminen tarkoittaa yksinkertaisimmillaan sitä, että ääni pidetään erillään videokuvasta, jolloin selostusääniraita voidaan korvata toisella videokuvaan puuttumatta. Käytännössä ääniraidan vaihtaminen ei välttämättä ole aivan yksinkertaista varsinkaan silloin, jos selostusääni on synkronoitu vahvasti videokuvaan, sillä eri kielissä samojen asioiden ilmaisemiseen kuluu erimittainen aika. Luvussa 6 esiteltävässä Metsäkonesimulaatio-opetuksen tuotteistaminen -projektissa tämä tuli vastaan siten, että englanninkieliset selostukset oli vaikea saada mahtumaan samoihin aikaikkunoihin aikaisemmin äänitettyjen ja videoeditoinnissa käytettyjen suomenkielisten selostusten kanssa.

Multimediatuotteen valmistamisessa käytetään tyypillisesti hyvin monen henkilön osamista sisällön tuottamisessa. Osa sisällöistä tuotetaan alusta loppuun tuotantoprojektissa, mutta tuotteeseen saatetaan sisällyttää huomattava määrä myös muualta hankittu tai saatua sisältömateriaalia joko sellaisenaan tai muokattuna. Tämä asettaa suuret vaatimukset tekijänoikeuksien hallinnalle: kaikkien sisällöntuottajien tai oikeuksien haltijoiden kanssa täytyy laatia tekijänoikeussopimukset, joissa määritellään tarkasti, miten ja missä yhteyksissä sisältömateriaaleja voidaan käyttää. Tarkoilla sopimuksilla estetään hankalat riitatapaukset yksittäisten mediaelementtien käytöstä myöhemmässä vaiheessa.

## **2.4. Video verkkomultimediassa**

Videokuva on vaativin mediatyyppi kaikessa multimediassa sekä sisällöntuotannon että siirron ja käsittelyn näkökulmasta, mutta erityisesti se on haastava verkkomultimediassa. Videomuotoinen sisältö vaatii muihin mediatyyppeihin verrattuna moninkertaisesti tallennus- ja tiedonsiirtokapasiteettia. Jos unohdetaan yliopistojen rakentamat tutkimuskäyttöön tarkoitetut suurikapasiteettiset videonsiirtoverkot, verkkomultimediassa joudutaan aina tekemään kompromissi videon laadun ja sen käytettävyyden välillä. Jokainen haluaisi katella verkkovideosa DVD-elokuvan kuvanlaadulla, mutta suurimmalta osalta käyttäjistä puuttuu sellainen internet-yhteys, jolla se olisi mahdollista tinkimättä käytettävyydestä.

Verkkovideon käytettävyyden yksi tärkeimmistä kriteereistä on käyttäjän kokemaa viive katselun aloituspäätöksestä videon käynnistymiseen. Nopeimmillaankin viive on verkosta ladattavan videosisällön kohdalla luokkaa 1 - 5 sekuntia riippuen soitto-ohjelmasta ja videoformaattista. Aloitusviive syntyy ennen kaikkea videodatan puskuroinnista (buffering) välimuistiin soitto-ohjelmassa [Microsoft 2004a]. Puskurointi on välttämätöntä, jotta internet-yhteyksillä videon sujuva katseleminen ilman toistuvia hetkellisiä katkoksia olisi mahdollista. Puskurointi antaa pelivaraa tietoverkon hetkellisiä häiriöitä ja datan epätasaisista siirtovuota vastaan.

Puskurin täyttymisnopeuteen ja myös toiston sujuvuuteen vaikuttaa videoleikkeen bittinopeuden (bit rate) suhde internet-yhteyden siirtokaistaan (bandwidth). Videoleikkeen bittinopeus kertoo, kuinka monta bittiä sekunnissa tarvitaan dataa videokuvan ja äänen esittämiseen. Pakkaamattoman videoleikkeen, jonka kuvakoko on neljäsosa TV:n kuvakoosta



eli 360x288 (leveys x korkeus kuvapisteinä) ja kuvanopeus 25 kuvaa sekunnissa, bittinopeus on yli 62 miljoonaa bittiä sekunnissa eli 62 Mbit/s. Kun muistetaan, että normaalin laajakaistaliittymän nopeus on alle 1Mbit/s puhumattakaan modeemiyhteyksistä, huomataan, että videokuvaa täytyy pakata runsaasti, jotta sen katselu onnistuisi reaaliaikaisesti eli samaa tahtia vastaanotetun videodatan kanssa. Videokuvaa voidaan pakata tiettyyn rajaan saakka ilman, että ihmissilmä huomaa pakkauksesta johtuvaa kuvatiedon häviämistä. Tämä raja tulee kuitenkin jo aikaisemmin vastaan kuin mitkä ovat normaaleja internet-yhteyksien nopeuksia kotikäytössä, jolloin kysymys kuuluukin, kuinka paljon katselija kestää videokuvan laadun heikkenemistä.

Jotta puskurointia ja siten videon soittamisen pysähtymistä ei tapahtuisi enää toiston käynnistyttyä, videoleikkeen bittinopeuden pitää olla pienempi kuin internet-liittymän siirtokaistan suuruus. Katselija, jonka ADSL-liittymän maksiminopeus on 512kbit/s, voi alkaa katsella 1 Mbit/s-bittinopeudelle pakattua videoleikettä, mutta minuutin videoleikkeen katseluun kuluu noin kaksi minuuttia ja puolet ajasta videoleike on pysähdyksissä, kun soitto-ohjelma täyttää datapuskuria. Jos edellä kuvattu valinta on vapaaehtoinen, asia on kunnossa ja katselija on valinnut tietoisesti paremman kuvanlaadun katselun sujuvuuden kustannuksella. Mutta jos katselija onkin odottanut näkevänsä videoleikkeen sujuvasti ilman katkoksia toistossa, asiakastytyväisyys jää heikoksi ja todennäköisesti katselija ei katso ensimmäistäkään tarjottua videoleikettä loppuun saakka, vaan vaihtaa sisällöntarjoajaa. Tämä takia sisällöntuottajan tulee pystyä tarjoamaan kaikille tuotteen tärkeimmille kohderyhmille heidän yhteysnopeuksiaan vastaavat versiot videosisällöistä joko automaattisesti tai käyttäjän valintoihin tukkeutuen, tai ainakin kertoa selvästi, että hitailla yhteyksillä joutuu sisältöjä odottamaan.

Mitä pienempää bittinopeutta videoleike käyttää, sitä voimakkaammin videokuvaa pitää pakata ja sitä enemmän videokuvan laatu kärsii. Videokuvan laatua voidaan mitata teknisesti kuvanopeuden (frame rate), resoluution ja kuvan tarkkuuden avulla. Videoleikkeen laadukkuuden päättää kuitenkin katselija itse ja tässä ihmiset ovat erilaisia. Esimerkiksi kuvanopeutta on yleisesti pidetty kriittisempänä ominaisuutena kuin kuvan tarkkuutta eli kuvassa näkyvien yksityiskohtien määrää, mutta Englannissa PDA-laitteilla tehdyssä tutkimuksessa [McCarthy et al. 2004] todettiin, että jalkapalloutisia katselleet testihenkilöt tinkivät enemmän kuvanopeudesta kuin kuvan terävyydestä.

Muunnettaessa videoleikettä suoratoistomuotoon tärkein valittava parametri on bittinopeus tai -nopeudet, jotka soveltuvat kohdeyleisölle. Valittujen bittinopeuksien puitteissa optimoidaan kuvan laatu lähinnä säätämällä kuvakokoa, joka vaikuttaa suoraan pakatun videon kuvanopeuteen. Seuraavassa taulukossa on esitetty RealMedialle [Real 2004b] annettu ja suosituskuvakokoja eri yhteysnopeuksille, mutta samat suositukset koskevat myös muita suoratoistoformaatteja. Taulukon bittinopeus-sarakkeessa olevat arvot pitävät jo sisälleen äänen pakkaukseen tarvittavan bittinopeuden. Ääni-sarakkeessa kunkin rivin pienempi lukema tarkoittaa puheäänisisältöä ja suurempi musiikkisisältöä.

<b>Bittinopeus</b> kbit/s	<b>Kuvakoko</b> leveys x korkeus	<b>Kuvanopeus</b> kuvaa sekunnissa	<b>Äänen tarvitsema</b> <b>bittinopeus</b>
alle 56	176 x 144	alle 10	4 - 8
56 - 256	176 x 144	12 - 15	8 - 16
256-768	320 x 240	15 - 25	64 - 96
yli 768	640 x 480	25	64 - 96

Taulukko 1 Suosituksia videoleikkeen kuvakoolle ja kuvanopeudelle eri yhteysnopeuksille.

Taulukon lukuarvoja ei voi noudattaa orjallisesti, sillä videosisällöt ovat erilaisia ja ne pakkautuvat siten myös eri tavalla. Puhuvan pää tarvitsee huomattavasti vähemmän bittejä hyvään kuvanlaatuun kuin musiikkivideo tai urheiluvideo. Oman videoleikkeen optimaaliset pakkausparametrit täytyy hakea loppujen lopuksi kokeilemalla.

Videoleikkeen laatu riippuu bittivirtanopeuden ja kuvakoon lisäksi oleellisesti pakkausmenetelmästä, jolla videokuva on pakattu. Videokuvan ja äänen pakkauksen yhteydessä puhutaan käsitteestä koodekki (codec), joka tarkoittaa videokuvan pakkaamiseen ja purkamiseen käytettävää algoritmiparia. Koodekkien kirjo on kaiken kaikkiaan hyvin suuri, mutta suoratoistokäyttöön soveltuvia yleiskäyttöisiä koodekkeja on suhteellisen pieni määrä ja useat niistäkin on sidottu johonkin suoratoistovideoformaattiin. Tästä ei kuitenkaan pidä tehdä sitä virheellistä päätelmää, että videoformaatti tai jopa tiedostopääte kertoisi käytetyn pakkauskoodekin, sillä videoformaatti ja videokoodekki ovat kaksi täysin eri asiaa: Videoformaatti on kuin juna, joka tarjoaa videotalle määrämuotoisen siirtoalustan, mutta se ei ota kantaa siihen, miten videodata on pakattu vaunuihin. Tosin vastaanot-

tajalle 'junan' pitää pystyä kertomaan 'rahtikirjassa' eli videoleikkeen otsikkotiedoissa käytetyn koodekin, jotta vastaanottaja eli soitto-ohjelma osaa valita oikean (purku-)koodekin videokuvan näyttämiseksi.

Tärkeimmät yleiskäyttöiset verkkovideoformaattit ovat MPEG-4, Quicktime, RealMedia ja Windows Media. Näistä MPEG-4 on ainoa standardoitu formaatti, muut ovat valmistaja-kohtaisia formaatteja. Lisäksi verkkomultimediassa on käytössä joitakin sovelluskohtaisia videoformaatteja kuten esimerkiksi Macromedian Flash-animaatioiden oma videoformaattinsa.

Real Networksin RealMedia ja Microsoftin Windows Media ovat eniten käytetyt suoratoistovideoformaattit verkossa. Ominaisuuksiltaan ne ovat hyvin tasaväkisiä ja valinta niiden välillä pitää tehdä muilla syillä kuin itse formaattien teknisiin ominaisuuksiin liittyvillä perusteilla. Jos käyttäjäkunta on puhtaasti Windows-käyttäjiä ja sisällöntuottaja on sitoutunut Windows-palvelimiin, Windows Median valinta on helppo perustella jo kustannussyillä, mutta jos katselijakunnasta löytyy myös Linux- ja Mac-käyttäjiä, vaakakuppi saattaa kääntyä RealMedian suuntaan. Kolmas ja turvallisin vaihtoehto yleisille videosisältöpalveluille on tukea molempia formaatteja. Videokuvanlaatua näiden formaattien ja niissä käytettävien koodekkien välillä voi verrata eri bittinopeuksilla esimerkiksi Sonera Medialabin esittelysivuilla [Sonera 2004].

Applen Quicktimella on vahva asema multimedian historiassa ja historian noste on kantanut myös verkkomultimediapuolelle. Quicktime oli jo menettämässä suoratoistomarkkinat RealMedialle ja Windows Medialle, mutta Applen vahva rooli MPEG-4:n standardoinnissa ja Quicktimen hyvä tuki syntyneelle suoratoistostandardille ovat tuoneet uutta uskoa Quicktimen säilymiselle varteenotettavana video- ja multimediaformaattina myös tulevaisuudessa. Erityisesti Quicktimen tuki kolmannen sukupolven matkapuhelimissa käytetylle 3GPP-videostandardille on tuonut uusia Quicktime-käyttäjiä, kun PC-käyttäjät haluavat katsella multimediamatkapuhelimilla tallennettuja videoleikkeitä erityisesti Aasiassa [Digit 2004].

MPEG-standardiperheeseen kuuluvan MPEG-4:n asema suoratoistomarkkinoilla on vielä heikko, vaikkakin sen uskotaan parantuvan tulevaisuudessa. Tällä hetkellä sitä ei voida

suositella vielä tuotantokäyttöön, ellei kohderyhmä ole mobiililaittevoittainen. Muidenkin kannattaa kuitenkin seurata formaatin yleistymistä. MPEG-4:n vahvuus on sen laaja tuki erilaisille käyttökohteille, suoratoistomarkkinoiden valtaus saattaa tapahtua televisioyhtiöiden kautta, joista viimeisetkin ovat siirtymässä analogiatekniikasta digitaalitekniikkaan ja tällä alueella MPEG-4 on jo haastanut MPEG-2:n erityisesti korkearesoluutioisessa teräväpiirtovideossa (HDTV) [StreaminMedia 2004].

### **3. Sisällön metatiedon tallentaminen ja hyödyntäminen**

Metatieto määritellään yleensä ”tiedoksi tiedosta” (data about data). Termi ”meta” tulee kreikasta ja tarkoittaa muutosta. Metatieto on Internet-ajan termi kuvailutiedolle, joka on yleisesti yhdistetty nimenomaan verkkodokumenttien kuvailuun. Metatietoa on käytetty aikaisemminkin, mutta kiinnostus siihen on herännyt sähköisten julkaisujen ja Internetissä olevan laajan kuvailemattoman aineiston ansiosta. Metatiedon avulla tiedonhakuja saadaan tarkemmiksi, mutta toisaalta metatieto auttaa sisällöntuotannossa suurten tietomäärien käsittelyä ja uudelleenkäyttöä.

Metatieto on dataa/tietoa dokumentin ominaispiirteistä, sisällöstä, alkuperästä, muutoksista ja käytöstä. Tätä tietoa voidaan käyttää moniin tarkoituksiin: tiedonhakuun, dokumenttien muutosten tai käytön seuraamiseen.

Metatieto voi olla termejä, tekstiä tai kuvia, jotka kuvaavat yksittäisiä tieto-olioita ja niiden välisiä suhteita. Tieto-olio voi olla kokonainen tekstidokumentti, kuva, äänileike tai video, tai vain tietty osa niistä. Tieto-olio voi olla esimerkiksi videotallenne luennosta. Metatietoja videotallenteesta ovat kuvauspäivä ja -aika, puhujien nimet, kurssin nimi, luennon aihe ja käytetyt esitysmateriaalit. Videotallenne voidaan jakaa edelleen aiheosuuksiin, joihin määritellään omat metatietonsa, kuten aiheosuuden alku- ja loppuaika, kuvaus käsiteltävästä aiheesta ja aikajakson aikana puhuneet henkilöt. Toinen, perinteisempi esimerkki metatietojärjestelmästä on kirjaston kortisto, joka sisältää kuvailevaa tietoa kirjasta tai muusta kirjastossa olevasta tallenteesta: tekijä, nimeke, julkaisuvuosi, aihe ja sijaintitieto, jonka avulla kohde löytyy hyllystä.

Metatietoa voidaan tuottaa sekä automaattisesti että manuaalisesti. Automaattisesti dataa saadaan dokumenttien tai medialeikkeiden ominaisuustiedoista, dokumenttien sisällöstä rakenteisuutta hyödyntäen, sisällönhallintajärjestelmistä ja työkaluohjelmista. Metatiedot

joudutaan syöttämään manuaalisesti, kun halutaan lisätä itse kirjoitettu kuvaus tai kun automaattinen tuotto ei ole mahdollista tai riittävän täydellistä.

Metatieto voidaan tallentaa yhdessä tieto-olion kanssa jopa samaan tiedostoon tai metatieto voidaan säilyttää täysin eri paikassa kuin varsinainen tieto-olio, esimerkiksi erillisellä metatietopalvelimella omassa tietokannassaan tai kirjasto-esimerkin tapauksessa erillisessä kortistossa. Erilaisia metatietoformaatteja on olemassa lukuisia ja ne ovat usein sovellusala- ja tieto-oliotyyppikohtaisia. Yleisistä sovellusaririippumattomista metatietoformaateista eniten käytetty on Dublin Core, josta lisää hiukan edempänä. Yleisten metatietoformaattien suurin etu on se, että ne toimivat hyvin metatiedon siirtoformaateina järjestelmästä toiseen. Näin tieto-olioita, esimerkiksi mediaelementtejä, voidaan siirtää järjestelmästä toiseen metatietoineen riippumatta käytetyistä ohjelmistoista ja sisäisistä metatiedon säilytystavoista. Samoin yhteinen metatietorajapinta mahdollistaa metatietokantojen yhdistämisen riippumatta siitä, millä tekniikalla kukin metatietokanta on toteutettu. Tällöin tarvitaan yhteisen metatietoformaatin lisäksi myös yhteinen termistö ja sen sisälle sovitut riippuvuudet, jotta myös eri järjestelmissä käytetyt termit vastaisivat toisiaan.

Multimedia-metatiedot jaetaan kolmeen luokkaan: mediakohtainen metatieto, median käsittelyyn liittyvä metatieto ja sisältöön liittyvä metatieto. Mediakohtainen metatieto on teknistyyppistä tietoa mediaelementistä, kuten tiedostoformaatti, kuvakoko, käytetty pakkauskoodekki äänelle ja videolle sekä medialeikkeen kesto. Nämä tiedot pystytään selvittämään medialeiketiedostoista yleensä ohjelmallisesti ilman manuaalisesti tapahtuvaa tietojen syöttöä. Käsittelyyn liittyviä tietoja syntyy mediaelementin elinkaaren aikana, kuten luonti- ja muokkauspäivämääriä, versiointitietoja, käyttöoikeustietoja, prosessointitietoja ja työvaihetietoja. Sisällöllisellä metatiedolla kuvataan mediaelementin aihepiiri.

Metatietoa hyödynnetään eniten hakujen suorittamiseen suuresta tietomassasta. Teksti- ja kuvamateriaaleista voidaan tosin tehdä hakuja varsin hyvin ilman sisällön kuvailua ja avainsanoja, mutta hakutulokset ovat tällöin varsin jäsentymättömiä kuten kaikki Internetin hakukoneita käyttäneet tietävät. Ääni- ja erityisesti videomateriaalille ei voida edes tehdä täsmällisiä sanahakuja, sillä automaattiset sisältöanalyysimenetelmät eivät ole vielä riittävän kehittyneitä mahdollistamaan sisältöhakuja ilman erillistä kuvailua. Metatietoa

hyödyntämällä voidaan tehdä täsmällisiä hakuja esimerkiksi luentotallennearkistosta luennoijan, aihealueen ja asiasanojen avulla.

Tietojen hakemisen ja käytön kannalta on välttämätöntä, että tiedot tallennetaan jollain yhtenäisellä tallennusperiaatteella, joka on esiteltävissä myös tietojen hakijalle. Yhtenäisiä tallennusperiaatteita varten on suunniteltu metatietostandardeja, kuten edellä mainittu Dublin Core. Esimerkiksi kirjastojen kortistot tai museoiden esineluettelot ovat metatietojärjestelmiä. Ne sisältävät kuvailevaa tietoa kuvailun kohteesta: kirjasta, tallenteesta tai museossa olevasta esineestä.

Dynaaminen multimediaesityksen koostaminen edellyttää sisällön rakenteistamista ja sen kuvailua metatiedon avulla. Rakenteistamisen avulla multimediasisältö jaetaan pienempiin osiin, joihin voidaan viitata suoraan ja joista tallennetaan tarvittavat tiedot mm. mahdollistamaan mediaelementin liittäminen osaksi suurempaa kokonaisuutta ja hakutoimintoja varten. Rakenteistaminen ei tarkoita aikajatkuvien mediatyyppien tapauksessa esimerkiksi videotallenteen jakamista fyysisesti eri osiin, vaan jako on looginen. Tieto video-ohjelmasta ja sen osista aikarajoineen on metatietoa, jonka avulla valitaan katsottavaksi halutut komponentit ja saadaan näistä lisätietoa. Katsottavaksi halutut osat valitaan metatiedon avulla. Segmentointi voidaan tehdä usealla eri tasolla; monitasoisessa jaottelussa tallenteen osia jaetaan edelleen osiin. Esimerkiksi TV-uutisista tehty videotallenne jaetaan ensin aiheisiin, aiheet aiheosuuksiin ja aiheosuudet otoksiin.

### **3.1. Metatietostandardeja multimedial kuvailuun**

Olennaista käytettävän metatietomäärityksen valinnassa on määrityksen soveltuvuus kuvailtavaan kohteeseen. Eri formaatit soveltuvat eri tavoin toisalta laajojen kokonaisuuksiin, esimerkiksi kokonaisen verkkokurssin kuvailuun, ja toisalta yksittäisten, itsenäisten aineistojen kuvailuun. Metatiedon käyttötarkoitus vaikuttaa siihenkin, minkä elementtien tulisi olla pakollisia. Minimitason määrittäminen on tarpeen myös, jotta metatietoa voidaan konvertoida eli muuntaa toiseen formaattiin.

## **Dublin Core**

Metatietostandardeista tunnetuin ja eniten käytetty on Dublin Core [Dublin Core 2004]. Dublin Coren kehitystyö alkoi vuonna 1995 ja tarkoituksena oli tehostaa verkkotallenteiden hakua luomalla väline niiden kuvailuun. Näin syntyi yhä kehittyvä Dublin Core standardi, joka on yksinkertainen mutta tehokas väline erilaisten verkotettujen resurssien kuvailuun. Se koostuu 15:sta elementistä, joita rakentamassa on ollut kansainvälinen, eri alojen ammattilaisista koostuva yhteisö. Mukana on ollut asiantuntijoita kirjastoalalta, tietojenkäsittelytieteestä, museoyhteisöstä ym. tieteenaloilta [Stenvall 1998]. Dublin Core -formaattista on tehty suomenkielinen SFS-standardi.

Dublin Core sallii omien laajennusten tekemisen ja tätä mahdollisuutta on käytetty runsaasti myös multimediamateriaalin kuvailussa, koska perus-Dublin Core soveltuu varsin huonosti aikajakuvien medioiden, äänen ja videokuvan, kuvailuun. Videopohjaisiin sisältöihin keskittyviä laajennuksia ovat tehneet Videnet [Videnet 2001], DSPACE-hanke [DSPACE Metadata 2004] ja CSC Suomessa osana Medar-hanketta [CSC Metadata 2003].

## **LOM**

IEEE:n laatima LOM-määrittely (Learning Object Metadata) [LOM 2004] on tarkoitettu nimenomaan oppisisältöjen kuvailuun ja sitä on tarkoitus käyttää mm. Suomen virtuaaliyliopiston oppimateriaalin metatietojärjestelmänä. TIEKE on vetänyt työryhmää, joka on suomentanut standardin ja laatinut kansalliset soveltamisohjeet standardista. Myös LOMin tallentamiseen on suunniteltu alustaa ja työn tuloksena on kehitetty Metakka-pilotointiympäristö LOM-metatiedon käyttämiseksi [Metakka 2004]. LOM-määrittelyksen mukainen metadata on periaatteessa muunnettavissa Dublin Coreksi.

## **MARC 21**

MARC-standardin (MACHINE-Readable Cataloging) [MARC 2004] kehittäminen aloitettiin Yhdysvaltain kongressin kirjaston tarpeisiin yli kolmekymmentä vuotta sitten. Sen tavoitteena oli luoda mekanismi, jolla tietokoneet pystyisivät vaihtamaan, käyttämään ja tulkitsemaan bibliografista informaatiota. MARC 21 -versio valmistui 1990-luvun lopulla ja se on käytössä hyvin laajalti maailmalla tieteellisissä kirjastoissa, myös Suomessa.



MARC on joustava rakenteinen metatietoformaatti, jolla voidaan kuvailla hyvinkin erilaisia aineistoja: sekä painettuja että digitaalisia julkaisuja, musiikkia, kuvaa, videoita, esineitä ja jopa tapahtumia ja organisaatioita. Multimedian erityispiirteiden kuvaamiseen MARC:issa ei ole kuitenkaan kenttiä, joten se soveltuu lähinnä mediasisältöjen luettelointiin.

## **MPEG-7 ja MPEG-21**

MPEG-7 [MPEG 2004] on ISO-standardi vuodelta 2001 multimediamateriaalin kuvaamiseen metatiedon avulla. Standardin tavoitteena ei ole esittää informaatiota sinällään, vaan tuottaa informaatiota informaatiosta eli kuvauksia audio- ja videodatan sisällöstä. Tässä se siis eroaa merkittävästi vanhemmista MPEG-standardeista. MPEG-1, -2 ja -4 standardit mahdollistavat sisällön saatavuuden tarjoamalla standardoidut tiedosto- ja siirtoformaattit. MPEG-7 mahdollistaa tarvittavan tiedon löytymisen. Sen takia on tärkeää huomata, että MPEG-7 on kohdistettu erilaisille sovelluksille ja erilaisiin ympäristöihin.

MPEG-7:n tavoitteena on tarjota menetelmä luokitella ja kuvata multimedian sisältöä standardoidulla, ohjelmallisesti tehokkaasti käsiteltävissä olevalla tavalla. Semanttisen kuvailun lisäksi MPEG-7 tarjoaa mahdollisuuden kuvailla multimedialeikkeitä myös visuaalisesti ja äänen ominaisuuksien perusteella: mediaobjektin tai otoksen värejä, muotoja, liikkeitä, melodioita ja äänenvärejä. Näiden metatietojen avulla on mahdollista toteuttaa hakukoneita, joissa multimedialeikkeitä voidaan hakea paitsi asiasanojen myös mallikuvien ja ääninäytteiden avulla.

MPEG-7 on standardina varsin nuori ja sen käyttö on vasta yleistymässä. Sen suurin ero ja myös vahvuus verrattuna muihin metatietostandardeihin on sen tuki aika- ja kuvaaluesidonnaiselle kuvailutiedolle. MPEG-7-metadatan tuottamiseen on olemassa joitakin ohjelmia, joista tunnetuin on IBM:n VideoAnnEx-työkalu [IBM 2004]. Työkalu osaa hakea automaattisesti videoleikkeestä yksittäiset otokset, mutta käyttäjä joutuu itse kuvailemaan otokset sekä tekemänsä kuva-alueajaukset. Kuvailun ja luokittelun helpottamiseksi työkalu käyttää sanastoja eli se tukee siten ontologioita.

MPEG-21 on myös metatietostandardi, mutta sen näkökulma on laajempi kuin MPEG-7:n. MPEG-21:n tavoitteena on mahdollistaa lisääntyvä multimedian käyttö läpinäkyvästi

verkkojen yli ja kattaa koko multimedian jakeluketju tuotannosta jakeluun ja kaupankäyntiin. Standardin perusyksikkönä on digitaalinen tuotos (Digital Item), joka yksilöidään ja kuvataan standardissa määritetyllä tavalla. Tuotos voi mikä tahansa mediaelementtien, niiden rakenteen ja metatiedon yhdistelmä, esimerkkeinä musiikkilevy, elokuva ja WWW-sivu. Mpeg-21 ottaa kantaa seuraaviin asioihin: digitaalisten tuotosten määrittely ja yksilöiminen, sisällön käsittely ja käyttö, tekijänoikeuksien hallinta ja suojaus, päätelaitteet ja siirtoverkot, sisällön esittäminen sekä tapahtumien raportointi [Burnett 2003]. MPEG-21 on monimutkainen ja abstrakti standardi ja sitä on otettu käyttöön varsin rajallisesti.

## 4. Staattinen vs. dynaaminen multimediaesityksen koostaminen

Multimediaesitykset kootaan nykyisin pääsääntöisesti staattisesti eli esityksen elementit, sen kulku ja mahdolliset vaihtoehtoiset etenemisjärjestykset valitaan ja määrätään esityksen tuotantovaiheessa. Tätä tukevat jo multimedian tuotantotyökalutkin: ne on tarkoitettu esitysten tuottamiseen epälineaaraisesti eli esitys kootaan muokkausvaiheessa ja julkaistaan lopullisessa formaatissa työstämisvaiheen lopuksi. Myös multimedian tiedostoformaatit tukevat staattisia esityksiä, sillä perusideana suurimmassa osassa niistä on paketoita kaikki esityksen mediaelementit yhteen tiedostoon.

Äänen ja erityisesti videokuvan tuottaminen täysin dynaamisesti eli niiden luominen jokaista esityskertaa varten erikseen on raskasta eikä sitä kannata tehdä kuin erikoissovelluksissa. Tämän takia ääni- ja videoleikkeet kannattaa yleensä pitää staattisina. Sen sijaan teksti- ja kuvaelementtien tuottaminen täysin geneerisesti tietokannassa olevan datan perusteella on täysin arkipäiväistä nykyisissä WWW-sovelluksissa. Jos multimediaesityksen eri mediaelementit pystytään pitämään erillään, jo näiden viimeksi mainittujen elementtien saaminen dynaamiseksi esityksen koostamisen lisäksi mahdollistaa multimediamateriaalin rakenteisuuden hyödyntämisen monella tavalla.

Dynaamisen eli ajonaikaisen koostamisen avulla multimediaesitykset saadaan tukemaan päivittyvää sisältöä. Vaikka esitys sisältäisikin muuttumattomia ääni- ja videoleikkeitä, teksti- ja kuvaosuudet voidaan tuottaa ohjelmallisesti tietokannassa olevan ja päivitettävän sisällön perusteella. Saman tekniikan ansiosta muuten staattiseen esitykseen, esimerkiksi luentotallenteeseen, voidaan tarjota yksinkertainen selainkäyttöliittymä, jonka avulla niin sisällön tuottaja kuin tallenteen katselijatkin voivat lisätä kommentteja tai tarkennuksia haluamiinsa kohtiin esitystä. Nämä lisäykset tulevat välittömästi näkyville muillekin tallenteen katselijoille.

Multimediamateriaalin rakenteistaminen ansiosta samoja mediaelementtejä voidaan hyödyntää monessa paikassa ilman kopiointia. Tämä koskee erityisesti ääni- ja videoleikkeitä,

koska tyypillisesti ne ovat vaikeinten päivitettäviä ja vievät paljon tallennuskapasiteettiä. Samoin HTML-pohjaisista materiaaleista tuttu räätälöitävyys käyttäjän ja tilanteen mukaan tulee mahdolliseksi, esimerkiksi multimediakoulutusmateriaaliin voidaan helposti vaihtaa logot ja tunnistetekstit koulutuksen kohderyhmän mukaan.

Käyttäjädaptiivisuuden lisäksi dynaaminen esitysten tuottaminen mahdollistaa myös muun tyyppistä adaptiivisuutta. Multimedian katseluun soveltuvien päätelaitteiden kirjo kasvaa jatkuvasti älykkäiden kämmenlaitteiden yleistyessä, jolloin sekä siirtokaistan rajallisuus että näytön koko asettavat vaatimuksia multimediasisällöille. Jos saman sisällön halutaan olevan käytettävissä sekä työasema- että mobiilipäätelaitteilla, sisällön tulee muokautua esitystilanteen mukaan vaihtamalla mediaelementtejä katselijan päätelaitteelle optimoituihin versioihin tai jättämällä tiettyjä mediaelementtejä kokonaan pois.

Multimediaesitysten dynaamisuus tuo mukanaan myös tiettyjä haasteita. Kun verkkomultimedian tapauksessa staattisille esityksille riittävät normaali WWW-palvelin ja mediapalvelin, vaativat dynaamiset esitykset palvelimelle oman ohjelmistonsa ja yleensä myös tietokannan. Valmisohjelmistojen valikoima tähän tarkoitukseen on hyvin niukka, joten oma ohjelmistokehitys on käytännössä vaatimuksena toteutettaessa dynaamiseen multimediasisältöön perustuvia verkkopalveluita.

Staattinen multimediaesitys toimii oikein tehtynä sekä verkossa että paikallisesti CD-ROM/DVD-levyltä ilman muokkauksia, mikä helpottaa aineiston siirrettävyyttä ja käyttöä paikoissa, joissa ei ole ollenkaan tai riittävän laadukasta verkkoyhteyttä käytettävissä. Dynaaminen materiaali vaatii aina verkkoyhteyden, mikä on selvä rajoitus aineiston käyttökelpoisuuden suhteen suhteen. Toki dynaamisesta materiaalista voidaan tehdä staattisia versioita, mutta se vaatii oman toimintalogiikan rakentamisen julkaisusovellukseen.

Verkon siirtoprotokollat eivät juuri tue multimedian siirtoa palvelun laadun (Quality of Service, QoS) kannalta, mikä aiheuttaa ongelmia erityisesti silloin, kun dynaamisen multimediaesityksen eri elementit ladataan eri paikoista. Jos siirtokaista on rajallinen ja katsottava multimediaesitys vaatii kaiken tai lähes kaiken käytettävissä olevan siirtokaistan, on todennäköistä, että osa mediaelementeistä ei saavu ajallaan päätelaitteelle. Jos soitto-ohjelman puskurointi ei riitä kompensoimaan eri mediaelementtien viipeitä, seurauksena

on joko katkoksia esitykseen tai ongelmia eri elementtien synkronoinnissa riippuen multimediasisällöstä ja katselusovelluksen virheenkäsittelylogiikasta. Staattiset esitykset on yleensä pakattu yhteen tiedostoon, joka on helpompi siirtää verkon yli ja soitto-ohjelman on helpompi huolehtia mediaelementtien synkronoinnista myös silloin, kun verkkoliikenteen kanssa tulee ongelmia.

## **4.1. Matemaattinen mallinnus -verkostohankkeen luentotallenteet**

Matemaattinen mallinnus -verkostohanke [Mallinnus 2004] on Tampereen teknillisen yliopiston (TTY) koordinoima yhdeksän yliopiston yhteinen projekti, jossa tuotetaan matemaattista mallinnusta käsittelevää materiaali ja kursseja verkkoon. Hankkeessa tuotetut kurssit ovat verkkokursseja, joiden luennot välitetään verkossa videoituina luentotallenteina. Tallenteet ovat luonteeltaan staattisia multimediaesityksiä eli ne koostuvat staattisista tiedostoista.

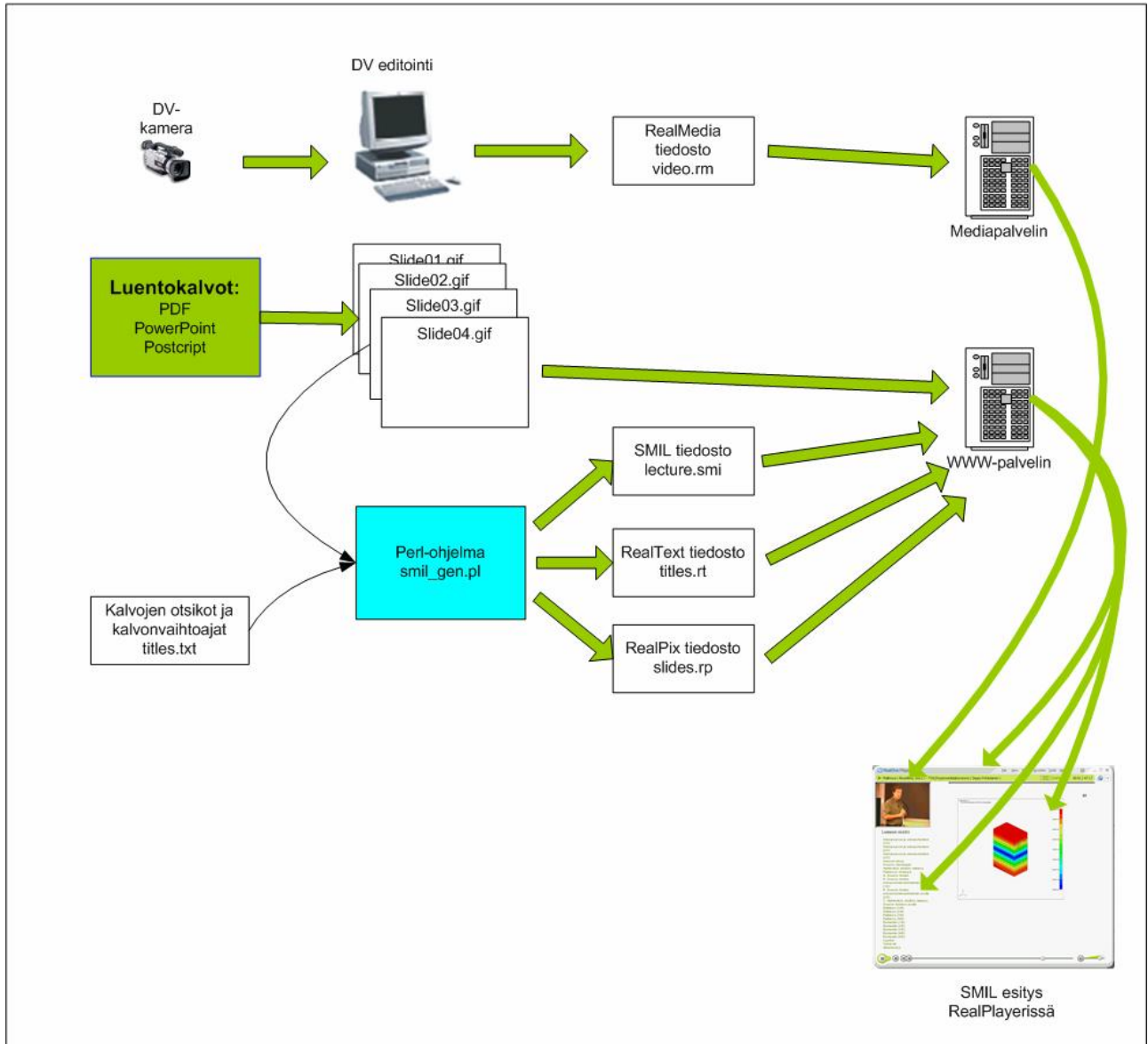
Luentotallenteiden lähtökohtana oli synkronoida luentojen esitysmateriaali eli käytännössä luentokalvot digitaaliseen videotallenteeseen. Ensimmäisessä vaiheessa luentotallenteet toteutettiin Sync-O-Matic -ohjelmalla [Sync-O-Matic 2004], joka tuotti HTML-kehysrakenteen HTML-sivulle upotettua videoleikettä ja kalvokuvia varten. Ohjelma itsessään oli helppokäyttöinen, mutta sen tuottamissa tallenteissa oli ongelmia selainyhteensopivuuksien ja tallenteiden siirrettävyyden kanssa. Kun ohjelman kehitys vielä pysähtyi, edessä oli luentotallenteiden tuottamistekniikan vaihto ensimmäisen vuoden jälkeen.

Seuraavaksi tekniikaksi valittiin SMIL-multimediakuvauskieli (Synchronized Multimedia Integration Language), joka takaa laajan asiakaspään laitetuen sekä paremman tallenteiden hallittavuuden Sync-O-matic -ratkaisuun verrattuna. SMIL-kielen avulla voidaan koostaa multimediaesityksiä useista eri mediatyypeistä määrittelemällä asemointi (layout) ja ajallinen esityksen kulku yhteen XML-pohjaiseen tiedostoon. Itse mediaelementit, joista esitys koostetaan, voivat sijaita missä tahansa verkossa. [Pohjolainen et al. 2003]

SMIL on World Wide Web Consortiumin (W3C) suositus, joka periaatteessa takaa yhteensopivuuden eri soitto-ohjelmien (player) välillä. Käytännössä tilanne on kuitenkin se, että SMIL-esityksen koostamiseen käytetyt mediaelementit, erityisesti videomateriaalin osalta, rajoittavat eri soitto-ohjelmien käyttökelpoisuutta. Matemaattisen mallinnuksen kurseilla soitto-ohjelmaksi on valittu RealPlayer, mikä mahdollistaa tiettyjen Real Networksin omien formaattien käytön tekstin ja kuvien esittämisessä.

### **Luentotallenteiden tuotantoprosessi**

Seuraavassa kuvassa on esitetty kurssilla käytetty tuotantoprosessi. Ensimmäisessä vaiheessa luento ja sillä käytetyt oheismateriaalit tallennetaan digitaalisesti tietokoneelle. Toisessa vaiheessa kootaan lähtötiedostot SMIL-esityksen luontia varten ja kolmannessa vaiheessa generoidaan SMIL-esityksen tarvitsemat tiedostot ja sijoitetaan ne mediapalvelimelle.



Kuva 3 Staattinen luentotalenne SMIL-tekniikalla

Luennot kuvattiin digitaalivideokameralla, johon ääni tuotiin erillisen mikrofonin kautta. DV-kameralla kuvattiin sekä itse luennoijaa että valkokankaalle heijastettuja luentokalvoja, jolloin luennoija pystyi luontevasti osoittamaan yksityiskohtia luentokalvoilta. Luennoijien esitysmateriaalille oli esitetty vaatimus, että niiden tuli olla digitaalisessa muodossa, jolloin luentokalvojen muuttaminen yksittäisiksi kuviksi oli yksinkertaista eli työlästä kalvojen skannausta tarvinnut tehdä.

Digitaalinen videomateriaali siirrettiin toisessa vaiheessa videoeditointitietokoneelle, jolla tehtiin yksinkertaiset videotallenteen muokkaukset. Haluttujen muokkausten jälkeen video- ja ääni pakattiin RealMedia-formaattiin.

Luentokalvot voidaan muuttaa yksittäisiksi GIF-kuviksi joko itse tuotantosovelluksesta tai ruudunkaappauksen kautta. Toisen vaiheen työläin osuus on luentokalvojen vaihtoaikojen määrittäminen. Jos kalvonvaihtoaikoja ei ole merkitty muistiin jo kuvausvaiheessa, luennoijan kalvojen tarkat vaihtoajat täytyy hakea kuvanauhan avulla, mikä kuluttaa aikaa. SMIL-luentotallenteen sisällysluetteloa varten tarvitaan lisäksi yksittäisten luentokalvojen otsikot. Myös tämä on pitkälti käsityötä, ellei otsikkoja saada ohjelmallisesti kaivettua esiin alkuperäisestä esitysmateriaalista.

Kun kalvokuvat, ajastustiedot ja kalvojen otsikot on saatu tallennettua omiin tiedostoihinsa, voidaan koostaa varsinainen SMIL-esitys. SMIL-tiedostojen muokkaaminen onnistuu millä tahansa tekstieditorilla. Koska tällä kurssilla tallenteissa käytettiin RealText- ja RealPix-formaatteja tekstin ja kuvien esittämiseen, kaikkien tarvittavien tiedostojen muokkaaminen käsin oli liian työlästä. Tämän takia oli käytössä pieni Perl-ohjelma (kuvan `smil_gen.pl`), joka tekee halutut loppuformaatit ja -tiedostot annettujen lähtötietojen perusteella. Ohjelman avulla on mm. mahdollista nopeasti muuttaa lopullisen esityksen ulkoasua.

Luentotallenteen videotiedosto, luentokalvojen kuvat ja Perl-ohjelmalla tuotetut uudet tiedostot siirrettiin mediapalvelimelle, josta ne ovat katseltavissa. Videotallenteissa käytettiin RealMedian ns. SureStream-ominaisuutta, mikä mahdollistaa usean eri yhteysnopeusversion tallentamisen samaan tiedostoon. Mediapalvelimen tehtäväksi jää valita oikea yhteysnopeusversio, joka varsinaisesti näytetään katselijalle. Tällä tavoin luentotallenteissa tuettiin sekä modeemikäyttäjää että laajakaistaliittymien omistajia.

Luentotallenteiden tallentamiseen ja jakamiseen verkon välityksellä mediapalvelin on välttämättömyys, koska WWW-palvelimen käytöllä menetettäisiin käytännössä mahdollisuus seurata luentotallennetta ei-lineaarisesti. Tavalliselle WWW-palvelimelle tallennetun videoleikkeen katselua ei voida aloittaa keskeltä leikettä ilman, että soitto-ohjelma joutuisi lataamaan koko alkupään videotallenteen. Tämä näkyisi katselijalle jopa minuuttien mittaisena odotusaikana tilanteessa, jossa hän haluaisi aloittaa luennon seuraamisen keskeltä luentoa. Mediapalvelin pystyy aloittamaan videoleikkeen suoratoiston eli videokuvan lähettämisen mistä tahansa kohtaa videotallenteesta eikä ylimääräisiä odotusaikoja synny.



Matemaattisen mallinnusten luentotallenteiden jakamiseen käytettiin CSC:n mediapalvelinta, koska Hypermedialaboratoriolla on käytössään ainoastaan testikäyttöön ja vähäiseen tuotantokäyttöön soveltuva mediapalvelin. Valintaan vaikuttivat myös kustannuskysymykset, sillä kymmeniä samanaikaisia katselijoita tukevan RealServer-palvelinohjelmiston vuosilisenssi on useita tuhansia euroja.

Tarkemmat tiedot mediapalvelimesta ja käytetyistä ohjelmaversioista ovat liitteessä 3.

Luentotallenteiden linkitys osaksi verkkokurssien WWW-sivuja tapahtui käsin sekä kurssin staattisille julkisille sivuille että kurssin verkkotoimintaympäristönä toimineeseen A&O-oppimisympäristöön. Muunlaista julkaisujärjestelmää ei missään vaiheessa harkittukaan, koska luentotallenteita on tuotettu verkostohankkeessa yhteensä muutama kymmenen ja yhtä kurssia kohden luentotallenteita on noin kymmenen kappaletta. Tällainen määrä pystytään hallitsemaan vielä ilman tietokantaratkaisuja ja automatisoitua julkaisujärjestelmää.

## **Pohdintaa toteutuksesta**

Edellä esitetty luentotallenteiden tuotantoprosessi on suhteellisen yksinkertainen, vaikkakin se sisältää useita manuaalisia työvaiheita. Tuotantoprosessin automatisointiin ei ollut resursseja ja siitä saavutetut hyödyt olisivat jääneet ainakin Hypermedialaboratorion mitakaavassa vähäisiksi, sillä luentotallenteiden lukumäärä jää vuositasolla alle kymmenen. Yksittäisten luentojen ja seminaarien tapauksissa tallenteiden tekemiseen videokuvaus mukaan lukien kulunut kaksinkertainen aika luennon kestoon verrattuna on vielä hyväksyttävissä, mutta yhden tai kahden viikoittain luennoitavan kurssin luentojen tallentamisessa tämä ei ole mahdollista.

Luentotallenteiden tuottamisessa suurin vaiva piti nähdä kalvojen ajastamisessa ja synkronoinnissa videoon. Kalvojen ajastamista voitaisiin huomattavasti yksinkertaistaa tallentamalla luennon aikana kalvojen vaihtumisajankohdat tietokoneella, jolta esitys pidetään. Tällöin vältettäisiin virheeltis aikakoodien käsin kirjaaminen luennon kuvauksen aikana ja työläs kalvonvaihtokohtien hakeminen ja tarkistaminen videonauhoitteesta. Tässä suhteessa siirtyminen SMIL-muotoisiin luentotallenteisiin oli askel taaksepäin, sillä Sync-O-

Matic-ohjelmalla kalvojen kestoajat saatiin talteen automaattisesti. Toisaalta SMIL-tekniikalla tuotetut tallenteet ovat siirrettävissä sellaisenaan joko palvelimelta toiselle tai optiselle medialle ilman mitään muokkaustoimenpiteitä.

Luentotallenteiden videoleikkeet on optimoitu laajakaistayhteyksille eli ADSL-yhteyksille tai vastaaville yli 300kbit/s nopeuksisille internet-yhteyksille. Vaikka videotiedostoissa on myös hitaiden yhteysnopeuksien versiot, ne toimivat huonosti alle 100kbit/s nopeuksilla, koska kuvakokoa ei ole pienennetty hitaita yhteyksiä varten. Suuri kuvakoko ja alhainen bittivirtanopeus näkyvät kuvan hyvin selvänä nykimisenä ja hitaana päivittymisenä kuvassa tapahtuvien isojen muutosten jälkeen. Tämä puute voitaisiin korjata suhteellisen yksinkertaisesti tuottamalla videoleikkeestä useita eri kuvakokoversioita ja käyttämällä SMIL-kielen switch-rakennetta valitsemaan oikea versio katselijan soitto-ohjelman yhteysnopeusasetusten perusteella (ks. kappale 5.1). Luentotallenteista ei ole myöskään tarjolla pelkkää audio-versiota, joka voisi kiinnostaa modeemikäyttäjiä ja kannettavien musiikki-soittimien omistajia erityisesti silloin, jos audiotallenteen voisi tallentaa omalle tietokoneelle.

Tiedonsiirtokapasiteetin riittämättömyydestä ei ole syntynyt ongelmia korkeakouluverkoissa, koska kaistaa on käytettävissä riittävästi niin CSC:n mediapalvelimen päässä kuin yliopistojen omissakin verkoissa. Sen sijaan ongelmia tallenteiden seuraamisessa aiheuttivat asiakaspään rajallinen siirtokaista kuluttaja-internet-liittymissä. Vaikka esimerkiksi kaapelimodeemiliittymän nimellisnopeus on 512kbit/s, aina tuo nopeus ei ole käytettävissä jaetun verkkomedian luonteen takia. Sama tilanne on kiinteistö- tai taloyhtiöliittymissä, joissa samaa laajakaistaliittymää käyttää useampi henkilö samanaikaisesti. Tällaisissa ruuhkatilanteissa katselijan pitää tunnistaa tilanne ja osata vaihtaa nopeusasetukset RealP-layeristään vastaamaan hitaampaa yhteyttä, jolloin soitto-ohjelma pyytää mediapalvelimelta myös heikompileatuisen mutta varmemmin toimivan version videoleikkeestä. Mediapalvelin pystyy tekemään vaihdoksen automaattisestikin, mutta se aiheuttaa epämiellyttäviä katkoksia ja ylimääräisiä datan puskurointiaikoja suoratoistossa.

Hitailta yhteysnopeuksilla SMIL-esityksissä pitää huomioida kuva- ja tekstielementtien vaatima siirtokaista, luentotallenteiden tapauksessa kuvien ja videon siirtokaistan tarpeet ovat samaa suuruusluokkaa alle 70 kbit/s:n yhteysnopeuksissa. Real Networks tarjoaa yk-

sinkertaisen laskurin arvioimaan kuvien tarvitsemaa kaistaa ja katselijan kokemaa datan puskuroinnista aiheutuvaa viivettä [Real 2004c].

Luentotallenteet edustavat staattista multimediaa esitysten koostamisen näkökulmasta. Staattisen tallennusmuodon ansiosta tallenteet ovat helposti siirrettävissä jakelu- tai tallennuspaikasta toiseen ja ne vaativat minimaalisen vähän ylläpitoa, kunhan mediapalvelin on toimintakunnossa. Toisaalta kaikki muokkaustoimenpiteet täytyy tehdä ensin muualla ja korvata muokatut tiedostot tämän jälkeen palvelimella tiedostojärjestelmätasolla. Luentotallenteita ei voida myöskään rikastaa lisäelementtien avulla käytön aikana eli esimerkiksi luennoija ei pysty itse lisäämään oheistietoa tai päivittyvää tietoa olemassa oleviin luentotallenteisiin.

Luentotallenteet tuotettiin hyvin suoraviivaisesti pyrkimyksenä minimoida tallenteen tekoon kuluva aika. Toimintamalli luentojen tallentamisesta ja julkaisusta tehtiin etukäteen käyttäen hyväksi Cernissä Web University -pilotissa [Web University 2004] [Rinta-Filppula et al. 2002] saatuja kokemuksia. Yksittäisiä luentotallenteita ei suunniteltu erikseen, luennoijia ainoastaan ohjeistettiin ennen luentoaan käyttämään digitaalisia oheismateriaaleja. Luennoijat saivat pitää luentonsa omalla tyylillään videokuvauksien tietenkin huomioon ottaen eli mitään sisältö- tai kuvakäsikirjoitusta ei tehty etukäteen yhdessä luennoijan kanssa.

Metatietoa hyödynnettiin luennoissa minimaalisesti liittyen ainoastaan tallennetiedostojen luomiseen Perl-ohjelmalla, joka tarvitsi syötteekseen sekä teknisiä tietoja SMIL-esityksen kuvakokoon ja ajalliseen keston liittyen että sisällöllistä tietoa puhujan nimen ja luennon otsikon muodossa. Lisäksi ohjelma tarvitsi syötteekseen luentokalvojen otsikot ja niiden kestot. Luentokalvoihin liittyviä tietoja olisi mahdollista käyttää rakenteistamaan luentoja, eli siihen, että yksittäiset luennot voitaisiin jakaa pienempiin osiin ja näihin osiin pystyttäisiin viittaamaan suoraan tallenteen ulkopuolelta. Tämä vaatisi kuitenkin olemassa olevan tiedon jatkojalostamista, metatietovaraston rakentamisen ja tarkemman aihejaottelun määrittelyä.

Teknisesti luentotallenteet toimivat hyvin, suurimmat ongelmat liittyivät soitto-ohjelmaan eli RealPlayeriin. RealPlayer ei kuulunut kaikkien hankkeeseen osallistuneiden yliopisto-

jen vakio-ohjelmistotarjontaan, joten soitto-ohjelmien saatavuus yliopistojen tietokoneilla oli joissakin tapauksissa ongelmallista. RealPlayer kävi läpi hankkeen aikana myös monta versiosukupolvea, mikä ei näkynyt Windows-alustoilla, mutta aiheutti jonkin verran yhteensopivuusongelmia Linux- ja erityisesti Mac-alustoilla. Koska luentotallenteita tuotettiin usean vuoden aikana, tuotanto-ohjelmistotkin vaihtuivat hankkeen aikana. Uudet tuotanto-ohjelmistot ja erityisesti RealMedian kehittyneet videon pakkausalgoritmit olisivat mahdollistaneet parempilaatuisten videoleikkeiden tuottamisen hitaammille yhteysnopeuksille, mutta näihin ei siirrytty nimenomaan yhteensopivusseikkojen takia.

## 5. Dynaamisen multimedian toteutustekniikat

Verkkomultimedian ja siten myös multimediaoppimateriaalin tuottamiseen on ollut olemassa lukuisia eri menetelmiä ja formaatteja jo vuosia. Tyypillistä ensimmäisille toteutuksille olivat valmistajakohtaiset ja suojatut tiedostoformaatit, multimediaesitysten jakaminen tiedostoina binäärisessä muodossa ja katselijakohtaisen räätälöitävyyden puuttuminen, koska multimediaesitykset koostettiin lopulliseen ulkoasuunsa tuotantovaiheessa. Tätä aikakautta edustavat Toolbook ja Macromedia Director, joita käytettiin 1990-luvulla hyvin yleisesti sekä multimedia-CD-ROMien että verkkomultimediaoppimateriaalin tuottamiseen.

Multimediaesitysten tuottaminen dynaamisesti käyttäjän tarpeiden ja toiveiden mukaisesti on yleistynyt vasta viime vuosina. Tekniikoita tämän dynaamisuuden tuottamiseen on olemassa useita. Puhtaasti video- ja äänipohjaisessa sisällössä katselijakohtaisten esitysten luominen onnistuu helposti soittolistojen avulla, sillä käytännössä kaikki mediasoittimet tukevat useiden video- ja äänileikkeiden soittamista peräkkäin soittolistassa määritellyssä järjestyksessä. Joissakin soittolistaformaateissa, kuten Realin ja Quicktimen tapauksissa, tuetaan myös soitettavien leikkeiden alku- ja loppukohtien määrittämistä aikakoodilla, jolloin leikkeestä voidaan soittaa vain haluttu osa riippumatta leikkeen oikeasta pituudesta. Tällaisten soittolistojen avulla on toteutettavissa esimerkiksi verkko-jukeboksi, jossa käyttäjä voi käydä vaihtoehdot nopeasti läpi soittamalla yhdellä valinnalla kaikista leikkeistä ensimmäiset 20 sekuntia ja valita tämän jälkeen kiinnostavat ääni- ja/tai videoleikkeet omaan henkilökohtaiseen soittolistaansa haluamassaan järjestyksessä.

Toinen soittolista-idea hyödyntävä sovellus on räätälöitävä videokatselu, jonka ideana on joko jättää näyttämättä elokuvasta tai muusta pitkästä videosesityksestä sopimattomat tai epäasialliset kohdat tai toisaalta näyttää ainoastaan muutama esimerkiksi opetuksellisesti kiinnostava kohta elokuvasta. Metatiedot näytettävistä kohdista talletetaan erillään varsinaisesta videotallenteesta, jolloin työlästä videoeditointia ei tarvita ja soittolista voidaan jakaa erikseen muiden videotallenteen käyttäjien kanssa. Tällainen toiminnallisuus on jo

toteutettu joihinkin DVD-soittimiin lähinnä helpottamaan vanhempien mahdollisuutta kontrolloida lastensa elokuvien katselua, mutta samaa ideaa voidaan hyödyntää myös opetuskäytössä sekä DVD-elokuvien että verkkovideoiden katselussa. [Bush 2004]

Kun dynaamiseen multimediaesitykseen tai -oppimateriaaliin halutaan mukaan muita mediaelementtejä ja eri mediaelementtien halutaan näkyvän ja kuuluvan saman aikaisesti, soittolistamalli ei enää riitä. Yleisimmät tavat yhdistellä eri mediaelementtejä verkkosisäloissä ovat HTML:n käyttö liimakielenä ja Macromedia Flash -formaatin hyödyntäminen. Molemmissa vaihtoehdoissa on omat rajoituksensa, joten tämän työn käytännön toteutuksessa tekniikaksi valittiin W3C:n määrittelemä SMIL-multimediakuvauskieli.

Toinen standardoitu multimediaformaatti MPEG-4 on tulossa laajempaan käyttöön erityisesti mobiilipuolella ja sen objektipohjainen rakenne soveltuu hyvin ajatusmalliin multimediaesitysten koostamisesta eri elementeistä. Sen tuki sekä soitto-ohjelmissa että tuotanto-ohjelmistoissa on kuitenkin vielä vajaata ja binäärisenä formaattina sen reaaliaikainen tuottaminen on raskasta.

Seuraavissa kappaleissa käydään läpi eri tekniikkojen vahvuuksia ja heikkouksia.

## **5.1. SMIL-kuvauskieli**

W3C-konsortion määrittelemä XML-pohjainen SMIL-kuvauskieli (Synchronized Multimedia Integration Language) on tarkoitettu nimenomaan luentotallenteiden kaltaisten multimediaesitysten koostamiseen ja synkronoimiseen erilaisista mediaelementeistä sekä niiden jakamiseen verkon välityksellä. SMIL ei ole suoranainen kilpailija muille vastaaville verkkomultimedia-arkkitehtuureille kuten esimerkiksi Macromedia Flashille, sillä SMIL on pelkkä kuvauskieli, se tarvitsee muita formaatteja itse mediaelementtien välittämiseen. Koska SMIL ei rajaa käytettäviä formaatteja, sen avulla voidaan hyödyntää eri multimediaformaattien vahvuudet yhdessä ja samassa esityksessä: vektorianimaatio voidaan toteuttaa Flashin avulla, videokuva RealMediana, kuvat PNG:nä ja tekstiosuudet puhtaana tekstinä.

Aloituskynnys SMIL-esitysten tuottamiseen on hyvin matala, sillä esityksessä käytettävien mediaelementtien lisäksi tarvitaan vain ASCII-tekstieditori. Toki on olemassa myös SMIL:iä varten kehitettyjä graafisia työkaluja, mutta ne eivät ole välttämättömiä etenkin rakenteeltaan yksinkertaisten SMIL-esitysten tuottamisessa. Valmiiden mallipohjien avulla uusien esitysten luominen onnistuu helposti käsinkin, jos HTML-merkkaus on tuttua.

SMILin suurimmat vahvuudet löytyvät sen integroitavuudesta WWW-sovelluksiin. XML-pohjaisena formaattina SMILin tuottaminen ohjelmallisesti on yksinkertaista millä tahansa WWW-sovellusalustalla. Tämä mahdollistaa dynaamisten multimediaesitysten tuottamisen missä tahansa WWW-sovelluksessa, oli se sitten toteutettu Java-, PHP- tai ASP-tekniikalla.

Ensimmäinen vuonna 1998 valmistunut SMIL-kielen suositus oli suhteellisen yksinkertainen ja sisälsi niukasti ominaisuuksia. Vuonna 2001 julkaistu SMIL 2.0 -suositus on huomattavasti laajempi ja pitää sisällään mm. lisää interaktiivisuusominaisuuksia, laajemmat ajustusmahdollisuudet ja paremman ulkoasun hallinnan. Lisäksi uusi suositus on määritelty ja jaoteltu loogisiin moduuleihin, joista on määritelty profiileita suorituskyvyltään ja näytön ominaisuuksiltaan eroaville päätelaitteille. SMIL Basic on suunnattu kämmentietokoneille ja TV-päätelaitteille ja 3GPP-profiili on tarkoitettu kolmannen sukupolven matkapuhelimiin [W3C 2004].

W3C:n tavoitteena kielen rakenteen suunnittelussa on ollut multimediaesityksen interaktiivisuuden ja ulkoasun kuvaaminen WWW-julkaisuun sopivalla tavalla, koska selaimet eivät aikaisemmin tukeneet integroitujen mediaelementtien vuorovaikutusta ja synkronointia. Esimerkiksi kahden eri liikkuvan kuvan esittäminen samaan aikaan ei ollut hallittua. Toinen tavoite on SMIL-rakenteiden käyttö muissa XML-sovelluksissa, joiden on tarkoitus esittää synkronoitua multimediaa. SMIL-rakenteita hyödynnetään tällä hetkellä W3C:n SVG-vektorigrafiikkaformaattissa (Scalable Vector Graphics) [SVG 2004] sekä XHTML:ssä XHTML+SMIL -profiilin kautta [XHTML+SMIL 2004].

## **Rakenne**

SMIL-esitys voi pitää sisällään videota, ääntä, kuvaa, tekstiä ja animaatioita sekä kaikkien näiden yhdistelmiä. SMIL-kuvauskieli muistuttaa ulkoasultaan HTML-kieltä, suurimmat

erot HTML:ään verrattuna ovat tarkassa layout-määrittelyssä ja aikaulottuvuuden mukanaolossa: SMIL-esityksessä määritellään tarkasti, mitkä mediaelementit näkyvät mihinkin aikaan ja missä järjestyksessä.

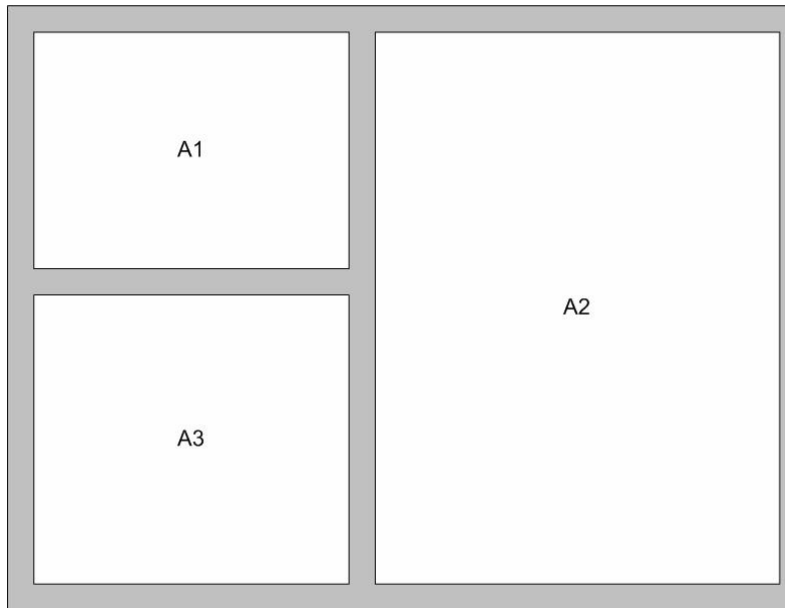
SMIL on kuvauskieli, sen avulla ainoastaan koostetaan multimediaesitys itsenäisistä mediaelementeistä, joihin viitataan SMIL-dokumentista URL-osoitteilla. Mediaelementit voivat siten fyysisesti sijaita missä tahansa verkossa. Tällä saavutetaan seuraavia etuja:

- Erityyppiset medialeikkeet voidaan sijoittaa tarkoituksenmukaisimmalle palvelimelle eli video- ja äänileikkeet mediapalvelimelle ja kuvat ja tekstitiedostot normaalille WWW-palvelimelle.
- Osa SMIL-esityksestä voi olla täysin dynaamista eli jokin mediaobjekti, yleensä kuva- tai tekstielementti, haetaan tietokantapalvelimelta.
- Itse SMIL-dokumentti on XML-dokumentti, joka voidaan helposti luoda dynaamisesti tietokannassa olevien mediaelementtitietojen ja esimerkiksi käyttäjätietojen avulla.
- SMIL-esityksen tuotantovaiheessa ei tarvitse huolehtia eri mediavirtojen yhdistämisestä ja synkronoisesta, vaan tämä jää soitinohjelman tehtäväksi.

SMIL-dokumentissa määritellään esityksen layout eli alueet, joille ulkoiset mediaelementit sijoitetaan, sekä eri mediaelementtien esittämisajankohta. Kuva 4 on esimerkki SMIL-esityksen asemoinnista. Esitykseen on määritelty kolme eri aluetta (A1-A3), joille voidaan sijoittaa visuaalisia mediaobjekteja. Seuraavassa SMIL-merkkaus, jolla kyseiset alueet määritellään [Häkkinen 2004a]:

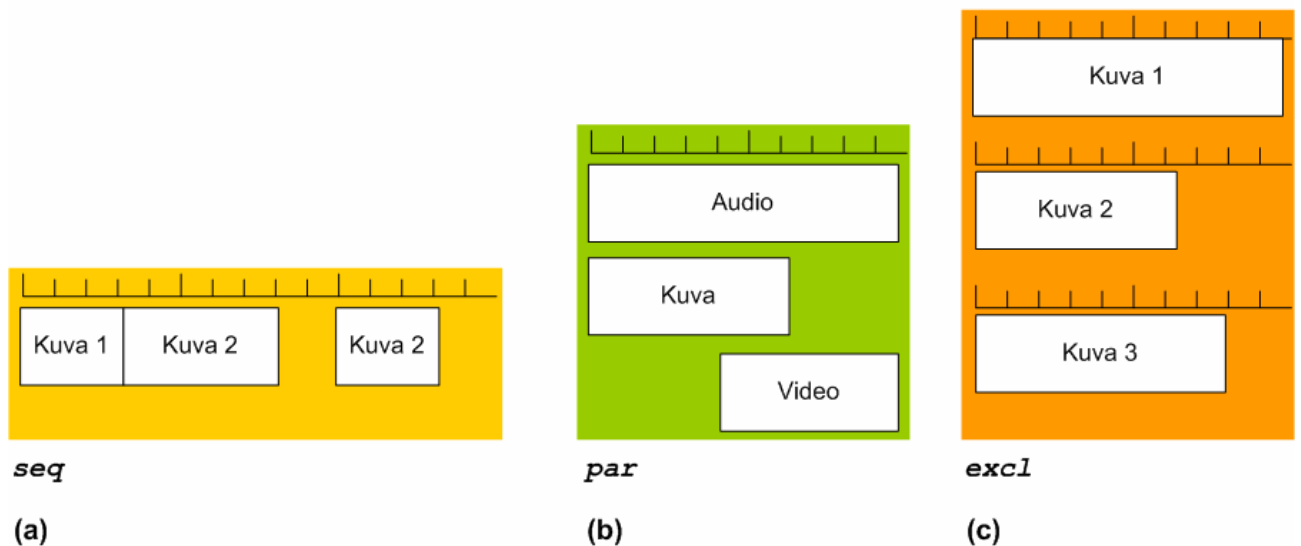
```
<?xml version="1.0" encoding="ISO-8859-1"?>
<smil xmlns="http://www.w3.org/2001/SMIL20/Language">
  <head>
    <layout>
      <root-layout width="572" height="528"/>
      <region id="A1" left="4" top="4" width="280" height="240" />
      <region id="A3" left="0" top="248" width="280" height="280" />
      <region id="A2" left="288" top="4" width="380" height="350" />
    </layout>
  </head>
  <body>
    ...
  </body>
</smil>
```





Kuva 4. SMIL-esityksen layout, johon on määritelty kolme esitysaluetta.

Ajallisesti mediaelementit voidaan sijoittaa toisiinsa nähden kolmella eri tavalla: peräkkäin, rinnakkain tai toisensa poissulkevasti (jälkimmäisin uutena ominaisuutena SMIL 2.0:ssa) kuvan 5 esittämällä tavalla. Aikaelementit ovat nimeltään **seq**, **par** ja **excl**.



Kuva 5. SMIL 2.0:n aikaelementit

Esimerkkien a ja b aikaesitykset saadaan seuraavilla SMIL-merkkauksilla:

**(a) Leikkeet peräkkäin:**

```
<body>
  <seq>
    
    
    
  </seq>
</body>
```

**(b) Leikkeet rinnakkain:**

```
<body>
  <par>
    <audio src="audio.mp3"
    
  </par>
</body>
```

SMIL-dokumentissa seq-elementin sisään sijoitetut mediaobjektit soitetaan peräkkäin eli kun edellinen mediaobjektin esittäminen päättyy, seuraavan alkaa. Kullekin mediaobjektille voidaan lisäksi määritellä kesto sekä alkamisaika ja päättymisaika, jotka lasketaan suhteellisina aikoina edellisen objektin päättymisestä. Mediaelementeille, jotka eivät sisällä aikatieta jo itsessään, on määritely oletuskestoksi 0 sekuntia, joten kaikille teksti- ja kuvaobjekteille täytyy määritellä kesto eksplisiittisesti.

Par-elementin sisään sijoitetut medialeikkeet esitetään samanaikaisesti. Esimerkiksi kuvan 4 mukaisessa esityksessä taustalla voi kuulua ääniraita, alueessa A2 voidaan näyttää kuvaa ja alueessa A1 käynnistyy videoleike pienellä viiveellä kuvan ilmestymisen jälkeen.

Excl-elementti on uusi aikaelementti SMIL 2.0:ssa ja se on tarkoitettu käytettäväksi yhdessä katselijan aktivoimien tapahtumien, esimerkiksi hiiren avulla tehtävien valintojen kanssa. Esimerkiksi käyttäjälle näytetään kolme painikkeen kuvaa, joita napsauttamalla katselija voi valita haluamansa kuvan toiseen kohtaa SMIL-esitystä [Learning SMIL 2004].

Edellä mainitut kolme aikaelementtiä voidaan sijoittaa toistensa sisään ja rakentaa hyvinkin monimutkaisia esitysrakenteita. Yksi SMIL:in vahvuus on nimenomaan aikatason adaptiivisuus: koko esityksen tai edes sen osan kesto ei tarvitse kiinnittää SMIL-dokumenttiin, vaan esityksen kesto mukautuu mediaobjektien keston mukaan ellei toisin haluta. Tämä mahdollistaa vaihtuvapituisten mediaobjektien käyttämisen SMIL-

esityksissä, jolloin tietyn elementin korvaaminen päivitettyllä, kestoiltaan erilaisella versiol-la alkuversiossa, ei riko esityksen rakennetta.

SMIL tukee suoraan esityksen mukautuvuutta katselijan mukaan switch-rakenteella. Eh-toina, joiden mukaan valitaan katselijalle esitettävät medialeikkeet, voidaan käyttää soitto-ohjelman kieliasetuksia ja katselijan valitsemaa Internet-yhteyden nopeutta, päätelaitteen suorituskykyä sekä katselijan määrittelemiä saavutettavuusasetuksia. Näiden tietojen avul-la voidaan toteuttaa tietyn asteista sisällön dynaamisuutta staattisilla SMIL-tiedostoilla.

W3C:n SMIL-suositus ei ota kantaa siihen, missä tiedostoformaattissa kukin mediaele-menti on. Sopivat formaatit määrää SMIL-soitto-ohjelma, jolla esityksiä on tarkoitus kat-sella. RealPlayer-soitto-ohjelman yleisyyden takia useimpien SMIL-esityksien koostami-seen käytetään käytännössä Real Networksin formaatteja (mm. Realmedia, RealPix, Real-Text), joita RealPlayer osaa näyttää. Quicktimen ja Internet Explorer 6:n SMIL-tuki tule-vat varmaan tulevaisuudessa rikastamaan formaattitarjontaa. Tällä hetkellä puhtaiden SMIL-esitysten katselemiseen käytetään lähes yksinomaan RealPlayer-ohjelmistoa. WWW-selainta ei siis tarvita, kaikki luentotallenteen elementit esitetään soitto-ohjelman sisällä.

### **SMIL-esitysten dynaaminen tuottaminen**

SMIL-kuvauskielen yksi tärkeimmistä lähtökohdista on ollut sillä tuotettujen esitysten joustavuus ja helppo muokattavuus sekä manuaalisesti että ohjelmallisesti. SMIL tukeekin erittäin hyvin monentasoista dynaamisen multimediasisällön tuottamista.

Yksinkertaisin vaihtuvasisältöinen SMIL-esitys voidaan toteuttaa luomalla staattinen SMIL-tiedosto, joka määrittelee esityksen ulkoasun ja mediaelementtien asemoinnin, ja vaihtamalla tarpeen mukaan SMIL-tiedostossa viitattuja medialeikkeitä uusiin päivitettyihin versioihin. Tämä toimii esimerkiksi tiedottamistyypissä toiminnassa, jolloin uutis-videotiedoston vaihtaminen uuteen onnistuu ilman muutoksia muihin esityksessä käytet-täviin tiedostoihin. Kun esitys on rakennettu sopivasti, videoleikkeen keston muuttuminen ei aiheuta ongelmia koko esityksen toimivuudelle.

Seuraava askel on tuoda SMIL-esitykseen ripaus aitoa dynaamisuutta liittämällä esitykseen komponentteja, jotka luodaan dynaamisesti jokaisella katselukerralla uudelleen. Tällä tavoin muuten staattiseen ja muuttumattomaan luentotallenteeseen tai muuhun valmiiseen multimediaoppimateriaaliin voidaan lisätä päivittyviä teksti- ja kuvaosuuksia, jotka generoidaan tietokannassa olevan datan tai muualta verkossa saatavilla olevan datan avulla. Esimerkiksi paikallissäätä käsittelemään tallenteeseen voitaisiin lisätä reaaliaikaisia säätietoja katselijan oman paikkakunnan sääoloista.

Viimeinen askel on tuottaa suurin osa tai kaikki SMIL-esityksessä käytettävät tiedostot katselijakohtaisesti itse SMIL-tiedostosta lähtien tietokannan tai muun datalähteen pohjalta. Tällöin katselija itse tai hänelle määritelty profiili määrittelee paitsi multimediaesityksen sisällön myös esitystavan. Tätä käsitellään tarkemmin luvussa 6 käytännön esimerkin kautta..

## **5.2. Muut verkkomultimedian koostamisformaatit**

Macromedia Flash [Macromedia 2004] on erittäin suosittu verkkomultimediaformaatti animaatioiden ja muun vektorigrafiikan tuottamiseen. Flashin avulla on mahdollista luoda erittäin näyttäviä ja kuitenkin suhteellisen hitailla yhteysnopeuksilla toimivia multimedia tuotteita, joihin voidaan upottaa runsaasti älykkyyttä asiakaspäähän Flashin ActionScript-ohjelmointikielen avulla. Flash-animaatioihin voidaan tuoda teksti- kuva-, ääni- ja videoleikkeitä dynaamisesti esimerkiksi tietokannasta, jolloin dynaamisen sisällön esittäminen on mahdollista suoraan Flash-animaationa toteutetussa käyttöliittymässä.

Flash-animaatioiden dynaamiseen tuottamiseen Macromedialla on oma palvelinohjelmistonsa Macromedia Flash Remoting. Sen avulla Flash-animaatioita voidaan tuottaa aivan vastaavalla tavalla kuin dynaamisia WWW-sivustoja tietokantapohjaisesti.

Flashin tuki eri mediaformaateille on varsin rajallinen, esimerkiksi videoleikkeet pitää pakata omaan formaattiinsa, jotta sen voi sisällyttää Flash-animaatioon. Macromedia tukee yleisiä verkkovideoformaatteja Macromedia Director -tuotteellaan, jonka uusimman versio pystyy esittämään niin RealMedia, Quicktime ja WindowsMedia-formaateissa olevia videoleikkeitä.

Sekä Flashin että Directorin Shockwave-formaatin vahvuutena voidaan pitää niiden soittamiseen tarvittavien selainlaajennusten (plug-in) yleisyyttä eli sisällöntuottaja voi luottaa siihen, että formaattivalinta ei karkota käyttäjiä. Kyseisten formaattien heikkoutena voidaan taas pitää sitä, että ne sitovat käyttämään yhden valmistajan tuotanto-ohjelmistoja. Nämä ohjelmistot ovat lisäksi tarkoitettu ennen kaikkea sisällöntuotannon ammattilaisille, jolloin aloittelijalla on selkeä kynnys lähteä tuottamaan edes staattista omaa sisältöä.

HTML:n ja HTML-sivulle upotettujen mediaelementtien avulla on suhteellisen yksinkertaista yhdistää eri medialeikkeitä samaan esitykseen niin staattisesti kuin myös dynaamisesti. Tietokantapohjaiset WWW-julkaisujärjestelmät ovat yleisessä käytössä ja jos tällaisen julkaisujärjestelmän tietosisältöön ja sen tuottamaan HTML-merkkaukseen pystytään vaikuttamaan, yksittäisten video- ja animaatioleikkeiden sijoittaminen WWW-sivuille on suhteellisen yksinkertaista. Tämä ratkaisu toimii niin kauan, kun erillisiä medialeikkeitä on vähän ja ne eivät ole sidoksissa toisistaan. Jos eri mediaelementtien, esimerkiksi tekstin ja video-osuuksien välille halutaan luoda riippuvuuksia, joudutaan käyttämään JavaScriptiä tai vastaavaa asiakaspään skriptauskieltä sekä dynaamista HTML:ää (DHTML). Näiden merkkausten tuottaminen on jo vaikeampaa ja vaatii käytännössä oman tuotanto-ohjelmistonsa.

Microsoft käyttää omissa tuotteissaan HTML:ää myös verkkomultimediaesitysten koostamisvälineenä. Edellä mainittujen tekniikoiden lisäksi Microsoftin WWW-selain Internet Explorer käyttää HTML-standardin laajennusta HTML+TIME, jonka avulla medialeikkeitä on voidaan asettaa aikajanelle sekä peräkkäin että rinnakkain. Tällaisten esitysten, esimerkiksi Microsoft Producerin [Microsoft 2004b] tuottamat luentotallenteet, ovat erittäin hankalia muokata muilla kuin alkuperäisillä tuotanto-ohjelmilla.

### **5.3. Soitto-ohjelmat**

RealPlayer/RealOnePlayer [Real 2004] on ainoa ei-käyttöjärjestelmään sidottu mediasoitto-ohjelma, joka on levinnyt laajalle käyttäjäkunnalle ja on pystynyt kilpailemaan Windows- ja Mac OS -käyttöjärjestelmien sisältämien soitto-ohjelmien kanssa. RealPlayer oli alun alkaen tarkoitettu puhtaasti verkkomedian katseluun ja se levisi RealAudio- ja Real-

Video-formaattien suosion myötä, mutta nyt sen käyttökohteet ovat monipuolistuneet. Real Networks on pyrkinyt tekemään RealPlayeristään universaalien soitto-ohjelman, joka soveltuu lähes minkä tahansa tyyppisen multimedian materiaalin katseluun WWW-sivuista ja DVD-tallenteista kilpaileviin suoratoistoformaatteihin.

RealPlayer tukee täysin SMIL-kuvauskieltä ja suurin osa SMIL-sisällöntuotannosta on kohdistettu tälle soitto-ohjelmalle. RealNetworks-yritys on ollut aktiivisesti mukana SMIL-suosituksen kehityksessä alusta saakka ja sen tuotteet mukaan lukien sen omat mediaformaattit ovat viritetty hyödyntämään SMIL:iä. RealNetworksin teksti- ja kuvaesitysformaatteja RealText ja RealPix on käytetty paljon SMIL-esitysten elementteinä paikkaamaan erityisesti SMIL 1.0:n köyhää teksti- ja kuvaefektivalikoimaa. Kun lisäksi ääni- ja videosisällön formaattina käytetään usein RealMedia-formaattia (entiset RealVideo ja RealAudio), SMIL-sisältöön on yhdistetty usein vaatimus RealPlayerin käytöstä. Ilmainen versio RealPlayeristä on helposti saatavilla eri käyttöjärjestelmäympäristöihin, mutta se on saanut myös kritiikkiä mainosten pakkosyötöstä ja spyware-ominaisuuksista [Wired News 1999].

Quicktime Player [Quicktime 2004] on Applen kehittämä multimediosoitto-ohjelma, joka on ollut käytössä 90-luvun alkupuolelta lähtien. Sen suosio ja levinneisyys myös Windows-maailmassa on peräisin multimediatorppujen ajalta, jolloin suuri osa videosisällöstä oli Quicktime-formaatissa. Tällä hetkellä se on palaamassa käyttäjämäärissä markkinoita aikaisemmin hallinneiden RealPlayerin ja Windows Media Playerin joukkoon.

Quicktime Playerin vahvuus on sen laaja tuki eri tiedostoformaateille, myös sellaisille, joita muut soitto-ohjelmat eivät ole koskaan hyväksyneetkään. Se tukee yli 200 eri formaattia, mikä tekee sen varsin huokuttelevaksi vaihtoehdoksi tilanteissa, joissa vanhaa sisältöä on runsaasti käytettävissä. Vaikka sen tuki SMIL-kielelle rajoittuukin vanhaan 1.0-versioon, laaja formaattituki kompensoi tätä puutetta ja mahdollistaa SMIL-esityksissä sellaiset sisällöt, joita muilla SMIL:iä tukevilla soitto-ohjelmilla ei voida käyttää. Tulevaisuudessa hyvä MPEG-4-tuki saattaa parantaa Quicktime Playerin asemaa kilpailussa muita mediasoittimia vastaan.

Microsoftin Windows Media Player [Windows Media 2004] on kaikkein laajimmalle levinnyt soitto-ohjelma, koska se tulee Windows-käyttöjärjestelmien mukana. Aluksi se oli tarkoitettu pelkästään paikallisten medialeikkeiden katseluun, mutta verkkomedian tuki tuli mukaan pikkuhiljaa 2000-luvun puolella ja tällä hetkellä uusin 10-versio on jo täysin kilpailukykyinen myös ominaisuuksiensa puolesta verrattuna esimerkiksi verkkomedian puolella syntyneeseen RealPlayeriin.

Windows Media Player tukee luonnollisesti parhaiten Microsoftin omia verkkovideoformaatteja. Muiden formaattien tuki on puutteellinen, ohjelma ei tue esimerkiksi MPEG-4-formaattia ollenkaan. Lisäksi Microsoft tukee käyttöjärjestelmälustoina vain eri Windows-versioita, joten tälle mediasoittimelle räätälöity sisältö rajaa muiden alustojen käyttäjät pois.

Microsoftin soitto-ohjelma ei tue SMIL-kieltä, vaan SMIL-tuki on sijoitettu Internet Explorer -selaimen. Yritys oli mukana jo SMIL 1.0:n kehitystyössä, mutta vetäytyi työryhmästä aivan loppuvaiheessa. Se teki oman SMIL-ehdotuksensa W3C:lle pian 1.0:n julkaisemisen jälkeen nimeltään HTML+TIME (Timed Interactive Multimedia Extensions), jonka ideana oli käyttää SMIL-pohjaisia laajennuksia HTML-sivujen sisällä. Microsoft liittyi takaisin SMIL 2.0 -työryhmään ja jatkoi valitsemaansa linjausta. Internet Explorer -selain tukee HTML:n SMIL-laajennusta eli SMIL-merkkausta HTML-merkkauksen sisällä. Se ei siis tue, kuten ei tue mikään muukaan Microsoftin ohjelmisto, itsenäisiä SMIL-tiedostoja kuten muut edellä esitellyt soitto-ohjelmat.

GriNS (Graphical Interface for SMIL) [GriNS 2004] on yksi vanhimmista SMIL-soitto-ohjelmista. Se on lähtöisin hollantilaisesta CWI-tutkimuslaitoksen projektista, jossa kehitettiin seuraavan kappaleen tuotanto-ohjelmistoissa mainittu Oratrix. Soitto-ohjelma kuuluu itse asiassa tuotanto-ohjelmistopakettiin, eikä se olekaan maksullisen levinnyt yleiskäyttöön. Eri mediaformaattien tuki tulee muiden soitto-ohjelmien kautta eli esimerkiksi RealMedia-sisältöjen katselu GriNS-ohjelmalla vaatii RealOnePlayerin asennuksen.

X-Smiles [X-Smiles 2004] on TKK:n Tietoliikenneohjelmistojen ja Multimedian Laboratorion kehittämä XML-selain, joka on toteutettu Javalla. X-Smiles toteuttaa SMIL Basic -profiilin.

## **6. Metsäkonesimulaatio-opiskelun verkkoympäristö: SIMUBA**

Osana Metsäkonesimulaatio-opetuksen tuotteistaminen -projektia (MetSimu) TTY:llä toteutettiin WWW-pohjainen oppimisympäristö SIMUBA tukemaan simulaattoripohjaista opiskelua [Ranta 2005][MetSimu 2004]. Oppimisympäristöstä löytyvät metsäkoneenkuljettajan koulutuksen opetussuunnitelma, hallitun puunkorjuuprosessin käsitteet, videoituja toimintamalleja sekä harjoituksia. Lisäksi omana toimintonaan on opiskelijan seurantapassi, johon opettajat merkitsevät suoritettavat tehtävät ja seuraavat opiskelijoiden etenemistä opetussuunnitelman mukaan. WWW-sovelluksessa on hyödynnetty projektissa tuotettuja oppimateriaalisiväläitä, jotka on tuotettu DVD-videoiksi ja seinäpostereiksi. [Häkkinen 2004b]

SIMUBA-ohjelmiston yhtenä komponenttina on videoiden arviointityöväline, jonka perusideana on mahdollistaa ajastettujen tekstikommenttien liittäminen videoleikkeisiin. Työkäytön avulla voidaan tuottaa sekä video-oppimateriaalia että toteuttaa harjoituksia, joissa opiskelijat tekevät omista simulaattorisuorituksista videotallenteet ja kommentoivat ne tehtävänannon mukaisesti. Lisäksi voidaan tuottaa ongelmakuvauksia sekä kommentoituja toimintamalleja harjoitusten tueksi. Videomateriaalin kommentointi on lähtökohdaltaan vaikeampaa kuin teksti- tai kuvamateriaalin, koska videotallenteessa on mukana aikaelementti ja kommentti halutaan yleensä kohdistaa tiettyyn aikajaksoon videotallenteessa. Arviointityövälineen tärkein ominaisuus onkin antaa sekä opettajille että opiskelijoille helppokäyttöinen käyttöliittymä tiettyyn videotallenteen kohtaan kohdistettavan kommentin lisäämiseksi.

SIMUBAssa on huomioitu lokalisointi ja siitä tehtiinkin projektin loppuvaiheessa englanninkielisellä käyttöliittymällä varustettu versio. Kaikkiin sisältöihin lokalisointi ei kuitenkaan ulotu, vaan harjoitustehtävien, toimintamallien ja muiden WWW-käyttöliittymän kautta tuotettujen sisältöjen lokalisointi vaatii käytännössä uuden SIMUBA-alustan asentamista ja sisältöjen uudelleen luontia halutulla kielellä. Kieliversioihin heränneen kiinnos-

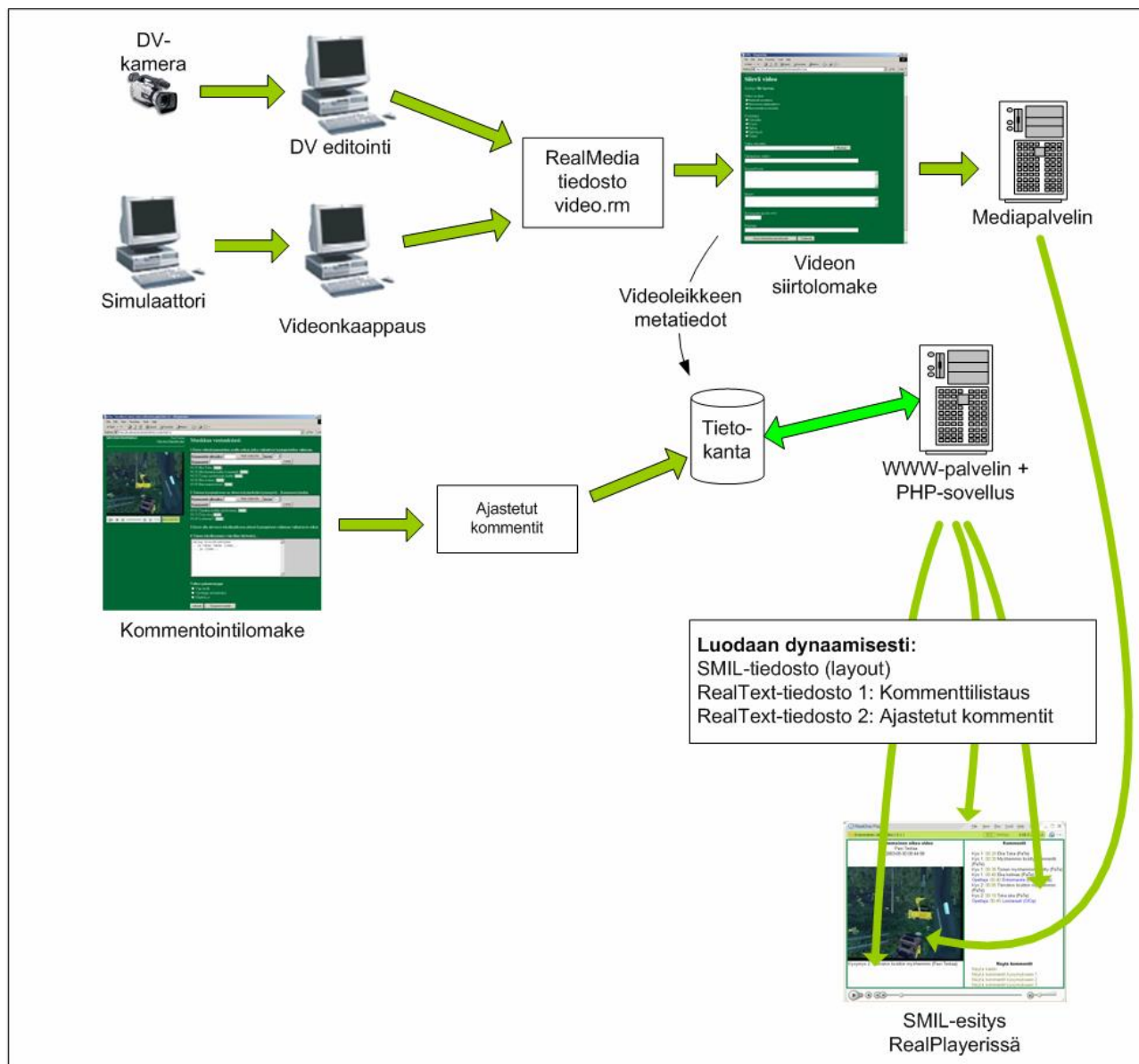


tuksen takia käsitteissä ja seurantapassissa huomioitiin kieliversiointi jo tietomallissa, mutta muiden toimintojen sisältöjen osalta aito kieliversiointi jäi myöhemmin toteutettavaksi.

## **6.1. Käyttöperiaate**

Kuvassa 6 on esitetty yksinkertainen prosessikaavio videotallenteiden tuottamisesta niiden katseluun kommentteineen. Työväline ei ota kantaa siihen, kuinka käytettävä videomateriaali on tuotettu. Tässä metsäkonesimulaatioprojektissa videomateriaalia syntyy kahdella eri tavalla: 1. valmis esimerkki- ja opetusmateriaali kuvataan digitaalivideokameralla joko luonnossa oikealla metsäkoneella tai simulaattoritietokoneen näytöltä ja 2. opiskelijoiden videotallenteet syntyvät videokaappaus-PC:llä, joka on kytketty simulaattori-PC:n näytönohjaimen. Ensimmäisessä tuottamistavassa saadaan hyvälaatuisia videomateriaalia, jota voidaan vielä editoida lopulliseen esitysmuotoon ennen kuin se pakataan julkaisumaattiin. Toinen tuottamistapa soveltuu parhaiten nopeaan ja yksinkertaiseen videoleikkeiden tuottamiseen, jossa simulaattorilta saatava kuva pakataan reaaliaikaisesti lopulliseen suoratoistoformaattiin.

Arviointityöväline astuu mukaan kuvaan siinä vaiheessa, kun videoleike on saatu pakattua suoratoistoformaattiin. Tässä projektissa formaatiksi valittiin RealMedia, mutta työväline ei itsessään rajoita formaattivalintaa. RealMedian on todettu toimivan hyvin SMIL-multimediakuvauskielen kanssa ja kun lisäksi Realin soitto-ohjelma on käytännössä ainoa vaihtoehto SMIL-soittimeksi, formaattivalinta oli yksinkertainen.



Kuva 6 Dynaaminen SMIL-esitysten luominen videoiden arviointityökalussa

## Videoleikkeen siirto

Tuotettu RealMedia-tiedosto siirretään WWW-selainkäyttöliittymän kautta palvelimelle, jonne videomateriaali tallennetaan. Työkalun käyttäminen sekä oppimateriaalin tuottamiseen että harjoitustehtävien toteuttamiseen näkyy jo videotiedoston siirtovaiheessa. Opettajien oletetaan siirtävän oppisisällöksi tulevia medialeikkeitä, jotka päätyvät videokirjastoon myös muiden opettajien hyödynnettäväksi. Jotta videokirjastosta olisi aidosti hyötyä, sinne tallennettavista videoista täytyy löytyä riittävästi hyödyllistä metatietoa. Opettaja-oikeuksilla siirrettävästä videoleikkeestä kysytään pakollisina tietoina videoleikkeen ot-

sikko ja kesto sekä metsäkoneen merkki. Lisäksi videoleikkeen siirtäjä voi halutessaan kuvailla videoleikkeen sisältöä muutaman vapaaehtoisesti täytettävän kentän avulla. Videoleikkeitä voidaan siten hyvin verrata LOM-metatietostandardin mukaisiksi oppimisasihioihin, vaikka LOM-mallia ei olekaan hyödynnetty SIMUBA:n metatietomallia määriteltäessä.

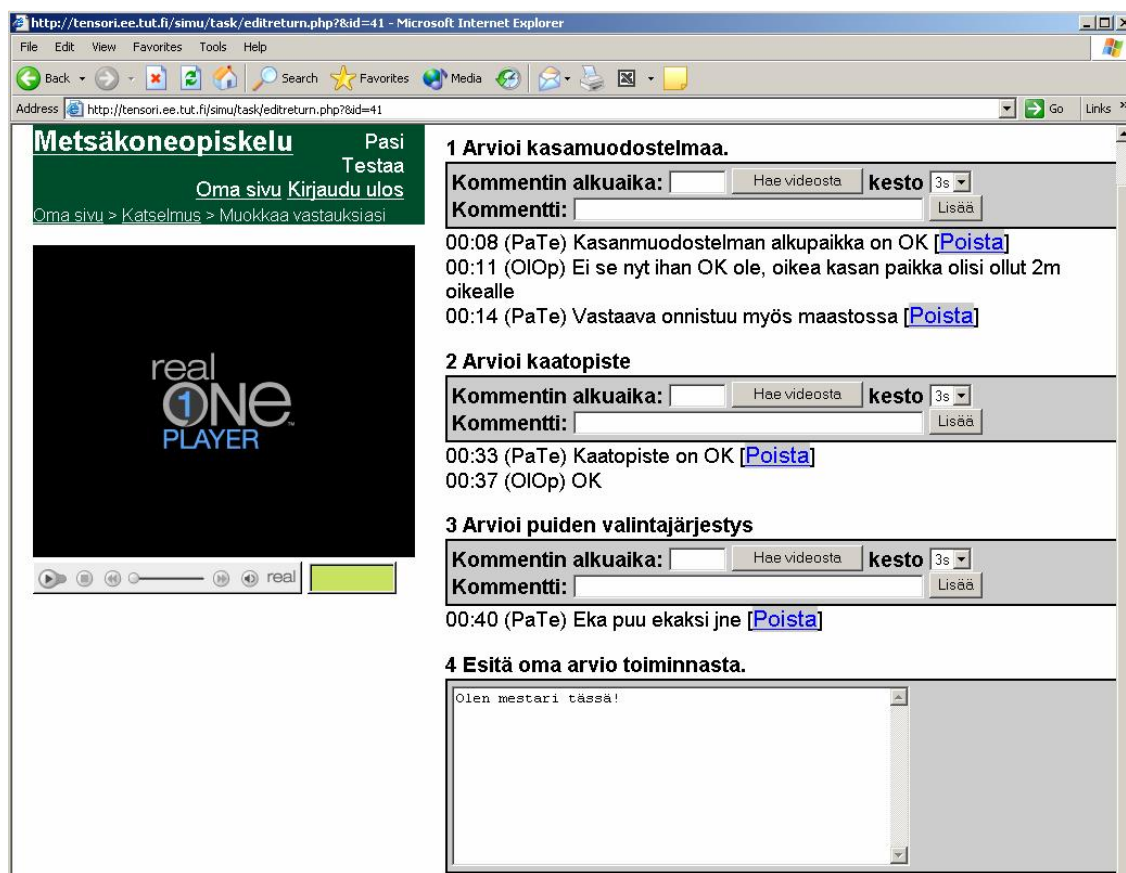
Opiskelijoiden videotallenteet sidotaan heti tietyn harjoitustehtävän palautukseen. Opiskelijoilla ei ole mahdollisuutta hallinnoida vastaavaa videokirjastoa kuin opettajilla ja tallennetut videoleikkeet ovat ainoastaan opiskelijan itsensä katseltavissa pois lukien harjoitustehtävän palautukseen mukaan liitetyt leikkeet, jotka ovat luonnollisesti opettajien katseltavissa. Opiskelija täyttää samat perustiedot omasta videoleikkeestään kuin opettajakin, mutta lisäkenttiä hänelle ei näytetä.

Varsinainen siirto tehdään WWW-lomakkeen avulla hyödyntäen WWW-selainten upload-toimintoa. Tämä siirtotapa sopii huonosti hyvin suurten tiedostojen siirtämiseen, mutta videoleikkeiden rajallisen pituuden ja hyvän pakkauksen ansiosta RealMedia-tiedostoista ei tule liian suuria. Käytetyillä RealMedia-pakkausparametreilla minuutin videoleikkeen tiedostokoko on noin 5 Mtavua riippuen siitä, mille yhteysnopeudelle se on optimoitu ja onko kyseessä SureStream-formaatti, jolla tallennetaan samaan tiedostoon useampi siirtonopeusversio samasta leikkeestä.

Videotiedosto tallennetaan palvelimelle sellaiseen paikkaan, josta mediapalvelin sen löytää. Videoleikkeen mukana WWW-lomakkeesta tulleet metatiedot tallennetaan tietokantaan lisättyinä tiedoilla, jotka työväline osaa itse selvittää. Näitä tietoja ovat videoleikkeen siirtäjä ja siirtoaika, tiedostokoko sekä videoleikkeen kuvakoko.

## **Videoleikkeiden kommentointi**

Ajastettujen kommenttien lisääminen videoleikkeisiin tapahtuu arviointilomakkeella (kuva 7). Lomake on periaatteeltaan sama sekä opettajille että opiskelijoille, lomakkeen varsinainen sisältö vaihtelee tilanteittain. Kuvassa olevaa lomaketta käytetään harjoitustehtävien palautuksissa. Kommentointinäkylässä vasemmalle on upotettu videoleike kontrollipainikkeineen. Oikealla puolella omassa kehyksessään on varsinainen kommentointilomake.



Kuva 7 Videoiden kommentointilomake

Kun käyttäjä haluaa lisätä kommentit tiettyyn kohtaan videoleikettä, hän voi hakea kommentin oikean alkuaian joko katselemalla videota ja painamalla 'Hae videosta' –painiketta, jolloin lomake hakee oikean aikakoodin videosta, tai käyttäjä voi kirjoittaa aikakoodin suoraan alkuaika-kenttään. Kommentin kesto aika valitaan alavetovalikosta ja varsinainen kommentti kirjoitetaan Kommentti-kenttään. Kuhunkin kysymykseen lisätyt kommentit listataan kommentointikenttien alle aikakoodeineen ja käyttäjän nimikirjaimilla varustettuna. Käyttäjä voi poistaa omat kommenttinsa yksitellen Poista-linkin avulla. Kommentit talletetaan muokausvaiheessa istuntodataan, jolloin kommentit eivät katoa, vaikka käyttäjä tarkoituksella tai vahingossa poistuisi kommentointinäköymästä. Tietokantaan kommentit talletetaan vasta kun käyttäjä painaa Lähetä-painiketta.

## Opettajan siirtämien videoleikkeiden hyödyntäminen

Sovelluksessa opettaja näkee omat ja muiden opettajien siirtämät videoleikkeet videokirjastonäkymässä (kts kuva 8). Videokirjasto esitetään taulukkomuodossa, jossa ovat näkyvissä videoihin liittyvät tärkeimmät metatiedot ja luokittelut. Opettajalla on mahdollisuus järjestää listausta halumansa sarakkeen mukaan, mikä helpottaa oikean videoleikkeen valintaa. Opettajien siirtämien videoleikkeiden käyttöoikeuksien hallintaa ei tähän ohjelmiin toteutettu, koska nähtiin, että sille ei ole tarvetta tässä ympäristössä ja se olisi turhaan monimutkaistanut toisten opettajien videoleikkeiden hyödyntämistä.

The screenshot shows a web browser window displaying the 'Videokirjasto - METSÄKONEOPISKELU' interface. The page title is 'METSÄKONEOPISKELU' and the URL is 'http://simuba.pkky.fi/simuba/teacher/videolibrary.php'. The page features a navigation menu with links like 'Etusivu', 'Oma sivu', 'Opetussuunnitelma', 'Käsitteet', 'Toimintamallit', 'Harjoitukset', 'Opeopas', and 'Ohje'. Below the navigation, there is a section titled 'Videokirjasto' with a link 'Lisää uusi videotallenne'. The main content is a table of video clips with columns for 'Otsikko', 'Konemerkki', 'Kesto', 'Lähde', 'Pääkäyttökohde', 'Tuotantotapa', 'Siirtopvm', and 'Siirtäjä'. The table contains 15 rows of data, including titles like '3.2. Siirtoauton tarkistukset' and '4.1.1 Valmet Työmaatietojen va...'. The status of each clip is indicated by a color-coded box: yellow for 'Ponsse' and red for 'Valmet'.

	Otsikko	Konemerkki	Kesto	Lähde	Pääkäyttökohde	Tuotantotapa	Siirtopvm	Siirtäjä
<input type="radio"/>	<a href="#">3.2. Siirtoauton tarkistukset</a>	Ponsse	00:02:11	maasto	Toimintamalli	tuotettu	18.03.2004	PeRa
<input type="radio"/>	<a href="#">3.3. Hakkuukoneen kuormaus sii...</a>	Ponsse	00:02:16	maasto	Toimintamalli	tuotettu	18.03.2004	PeRa
<input type="radio"/>	<a href="#">3.4. Siirtoauton varustelu ja ...</a>	Ponsse	00:03:12	maasto	Toimintamalli	tuotettu	18.03.2004	PeRa
<input type="radio"/>	<a href="#">3.5. Siirtoautoon kohdistuva p...</a>	Ponsse	00:01:59	maasto	Toimintamalli	tuotettu	18.03.2004	PeRa
<input type="radio"/>	<a href="#">4.1.1 Valmet Työmaatietojen va...</a>	Valmet	00:00:58	maasto	Toimintamalli	tuotettu	17.05.2004	PeNe
<input type="radio"/>	<a href="#">4.1.2 Valmet Leimikon sijainti</a>	Valmet	00:00:21	maasto	Toimintamalli	tuotettu	17.05.2004	PeNe
<input type="radio"/>	<a href="#">4.1.3 Valmet Työohjeistukset</a>	Valmet	00:01:12	maasto	Toimintamalli	tuotettu	17.05.2004	PeNe
<input type="radio"/>	<a href="#">4.1.4 Valmet Apteeraustiedosto...</a>	Valmet	00:01:37	maasto	Toimintamalli	tuotettu	17.05.2004	PeNe
<input type="radio"/>	<a href="#">4.1.5 Valmet Leimikon apterau...</a>	Valmet	00:00:32	maasto	Toimintamalli	tuotettu	17.05.2004	PeNe
<input type="radio"/>	<a href="#">4.2.1 Varasto- ja varikkopaika...</a>	Valmet	00:00:56	maasto	Toimintamalli	tuotettu	26.04.2004	PeNe
<input type="radio"/>	<a href="#">4.2.2 Varaston tilatarve</a>	Valmet	00:01:10	maasto	Harjoitustehtävä	tuotettu	19.04.2004	PeNe
<input type="radio"/>	<a href="#">4.2.3 Varastoalueen raivaus</a>	Valmet	00:00:38	maasto	Harjoitustehtävä	tuotettu	19.04.2004	PeNe
<input type="radio"/>	<a href="#">4.2.4 Hakkuutyömaan merkintä</a>	Muu	00:00:30	maasto	Toimintamalli	tuotettu	17.05.2004	PeNe

Kuva 8 Opettajan videokirjastonäkymä

Opettajat voivat hyödyntää videoleikkeitä tehdessään SIMUBAssa opetusmateriaalia toimintamalliesimerkkien muodossa tai määritellesään harjoitustehtäviä. Toimintamallit ovat videomuodossa olevia esimerkkisuorituksia, jotka ovat joko videokuvattu oikean metsäkoneen kanssa tai tallennettu simulaattorilta. Opettaja voi käyttää videoleikettä sellaisenaan ilman kommentointia (vrt. kuva 9) tai hän voi halutessaan merkitä tärkeitä koh-

tia suorituksessa tai kirjoittaa muita tarkennuksia ajastettujen kommenttien avulla. Vaikka esimerkin toimintamallissa ei olekaan kommentteja, se esitetään kuitenkin SMIL-esityksenä, mikä mahdollistaa tekstielementtien helpon päivitettävyyden, räätälöitävyyden sekä kieliversioinnin.



Kuva 9 Toimintamalli ilman ajastettuja kommentteja

Videoleikkeiden toinen pääkäyttökohde ovat harjoitustehtävät. Jokaiseen harjoitustehtävään voidaan liittää videoleikkeen lisäksi yksi kuva sekä useita liitetiedostoja. SIMUBA tukee neljän tyyppisiä tehtäviä:

- 1) perustehtävä: perinteinen lomakekysymys, jossa video toimii vain taustamateriaalina
- 2) toimintamallin arviointi: opettaja valitsee videokirjastosta videoleikkeen, johon opiskelija lisää ajastetut kommentit tehtävänannon perusteella
- 3) oman toimintamallin arviointi: tehtävänannon perusteella opiskelija tekee halutun suorituksen simulaattorilla, josta suoritus tallennetaan videoleikkeeksi. Opiskelija siirtää video-

tiedoston palvelimelle ja arvioi suorituksensa kysymysten perusteella mm. käyttämällä ajastettuja kommentteja. Opettaja voi halutessaan liittää tehtävänantoon videoleikkeen videokirjastosta.

4) opetustuokio: teoriaosuus, joka toimii yhden tai useamman oikean tehtävän taustamateriaalina, ei sisällä vastausosuuia.

Seuraavassa kuvassa on näkymä harjoitustehtäväkirjastosta, jonne opettajat tuottavat harjoitukset ja josta ne julkaistaan erikseen opiskelijoiden nähtäville.

The screenshot shows a web browser window titled 'Tehtäväkirjasto - METSÄKONEOPISKELU - Mozilla Firefox'. The address bar shows the URL 'http://simba.pkky.fi/simba/teachers/tasklibrary.php?sortorder=task\_title'. The page content includes a navigation menu with links like 'Etusivu', 'Oma sivu', 'Opetussuunnitelma', 'Käsitteet', 'Toimintamallit', 'Harjoitukset', 'Opeopas', and 'Ohje'. Below the menu is a section titled 'Tehtäväkirjasto' with a link '>> Luo uusi harjoitustehtävä'. There are buttons for 'Julkaise harjoitustehtävä', 'Muokkaa tehtävää', 'Kopioi uuden pohjaksi', and 'Poista'. A table lists tasks with the following columns: Moduuli, Otsikko, Tehtävätyyppi, Konemerkki, Taso, Aika, Laajitus, and Paikka.

Moduuli	Otsikko	Tehtävätyyppi	Konemerkki	Taso	Aika	Laajitus	Paikka
M 2: Johdatus koneelliseen puunkorj...	<a href="#">Hakkuukonemittausohje</a>	Perustehtävä	C P T V	Syv	06.04.2004	PeNe	Ti Lu
M 2: Johdatus koneelliseen puunkorj...	<a href="#">Ajokoneen kuormaimen käyttö</a>	Toiminta-arvio	C P T V	Per	20.02.2004	MaHä	Muu
M 12: Aj...	<a href="#">Moduuli 2: Johdatus koneelliseen puunkorjauseen 2: Ajokoneen käytön arviointi suunnittelu</a>	Perustehtävä	-	Syv	23.02.2004	MaHä	Ma
M 12: Ajouraverkoston suunnittelu	<a href="#">Ajouraverkoston suunnittelu1</a>	Perustehtävä	M	Per	08.10.2004	PeNe	Lu
M 12: Ajouraverkoston suunnittelu	<a href="#">Ajouraverkoston suunnittelu2</a>	Perustehtävä	M	Per	08.10.2004	PeNe	Lu
M 12: Ajouraverkoston suunnittelu	<a href="#">Ajouraverkoston suunnittelu3</a>	Perustehtävä	M	Per	08.10.2004	PeNe	Lu
M 12: Ajouraverkoston suunnittelu	<a href="#">Ajourien suunnittelu</a>	Opetustuokio	-	Syv	23.02.2004	MaHä	Lu
M 3: Koneellisen puunkorjuun perust...	<a href="#">Hakkuukoneen mittarit ja valvontalaitteet</a>	Perustehtävä	C P T V	Per	10.06.2004	PeNe	Ma Lu
M 1: Johdatus koneelliseen puunkorj...	<a href="#">Hakkuukonemittauksen toimintaperiaate</a>	Opetustuokio	C P T V	Per	23.03.2004	MaHä	Lu Muu
M 1: Johdatus koneelliseen puunkorj...	<a href="#">Hakkuukonemittauslaki</a>	Perustehtävä	-	Per	24.03.2004	PeNe	Ti Lu

Kuva 10 Harjoitustehtäväkirjasto

Videopohjaisen oppimateriaalin käyttö verkko-oppimateriaalina koetaan yleensä vaivalloiseksi useistakin eri syistä, joita on kuvattu edellisissä luvuissa. MetSimu-hankkeeseen osallistuneet metsäkonealan opettajat ottivat kuitenkin hyvin videosisällön käyttöön SIMUBA-ohjelmiston avulla ja tuottivat itse harjoitustehtävät ja toimintamallit tuotettujen

videoleikkeiden päälle. Opettajat kokivat videoleikkeiden siirron, videokirjaston käytön ja videoiden liittämisen harjoitustehtäviin yksinkertaisina toimenpiteinä ja ottivat ohjelmiston mielellään käyttöön. On toki huomattava, että suurin osa videoleikkeistä on ammattilaisten tuottamaa, joten vaativin työvaihe eli itse videoleikkeiden kuvaaminen ja editointi jäi suurelta osin pois, mutta jo projektin aikana muutama opettaja kokeili onnistuneesti omien videoleikkeiden tuottamista videoeditointiohjelman avulla PC:llä.

Videoleikkeet, joita SIMUBA-ympäristössä käytetään toimintamalleissa ja harjoitustehtävien liitteinä, ovat suurelta osin ammatti-ihmisten tekemiä ja ammattikalustolla tuotettuja. Näin ollen ne ovat erittäin hyvälaatuisia. Alkuperäiset videoleikkeet on tarkoitettu DVD-julkaisua varten ja verkkokäyttöä varten ne on pakattu RealMedia-formaattiin bittivirtanopeudella (bitrate) 512 kbit/s käyttäen RealVideo 9 -koodekkia. Käytännössä videoleikkeet ovat siis katseltavissa ilman ylimääräisiä odotusaikoja perus-ADSL-liittymän kautta. Videoleikkeiden kuvanlaatu riittää videokuvan skaalaamiseen koko näyttöruudun kokoiseksi, jolloin kuva vastaa hyvälaatuisia VHS-nauhaa.



## 6.2. Tekninen toteutus

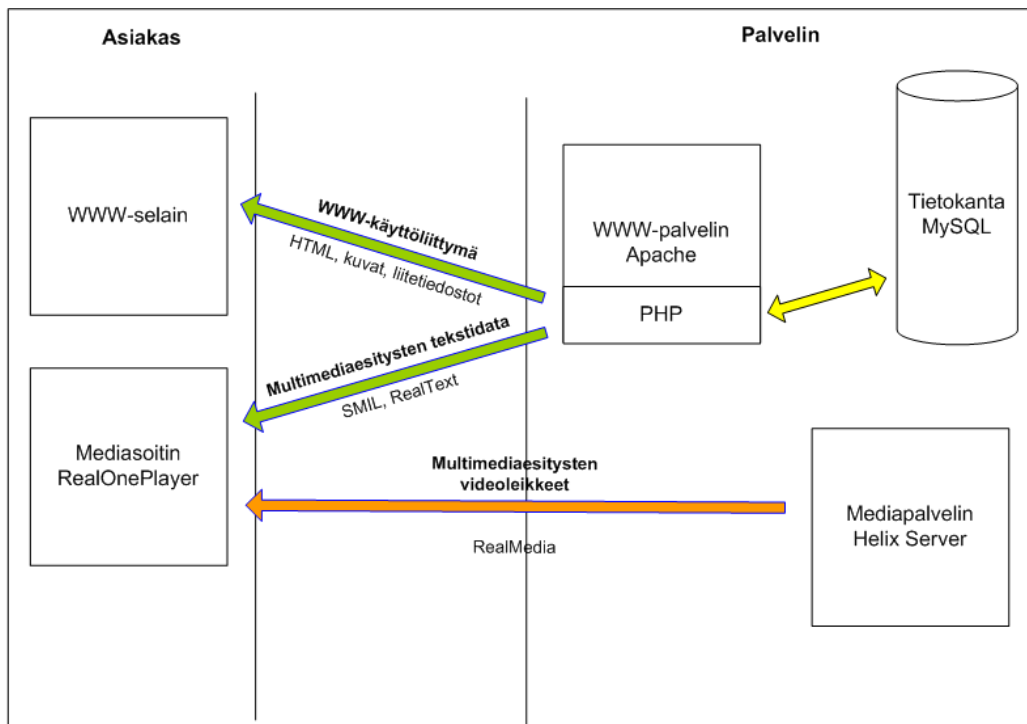
Arviointityövälineen tekniseksi alustaksi haluttiin mahdollisimman avoin ohjelmistoympäristö, jotta työvälineen siirrettävyys olisi mahdollista niin teknisesti kuin kustannustenkin kannalta. Työvälineen käyttöliittymä ja toiminnallisuus päätettiin toteuttaa PHP-kielellä [PHP 2004], joka on yksi yleisimmin käytetyistä tekniikoista dynaamisten WWW-sivujen toteuttamisessa. PHP:n ideana on upottaa WWW-palvelimella suoritettavaa ohjelmakoodia HTML-merkkauksen sisään. WWW-palvelimeen asennetaan lisämoduuli, joka huolehtii PHP-tiedostojen prosessoinnista. Vastaavia kilpailevia tekniikoita ovat lähinnä Windows-ympäristössä käytettävä ASP (Active Server Pages) ja Java-kieleen perustuva JSP (Java Servlet Pages). Ideana kaikilla näillä tekniikoilla on mahdollistaa nopea WWW-sovelluskehitys yhdistämällä HTML-käyttöliittymään sovellus- ja tietokantalogiikka.

Työvälineen suunnittelun lähtökohtana oli rakentaa koko järjestelmä tietokannan päälle. Tietokantavalinta kohdistui luonnostaan MySQL-ohjelmistoon [MySQL 2004], joka on eniten käytetty tietokanta PHP-sovelluksissa. PHP:n tavoin MySQL kuuluu useimpiin Linux-jakeluversioihin automaattisesti. Avoimen lähdekoodin ohjelmistona MySQL:n saa ilmaiseksi asennettua myös muille Unix-alustoille ja Windows-koneille.

Käyttöjärjestelmäalustaksi valittiin Linux, koska se oli tuttu kehitystiimille aikaisemmista ohjelmistokehityshankkeista ja se koettiin luotettavammaksi alustaksi kuin Windows-ympäristö. Arviointityövälinettä ei ole kuitenkaan sidottu Linux-ympäristöön, vaan kehitetty ohjelmisto toimii myös Windows-alustalla, kunhan PHP ja MySQL-tietokanta ovat käytettävissä. Itse asiassa suuri osa SIMUBAn kehitystyöstä tehtiinkin Windows-alustalla.

Kuvassa 11 on esitetty arviointityökalun asiakas-palvelin -malli. Kolmitasoinen arkkitehtuuri mahdollistaa palvelintoteutuksen hajauttamisen usealle eri palvelinkoneelle, mutta tässä toteutuksessa niin WWW-palvelin, tietokanta kuin mediapalvelinkin sijaitsevat samalla palvelimella. Näistä kolmesta tietokanta on kaikkein helpoin hajauttaa uudelle palvelimelle tietokantayhteyden luonteen takia. WWW-palvelimen ja mediapalvelimen välil-

lä on tiedostotason riippuvuussuhde, koska mediapalvelimen sisältö, videoleikkeet, siirretään asiakkaalta WWW-palvelimen kautta. Tässä toteutuksessa ongelmaa ei synny, koska käytössä on yhteinen tiedostojärjestelmä. Jos mediapalvelin halutaan sijoittaa fyysisesti eri palvelimelle kuin WWW-palvelin, PHP-sovelluksen täytyy pystyä siirtämään videotiedostot jollakin tekniikalla WWW-palvelimen tiedostojärjestelmästä mediapalvelimen tiedostojärjestelmään. Yhteisen verkkolevyn käyttö on yksi helpoimmin toteutettavista vaihtoehdoista.



Kuva 11 Arviointityökalun asiakas-palvelin -malli

Itsenäisen mediapalvelimen käyttö tulee ajankohtaiseksi silloin, kun yhden palvelinkoneen tai verkkoyhteyden kapasiteetti ei riitä palvelemaan kaikkia yhtäaikaista asiakkaita. WWW-palvelimen tarvitsema verkkokapasiteetti arviointityövälineen käyttöliittymän ja tekstimuotoisen datan siirtämiseen on huomattavasti pienempi kuin mediapalvelimen kais-tavaatimukset videokuvan välittämiseen. Jos työvälineen käyttäjät ovat maantieteellisesti laajalla alueella usean eri operaattorin verkoissa, voi olla siirtokaistan käytön optimoinnin kannalta järkevää sijoittaa mediapalvelin jopa täysin eri paikkaan kuin varsinainen sovel-luspalvelin. Käytetyn SMIL-tekniikan ansiosta mediaesitysten elementit voidaan koota täysin vapaasti eri palvelimilta. Tämä mahdollistaa esimerkiksi keskitetyn ja nopeiden

verkkoyhteyksien päähän sijoitetun järeän mediapalvelimen hyödyntämisen myös tässä sovelluksessa. WWW-palvelin voidaan jättää heikompien verkkoyhteyksien taakse.

## **Palvelinpään toteutus**

Palvelinpäässä kaikki sovelluslogiikka ja käyttöliittymän muodostaminen hoidetaan PHP:llä. PHP-ohjelmilla huolehditaan niin käyttäjien autentikointi, WWW-sivujen generointi, tiedostojen siirtäminen palvelimelle kuin SMIL-esityksissä tarvittavan tekstimuotoisen data tuottaminen. Kaikki järjestelmän käyttämä tieto video-, kuva- ja liitetiedostoja lukuun ottamatta tallennetaan tietokantaan.

Arviointityökalu käyttää PHP:n tarjoamaa istuntomallia, joka mahdollistaa käyttäjien tunnistamisen eri sivupyynnöiden välillä ja sisällön räätälöinnin käyttäjäkohtaisesti. Käyttäjätietojen perusteella huolehditaan myös käyttöoikeuksien tarkistaminen eli opiskelijatunnuksilla ei ole pääsyä esimerkiksi opettajan työvälineisiin.

PHP:n istunto sallii istuntokohtaisen tietojen tallettamisen ja hyödyntämisen palvelimella ilman, että tietoja täytyisi kierrättää tietokannan kautta. Tätä hyödynnetään mm. harjoitustehtävien vastausvaiheessa, jolloin videoleikkeeseen lisätään aikakommentteja. Kommentteja lisätään yksitellen ja ne tallennetaan väliaikaisesti istuntoon ennen kuin käyttäjä painaa varsinaista Tallenna-painiketta. Jos käyttäjä poistuu vahingossa tai tahallaan kommenttilomakkeesta, tiedot säilyvät palvelimella ja kun käyttäjä palaa saman harjoituksen tehtäväsivulle, aikaisemmin lisätyt kommentit ja vastaukset ovat tallessa. Tällä tekniikalla toisaalta estetään käyttäjää turhauttavat tietojen katoamiset virhepainallusten takia ja toisaalta vältetään ylimääräistä tietokantaliikennettä verrattuna tilanteeseen, jossa jokainen kommentin lisäys tallennettaisiin aina suoraan tietokantaan.

Vaikka PHP on tarkoitettukin lähinnä HTML-muotoisten sivujen tulostamiseen, sillä voidaan tuottaa tai välittää myös muun tyyppistä sisältöä aina dynaamisesti luotavista kuvatiedostoista lähtien. Tässä sovelluksessa kaikki liite- ja kuvatiedostot ajetaan PHP-ohjelman läpi käyttäjäoikeuksien tarkistamisen ja palvelinpään hakemistorakenteen piilottamisen takia. SMIL-esitykset luodaan täysin dynaamisesti tietokannasta saatavien tietojen perusteella. Varsinaisen esityksen ulkoasun ja mediaelementit määrittelevän SMIL-

tiedoston lisäksi PHP:llä muodostetaan myös RealText-tiedostot, jotka sisältävät esitysten tekstimuotoisen datan: ajastetut kommentit aikakoodeineen, navigointilinkit kommenttien alkamisajankohtiin ja yleistiedot tallenteesta. RealText-formaatin käyttöön päädyttiin, koska SMIL ei tue hyvin tekstimuotoisen sisällön ulkoasun muotoilua ja ei ole olemassa yhtä yhteensopivaa ajastetun tekstin formaattia. W3C:ssä on aloitettu työ tällaisen mediasoitimesta riippumattoman formaatin kehitystyö, mutta suositus ei ollut valmis tämän projektin aikana eikä yksikään mediasoitin tukenut uutta Timed Text -formaattia [Timed-Text 2004].

Dynaamisten multimediaesitysten tuottamiseen PHP soveltuu siinä missä mikä tahansa muukin WWW-ohjelmointikieli. PHP ei tue SMILiä millään valmiilla kirjastofunktioilla ja valmiita kirjastoja SMILin tuottamiseen dynaamisesti ei ole muutenkaan ainakaan yleisesti saatavilla. Sama pätee muihinkin ohjelmointikieleen, ainoastaan Perl:illä on tehty yksi vapaasti käytettävissä oleva moduuli dynaamiseen SMIL-tiedostojen luontiin [PerlySMIL 2003]. Tämä osoittaa sen, että SMIL-esitysten dynaaminen tuottaminen on vielä varsin vähän käytetty toiminnallisuus verkossa. Staattisia SMIL-tuottavia editoreita on olemassa jo useita.

Työssä päädyttiin käyttämään mediaelementtien vakioasettelua toistensa suhteen SMIL-esityksissä. Ainoa muuttuja on videoleikkeen kuvakoko, jonka mukaan määräytyy myös koko SMIL-esityksen korkeus ja leveys. SMIL-esitysten dynaaminen tuottaminen mahdollistaa sivupohjien käytön eli samat medialeikkeet voidaan asemoida katseluvaiheessa eri tavoin joko esityskohtaisesti tai jopa katselijakohtaisesti, mutta SIMUBA-ohjelmistossa tälle ei nähty tarvetta. Liitteessä 1 on tiedostolistauksia tyypillisestä SMIL-esityksestä.

Ulkoisista ohjelmista arviointityökalun palvelinosuus käyttää RealNetworksin rmedit-ohjelmaa, joka on tarkoitettu pääasiassa komentoriviltä tapahtuvaan RealMedia-tiedostojen yksinkertaiseen muokkaamiseen [rmedit 2004]. rmedit-ohjelma kuuluu osana RealProducer-ohjelmapakettia. Arviointityökalu hyödyntää rmedit-ohjelmaa videotiedostojen siirtovaiheessa. rmedit:illä tarkistetaan, että siirretty tiedosto on RealMedia-formaatissa ja selvitetään kuvakoko ja muutama muu tekninen metatieto videoleikkeestä. Videoleikkeen kestoa rmedit-ohjelmalla ei kuitenkaan saa selville, mikä on selkeä puute tämän tyyppisessä käytössä. Pituustiedon välittäminen jää videon siirtäjän vastuulle.

## **Asiakaspään toteutus**

Arviointityökalu vaatii asiakaspäässä WWW-selaimen, jossa toimii RealPlayerin selainlaajennus (plug-in) eli käytännössä Microsoft Internet Explorerin tai Mozillan. Tämä johtuu siitä, että videoleikkeiden kommentointi on toteutettu upottamalla RealMedia-leike HTML-sivulle kuvan 7 mukaisesti. Kommentointivaiheessa hyödynnetään RealPlayerin selainlaajennuksen JavaScript-rajapintaa, jonka avulla selain voi sekä saada tilatietoja videoleikkeen soistosta että antaa komentoja soitto-ohjelmalle. Tämä näkyy käytännössä siten, että kommentointilomakkeeseen saadaan tarkka aikakoodi videoleikkeen sen hetkisestä soittokohdasta lomakkeen painiketta painamalla, ja videoleikkeessä voidaan hypätä haluttuun kohtaan HTML-sivulla olevan aikakoodilinkin avulla. Tämä helpottaa navigointia videoleikkeen sisällä.

Kommentoitujen videoleikkeiden eli käytännössä SMIL-esitysten katselu tapahtuu erillisessä RealPlayer-ikkunassa. SMIL-esitykset on toki mahdollista upottaa HTML-sivulle pelkän videon tavoin, mutta erillisen soitto-ohjelmaikkunan käytöllä haluttiin sallia soittokkunan koon muuttaminen ja arviointityövälineen vapaa käyttö samanaikaisesti videotallenteen katselun aikana. Erillisen ikkunan käyttöä puoltaa myös sen parempi luotettavuus. Upotettu videoikkuna on riippuvainen selaimen ja soitto-ohjelman yhteensopivuudesta, missä on havaittu ongelmia erilaisilla selain - RealPlayer -yhdistelmillä. Lisäksi selaimen ja RealPlayerin välimuistiasetukset vaikuttavat merkittävästi siihen, kuinka luotettavasti dynaamisesti luotu viittaus videoleikkeeseen toimii HTML-sivulle upotetun medialeikkeen tapauksessa.

Liitteeseen 2 on kerätty versiotiedot käytetyistä palvelin- ja asiakaspään ohjelmistoista.

## **6.3. Toteutuksen arviointi**

Tuotantoprosessin näkökulmasta sisällöntuotanto MetSimu-hankkeessa jakautui liiankin selvästi kahteen eri osaan eli videotuotantoon ja ohjelmistotuotantoon. Videotuotannon toteutus ulkoistettiin aivan perustellusti videoammattilaisten tehtäväksi, käsikirjoitus syntyi hankkeeseen osallistuneiden opettajien voimin. Videotuotanto onnistui hyvin, sillä sen tärkeimmät lopputuotteet, video-DVD-muotoon tuotettu oppimateriaali, sai kiitosta niin

projektin osapuolilta kuin ulkopuolisiltakin ja DVD:t päätyivät maailmanlaajuiseen jakeluun konevalmistajien kautta. Yhdeksi tärkeimmäksi onnistumisen syyksi ammattitaitoisten tekijöiden lisäksi projektissa nähtiin videotuotannon pilotointi. Kolmevuotisen hankkeen ensimmäisenä vuonna tehtiin ensimmäiset videokuvaukset sekä harvesterisimulaattoreilta että oikeilla koneilla. Tekniseltä laadultaan tuotokset olivat ihan hyviä, mutta videoleikkeiden kevyesti tehty käsikirjoitus näkyi selvästi lopputuloksessa. Vaikka ensimmäisten kuvausten tuotantoa ei testikäytön lisäksi opetuksessa käytettykään, toimi se vankkana pohjana hankkeen toisena vuonna käynnistyneelle huomattavasti tarkemmalla videotuotannon sisältöjen suunnittelulle ja videokuvausten käsikirjoittamiselle. Pilotoinnista huolimatta lopullisten videoleikkeiden saaminen julkaisukuntoon selostusraitoineen ja itse DVD-levyjen toteutus osoittautuivat yllättävän työläiksi, minkä takia suunnitellussa resursoinnissa ei aivan pysytty.

MetSimu-hankkeen ohjelmistotuotanto eli SIMUBA-verkkoympäristön toteuttajat oli koko ajan tietoisia hankkeen muun sisällöntuotannon etenemisestä, mutta verkkojulkaisemisen vaatimuksia tai toiveita ei juurikaan välittynyt takaisinpäin sisällöntuotantoon. Yksinkertaisimmillaan puhutaan video- ja kuvatiedostojen nimeämispolitiikasta, joka tässä hankkeessa nousi tavallista merkityksellisemmäksi konemerkki- ja kieliversioiden takia. Esimerkiksi jokaisesta käsitekuvasta syntyi kuusi eri versiota: kolme merkkikohtaista versiota ja niistä jokaisesta kaksi eri kieliversiota. Tiedostojen nimeäminen tehtiin DVD-tuotannon ja ihmiskäyttäjien näkökulmasta, mikä ei vastannut ohjelmallisen käsittelyn vaatimuksia. Sisällöntuotannosta syntyi myös paljon metatiedoksi laskettavaa tietoa, jota ei kuitenkaan hyödynnetty SIMUBAssa, koska tätä ei oltu ajateltu ohjelmiston suunnitteluvaiheessa eikä siten ollut mitään määrityksiä siitä, kuinka nämä tiedot olisivat siirtyneet SIMUBAan ilman käsin tapahtuvaa kopiointia.

Metatietostandardeja ei käytetty muutenkaan hyväksi määriteltäessä SIMUBA-ohjelmiston metatietomalleja. Metatietomallin räätälöinti minimalistiseksi juuri tähän käyttötarkoitukseen yksinkertaisti tietojen syöttöä, mutta esimerkiksi videoiden tietoja ei voida hyödyntää muualla ja sisältöjen siirrettävyys muihin ympäristöihin vaatisi erikoisratkaisuja. Jos ohjelmisto tehtäisiin uudelleen, siinä hyödynnettäisiin valmiita metatietomalleja erityisesti jo muualle tuotetun kuvailevan tiedon tuomisessa järjestelmään.

Luvussa 3 esitellyistä metatietostandardeista mielenkiintoisimmat SIMUBAn kannalta ovat LOM ja Dublin Core. Vaikka oppimisaihio-ajattelu ei ollutkaan suunnittelun lähtökohdana, toteutus erityisesti toimintamallien osalta noudattaa LOM:in ideoita varsin hyvin. Lisäämällä kuvailemaa tietoa ja luokittelemalla se laajemmin toimintamallien tuottamista-paa voitaisiin hyödyntää helposti myös tuotettaessa kommentoituja videosisältöjä SIMUBAn ulkopuolella käytettäväksi. Dublin Core ja erityisesti joku sen multimediakäyttöön tehty laajennus soveltuu hyvin videoleikkeisiin liittyvän metadatan tallennus- ja siirtomuodoksi.

Luvussa 2 esitetyn sisällöntuotantoprojektin vaiheet ovat löydettävissä SIMUBAn kehityksestä, vaikkakaan ei aivan yhtä selkeinä vaiheina eli ne ovat tapahtuneet osittain limittein, kuten varmaan monessa muussakin oikeassa multimediasisältöprojektissa. Ideointi ja suunnitteluvaiheet tapahtuivat käytännössä samanaikaisesti, eivätkä ne edes päättyneet toteutuksen käynnistyttyä. Käyttäjärhmiä määrittely ja mediavalintojen tekeminen olivat yksinkertaisia toimenpiteitä, mutta toimintojen ryhmittelyä ja niiden lukumäärää ja laajuutta ei pystytty määrittelemään tarkasti suunnitteluvaiheessa. Toimintojen tarkat ominaisuudet määrittyivät käytännössä vasta toteutuksen aikana, kun loppukäyttäjät näkivät toiminnot oikeasti käytössä. Tällainen iteratiivinen ohjelmistokehitys aiheutti luonnollisesti ylimääräistä työtä, mutta ohjelmiston modulaarisuuden ansiosta se ei tuottanut ylivoimaisia ongelmia vaikkakin vaikeuttivat aikataulussa pysymistä.

Kieliversioinnin merkitys tiedostettiin jo projektin alkuvaiheessa projektissa mukana olleiden konevalmistajien kansainvälisen toiminnan takia, mutta lokalisoinnin vieminen loppuun saakka aina sisältöihin asti SIMUBA-ohjelmistossa nähtiin liian vaativana ja työllään urakkana projektin alkuvaiheessa, joten ohjelmistossa tyydyttiin käyttöliittymän lokalisointimahdollisuuteen. Tämä mahdollisuus konkretisoitui projektin loppuvaiheessa, kun alkuperäisistä suunnitelmista poiketen SIMUBA-ympäristöstä tehtiin englanninkielinen versio jo hankkeen aikana. Kieliversioinnin huomioiminen onnistui ohjelmistossa hyvin, sillä uusi kieliversio syntyi muutamassa päivässä ja se toteutettiin siten, että käyttäjä pystyy vaihtamaan käyttöliittymän kielen itse vaikka kesken istunnon. SIMUBAn staattiset sisällöt saatiin myös kieliversioinnin taakse, joten ainoastaan itse ympäristössä tuotettavat sisällöt jäivät kieliversioinnin ulkopuolelle.

Muista MetSimu-projektin tuotteista, opetussuunnitelmasta, puunkorjuuprosessin käsitteistä sekä DVD-video-oppimateriaaleista, tuotettiin englanninkieliset versiot ja niitä toki hyödynnettiinkin WWW-sovelluksessa, mutta kehitettävää jäi tältä osalta. Sisällöntuotannon osalta myös sisällöltään täysin englanninkielisen SIMUBA-ympäristön luominen uuden asennuksen päälle ei ole juurikaan sen työläämpää kuin jos tehtävät ja toimintamallit olisi pystynyt luomaan jo alun alkaenkin monikielisinä, mutta päivitettävyyden kannalta erillisten kieliversioalustojen ylläpito on huomattavasti hankalampaa.

Ohjelmiston käytännön maksimisuorituskykyä ei testattu formaalisti, mutta muutamalla samanaikaisella käyttäjällä vasteajat jäivät kaikissa toiminnoissa alle sekunnin käytetyllä palvelimella (ks. liite 2). SIMUBAn raskaimmat toiminnot ovat opettajan roolilla käytettävät kirjastonäkymät, jotka tuottavat paljon tietokantahakuja jokaista taulukon luontia kohden. Suorituskyvyn pullonkaulana onkin tietokanta, sillä ohjelmiston sovelluslogiikka on suhteellisen yksinkertainen ja vaatii vähän laskentaa. Toisen pullonkaulan muodostaa yhtäaikainen videoleikkeiden katselu, joka rasittaa sekä palvelimen tiedostojärjestelmää että kuluttaa paljon tiedonsiirtokapasiteettia palvelimen lähiverkossa ja internetliittymässä.

Ohjelmisto paisui ominaisuuksien puolesta videoiden arviointityökalusta oppimisalustaksi ja suurin osa ohjelmointityöstä käytettiin muuhun kuin videomateriaalin käsittelylogiikan tuottamiseen. Tämän vuoksi SMIL-tekniikan mahdollistamaa multimediaesitysten dynaamisuutta ei hyödynnetty tässä työssä kovin pitkälle ja jatkoprojekteille jäi siten paljon kehitettävää.

Kehityskohteista tärkeimmät ovat SMIL-esitysten suurempi räätälöitävyys, ohjelmiston tuki videoleikkeiden versioinnille ja ulkoisten videoleikkeiden käytölle, laajempi videoformaattien tuki sekä dynaamisten sisältöjen julkaisutoiminto staattisille tallenteiksi. SMIL-esityksissä oli käytössä ainoastaan yksi julkaisupohja (template), josta käytettiin kolmea eri versiota esitettävien tekstisisältöjen mukaan. Julkaisupohjia pitäisi olla enemmän ja niiden pitää olla käyttäjän valittavissa, jos SIMUBAn ideoita halutaan hyödyntää yleiskäyttöisemmässä sisällöntuotannossa.



Videoleikkeiden versioinnille olisi ollut käyttöä jo MetSimu-hankkeessa, koska samoista videosisällöistä oli tuotettuna maksimissaan neljä eri versiota: kaksi eri konemerkkiä ja niistä kieliversiot suomeksi ja englanniksi. Kieliversioinnin huomioiminen tietokannan ja sovelluslogiikan suunnittelussa olisi ollut jälkikäteen katseltuna varsin pieni lisätyö, ja se olisi mahdollistanut oikeasti kieliversioidun SIMUBAn toteuttamisen. Ohjelmiston laajemman käyttökelpoisuuden kannalta myös useiden eri nopeusversioiden tukeminen samasta videoleikkeestä olisi hyödyllinen ominaisuus.

Projektissa ei tarvittu ohjelmiston tukea ulkoisille eli jo valmiiksi verkosta löytyville videoleikkeille, mutta jatkon kannalta se olisi erittäin hyödyllinen ominaisuus. Lisäksi olisi vakavasti harkittava muiden videoformaattien tukemista samasta syystä, sillä RealPlayer-soitto-ohjelmavalinta mahdollistaa Quicktime- ja MPEG-4-sisältöjen käytön.

SIMUBA-sisältöjen julkaisemista staattiseen tiedostomuotoon mietittiin hankkeen aikana, mutta se jäi toteuttamatta ajanpuutteen takia. Staattista tallennetta harjoituksista ja toimintamalleista voitaisiin hyödyntää tilanteissa, joissa verkkoyhteyttä ei ole ollenkaan tai se on liian heikkotasoinen videomateriaalien käytölle. Silloin CD- tai DVD-tallenne mahdollistaisi sisältöjen käytön WWW-selainkäyttöliittymästä ilman interaktiivisia ja personoituja toimintoja.

## 7. Yhteenveto ja johtopäätökset

Videopohjaisen sisällön käyttö verkossa on menettänyt hype-arvonsa ja on muuttunut arkipäiväiseksi toiminnaksi. Liikkuvalla kuvalla ei ole enää itseisarvoa, vaan sisältö on saanut arvoisensa huomion myös verkkomultimediassa ja verkko-oppimateriaaleissa videon käytön arkipäiväistymisen myötä.

Multimediasisältöjen tuottaminen on edelleen haasteellista työtä. Laadukas videotuotanto vaatii tekniikan kehitymisestä huolimatta ammattitaitoa sekä käsikirjoittamisessa että tuotantovaiheessa, jolloin tuotetun sisällön uudelleenkäytettävyys voi olla ratkaiseva kynnyskysymys resursseja laskettaessa ja tuotantopäätöstä tehtäessä. Toisaalta koulutustuotteiden käyttäjät ovat usein päätelaitteiden osalta varsin heterogeeninen ryhmä. Vaikka lankamodeemikäyttäjät alkavatkin olla vähemmistössä, uudet mobiilikäyttäjät asettavat kovia vaatimuksia sisältöjen taipumiselle pienelle näytölle ja laajakaistanopeuksia hitaammille yhteysnopeuksille.

Verkkomultimediasisältöjen dynaaminen tuottaminen ja katselijakohtainen räätälöinti tuovat ratkaisuja edellä mainittuihin haasteisiin. Kun tuotetut sisällöt pilkotaan riittävän pieniin osiin, kuvataan tarkasti metatietojen avulla ja käytössä on älykäs julkaisujärjestelmä, joka osaa koota yksittäisistä osista kuhunkin käyttötilanteeseen optimoidun esityksen, sisällöntuottaja pystyy löytämään ja käyttämään uudelleen joskus aikaisemmin tehtyjä sisältökomponentteja ja hänelle jää vapaat kädet hyödyntää valmista materiaalia eri tilanteissa eri tavoin. Lisäksi loppukäyttäjistä voidaan tehdä multimediamateriaalin sisällöntuottaja hänestäkin.

SMIL osoittautui tässä työssä hyväksi valinnaksi dynaamisen videopohjaisen multimediaoppimateriaalin tuottamisessa. Palvelinpään toteutus vaati ohjelmointipanosta, mutta tuo panos saatiin jo projektin aikana takaisin säästyneinä sisällöntuotantotunteina, koska videoleikkeiden julkaisu yksinkertaistui käytännössä kolmen WWW-lomakkeen täyttämiseen: videoleikkeen siirto, harjoitustehtävän tai toimintamallin luonti sekä sen julkaisu haluttuun kohtaan julkaisurakennetta. Toki on muistettava, että opettajat saivat videoleik-

keet valmiina loppuformaateissaan video-DVD:t tehneeltä videotuotantoyhtiöltä, joten kaikkia vaiheita heidän ei tarvinnut itse tehdä.

Sekä vapaasti käytettävissä olevan että kaupallisen videomateriaalin määrä verkossa aiheuttaa valinnan vaikeutta ja vaikka tiedettäisiin, että omaan opetukseen liittyvää videomateriaalia varmasti löytyy verkosta, sen löytäminen on vaikeaa. Tässä multimediasisältöihin liitettävä metatieto ja sen hallinta näyttelevät merkittävää roolia.

Eri oppilaitoksissa niin Suomessa kuin muuallakin on herätty huomaamaan videopohjaisen sisältöjen arvo opetuksen tukena, kun videokuvaus ja -käsittely ovat digitalisoituneet. Nykytekniikalla kuka tahansa opettaja pystyy pienen harjoittelun jälkeen tuottamaan käyttökelpoista videosisältöä hyvin pienillä laiteinvestoinneilla. Ongelmaksi muodostuikin, miten tuotettua videomateriaalia pystytään hallinnoimaan ja yhdistämään muuhun verkkooppimateriaalin. Henkilöt, jotka ovat manuaalisesti linkittäneet HTML-sivuihinsa palvelimelle jollakin tavalla siirtämiään videoleikkeitä, tietävät, että se ei tunnu enää kovin järkevältä toiminnalta kymmenennen tai kahdennenkymmenennen videoleikkeen jälkeen. Lisäksi voi herätä kysymys, miten opettaja itse tai hänen kollegansa voisivat hyödyntää uudelleen jo kertaalleen julkaistuja videoleikkeitä.

Kun hallinnoitavia videoleikkeitä alkaa olla kymmeniä tai satoja, videoleikekirjasto ja käyttäjäautentikoinnin sisältävä julkaisujärjestelmä ovat järkeviä investointeja koulutusyksikkökohtaisissa ratkaisuissa. MetSimu-projektissa opettajilta saadun palautteen perusteella leikekirjastot ja selainkäyttöliittymän kautta tapahtuva eri medialeikkeiden yhdistäminen toimivat hyvin tilanteissa, joissa tuotetaan rakenteeltaan suhteellisen yksinkertaisia ja määrämuodossa olevia multimediasisältöjä.

Videoneuvottelun käytön yleistymisen tuo myös oman lisänsä videotallenteiden käyttöön. Videoneuvotteluna toteutetut luennot, seminaarit tai muut opetustilanteet pystytään tallentamaan reaaliaikaisesti ja lähes automaattisesti mediapalvelimelle, josta ne ovat välittömästi käytettävissä tilaisuuden jälkeen. Tallenteita käytetään tällä hetkellä korvaamaan reaaliaikaista osallistumista ja kertaamistarkoituksiin, mutta mikäli tekijänoikeudet sen sallivat, videoneuvottelutallenteiden osia voitaisiin hyödyntää täysin irrallaan alkuperäisestä kontekstistaan.

Haasteet videopohjaisen oppi- ja muun materiaalin tuottamisessa liittyvät prosesseihin videokameran ja valmiin videoleikkeen julkaisemisen välissä. Luvussa 2 esitettyyn TV-tuotantomaiseen prosessiin on hyvin harvoin varaa oppimateriaalituotannossa, varsinkaan silloin, kun opettaja tekee ehkä kertaluonteista tukimateriaalia omalle kurssilleen. Ihanne-tilanteessa koko sisällöntuotantoprosessi viedään läpi viiden minuutin valmistautumisajalla, kuvauksella ja minuutin jälkikäsitteilyllä: puhe- ja neuvottelutaidon opettaja kytkee virrat tietokoneeseen ja siihen kytkettyyn DV-kameraan, sijoittaa kameras ja mikrofonin oikeisiin paikkoihin, säätää valaistuksen ja antaa neuvotteluharjoitukseen osallistuville opiskelijoille luvan aloittaa. Harjoitus kuvataan ja tallennetaan suoraan tietokoneelle loppuformaattiin. Harjoituksen päätyttyä opettaja siirtää videoleikkeen mediapalvelimelle, kirjautuu videoleikkeiden hallinta- ja julkaisujärjestelmään ja julkaisee hetki sitten tehdyn tallenteen kurssilaisille. Harjoitukseen osallistuneet ja muut kurssilaiset voivat katsella tallenteen ja kommentoida sitä välittömästi harjoituksen jälkeen opettajan antaman tehtävänannon mukaan.

Edellä kuvattu skenaario ei toteudu vielä käytännössä, mutta siihen suuntaan on syytä pyrkiä. Teknisesti vastaava järjestelmä on mahdollista rakentaa räätälöitynä tiettyyn käyttötapaukseen, mutta yleiskäyttöisten järjestelmien toteuttaminen vaatii yhteensopivuutta ja -toimivuutta tuotanto-ohjelmistojen ja tallennus- ja julkaisujärjestelmien välillä. On hyvin mahdollista, että tämä saavutetaan ensiksi mobiililaitepuolella, esimerkkejä matkapuhelimille tarkoitetuista helppokäyttöisistä kuvien julkaisupalveluista on jo olemassa.

Tässä työssä kuvattu tietokannan päälle rakennettu dynaaminen verkkomultimedias tuottaminen ei korvaa tai tee tarpeettomaksi staattista tiedostopohjaista multimedias tuottamista. Staattinen esitysten koostaminen soveltuu hyvin tilanteisiin, jossa tuotettavan sisällön määrä on pieni, päivittämistä ja muokkausta tehdään harvoin ja multimediaesitykset ovat rakenteeltaan uniikkeja. Ei pidä myöskään unohtaa sitä tosiasiaa, että dynaaminen multimediaesitysten koostaminen vaatii tietoverkkoyhteyden, mikä ei edelleenkään ole itsesäänselvyys. On kuitenkin nähtävissä, että multimediasisältöjen käytön lisääntyminen vie väijäämättä kohti dynaamiseen esitysten koostamiseen perustuvia julkaisujärjestelmiä ja multimediasisällöntuottajien tulee varautua tähän muutokseen.

## Lähdeluettelo

Verkkoviittaukset tarkistettu 31.12.2004.

[Aspen Moviemap 2004] Aspen Moviemap. [online] <URL: <http://www.naimark.net/projects/aspen.html>>

[Backer 1988] David .S. Backer. Structures and Interactivity of Media: A Prototype for the Electronic Book. PhD Dissertation, Massachusetts Institute of Technology, June 1988.

[Burnett 2003] Burnett, Ian *et al.* MPEG-21: Goals and Achievements. IEEE Multimedia October-December 2003. Sivut 60 - 70. Saatavilla sähköisenä <URL: <http://www.chiariglione.org/mpeg/events&tutorials/MMpaper.pdf>>

[Bush 2004] Bush, Michael *et al.* Customized Video Playback: Standards for Content Description, Customization and Personalization. Educational Technology July-August 2004. Sivut 5 – 13.

[Clarke 2001] Clarke, Alan. Designing Computer-Based Learning Materials. Gower Publishing Limited 2001.

[CSC Metadata 2003] Medar XML schemas [online] <URL: <http://tv.funet.fi/media-arkisto/tekniikka/metadata-schemas.jsp.en>>

[Digibarn 2004] Digibarn Computer Museum. Bruce Damer's Personal Histories of the Desktop User Interface. [online] <URL: <http://www.digibarn.com/stories/desktop-history/index.html>>

[Digit 2004] Digit-verkkolehti. MPEG-4 Part 10 - Apple flies the standard. Kesäkuu 2004. [online] <URL: <http://www.digitmag.co.uk/news/index.cfm?fuseaction=displaynews&NewsID=4151>>

---

[The Dot Eaters 2004] The Dot Eaters - videogame history [online] <URL: <http://www.emuunlim.com/doteaters/>>

[Donelson 1978] Spatial management of information. Proceedings of the 5th annual conference on Computer graphics and interactive techniques. Sivut 203 - 209. [online] <URL: <http://doi.acm.org/10.1145/800248.807391>>

[DSpace Metadata 2004] Metadata of DSPACE at MIT [online] <URL: <http://dspace.org/technology/metadata.html>>

[Dublin Core 2004] Dublin Core –metatietostandardi [online] <URL: <http://www.dublincore.org>>

[Häkkinen et al. 1997] Häkkinen, P. & Suomela, K. On the Development of the Hypermedia Courseware on the Web. IFIP WG 3.3 Working Conference, Sozopol, Bulgaria, May 27th-27th 1997, Human Computer Interaction and Educational Tools, 215-221.

[Häkkinen 2004a] Häkkinen, Pasi. SMIL-kielen hyödyntäminen verkko-oppimateriaalin tuotannossa. Verkko-pedagogiikka ja -teknologia seminaari 11.5.2004, Otaniemi, Espoo. <URL: [http://www.csc.fi/suomi/koulutus/videopedagogiikka\\_ja\\_teknologia\\_seminaari.phtml.fi](http://www.csc.fi/suomi/koulutus/videopedagogiikka_ja_teknologia_seminaari.phtml.fi) >

[Häkkinen 2004b] Häkkinen, Pasi. SIMUBA - Internet based learning environment supporting simulator based education. The International Seminar on Simulator-Based Training of Forest Machine Operators, 17.-19.11.2004, Joensuu. <URL: <http://simumedia.pkky.fi>>

[IBM 2004] IBM VideoAnnEx -annotointityökalu [online] <URL: <http://www.research.ibm.com/VideoAnnEx/>>

[Learning SMIL 2004] Learning SMIL Tutorial, Boston University [online] <URL: <http://www.bu.edu/webcentral/learning/smil1/>>

[LOM 2004] Learning Object Metadata [online] <URL: <http://ltsc.ieee.org/wg12/>>

[Lukkari 2004] Lukkari, Ulla. Digitaalisen sisältötuotantoprojektin hallinta. IT Press 2004, Helsinki.

[Luukkonen 2000] Luukkonen, Jussi. Digitaalisen median käsikirjoitusopas. Edita 2000, Helsinki.

[Macromedia 2004] Macromedia Flash ja Director -sovellukset verkkomultimedian tuottamiseen. [online] <URL: <http://www.macromedia.com>>

[MAGpie 2004] Media Access Generator. Sovellus tekstitysten ja auditiivisten kuvailujen lisäämiseen multimediaesityksiin [online] <URL: <http://ncam.wgbh.org/webaccess/magpie/>>

[Mallinnus 2004] Matemaattisen mallinnuksen verkostohanke [online] <URL: <http://alpha.cc.tut.fi/mallinnus/>>

[MARC 2004] MARC Standards [online] <URL: <http://www.loc.gov/marc/>>

[McCarthy et al. 2004] McCarthy, J. Sasse, A., Miras, D. Sharp or Smooth? Comparing the effects of quantization vs. frame rate for streamed video. [online] <URL: [http://www.cs.ucl.ac.uk/research/higherview/mccarthy\\_video\\_quality.pdf](http://www.cs.ucl.ac.uk/research/higherview/mccarthy_video_quality.pdf)>

[Medar 2004] Funet-TV:n media-arkiston Medar-metatietokanta [online] <URL: <http://tv.funet.fi/medar/>>

[Metakka 2004] Metakka LOM-metatiedon pilotointiympäristö. TIEKE. [online] <URL: <http://www.tieke.fi/metakka/lom.nsf>>

[MetSimu 2004] Metsäkonesimulaatio-opetuksen tuotteistaminen -projekti [online] <URL: <http://simumedia.pkky.fi>>

---

[Microsoft 2004a] Reducing Broadcast Delay [online] <URL: <http://www.microsoft.com/windows/windowsmedia/howto/articles/BroadcastDelay.aspx>>

[Microsoft 2004b] Microsoft Producer. [online] <URL: <http://www.microsoft.com/windows/windowsmedia/technologies/producer.aspx>>

[MPEG 2004] Moving Pictures Expert Group [online] <URL: <http://www.chiariglione.org/mpeg/>>

[MySQL 2004] MySQL tietokanta [online] <URL: <http://www.mysql.com/>>

[Opinportti 2004] YLE Opinportti [online] <URL: <http://www.yle.fi/opinportti/>>

[PerlySMIL 2003] PerlySMIL [online] <URL: <http://www.webiphany.com/perlysmil/>>

[PHP 2004] PHP: Hypertext Preprocessor [online] <URL: <http://www.php.net/>>

[Pohjolainen et al. 2003] Pohjolainen S., Suomela K. & Häkkinen, P. Mathematical Modelling over the Internet – A National Network Project of Finnish Virtual university. PEG 2003, The Eleventh International PEG Conference, 28 June – 1 July in St. Petersburg, Russia.

[Quicktime 2004] Quicktime [online] <URL: <http://www.apple.com/quicktime/>>

[Ranta 2005] Ranta, Pekka. Metsäkonesimulaatio-opetuksen tuotteistaminen projekti. ESR-hankkeen loppuraportti. Ilmestyy tammikuussa 2005. Itä-Suomen lääninhallitus. ESRA-seurantajärjestelmä.

[Real 2004a] Real Networks [online] <URL: <http://www.real.com>>

[Real 2004b] Real Documentation Library: Content Production and Authoring [online] <URL: <http://service.real.com/help/library/encoders.html>>



---

[Real 2004c] Real Networks Tools & Plug-ins: RealPix Bandwidth Calculator [online] <URL: <http://www.realnetworks.com/resources/toolsplugins/>>

[Redant 2001] Redant, John. The Contributions and the Downfall of the Xerox Star. [online] <URL: <http://xeroxstar.tripod.com/>>

[Rinta-Filppula et al. 2002] Rinta-Filppula, R., Koskenvaara, T. Web University - Kansainvälinen etäopiskelu ja -työympäristö. Teoksessa: Hautakangas, S. & Pohjolainen, S. (toim.). Avoin Oppimisympäristö (AO)-hankkeen loppuraportti, 1, 123-143. Saatavilla sähköisesti <URL: <http://matriisi.ee.tut.fi/ao/AO-loppuraportti.pdf>>

[rmedit 2004] Editing RealVideo Files [online] <URL: <http://service.real.com/help/videoccg/editing.html>>

[Salminen 2004] Salminen, Harri. Report on video-on-demand metadata and portals. TF-netcast -työryhmän raportti. [online] <URL: <http://tv.funet.fi/funet-tv/projekteja/tf-netcast/tf-netcast-deliverable-g.jsp.en>>

[Sariola et al. 2003] Sariola, Janne (toim.) Videoteknologian käyttö yliopistoissa 2003-2006. Suomen virtuaaliyliopiston palveluhankkeiden määrittelyraportti. [online] <URL: <http://www.virtuaaliyliopisto.fi/e-julkaisut/julkaisu006.pdf>>

[Sonera 2004] Sonera Mediolab. RealMedian ja Windows Median vertailua eri videosisäilyillä, koodekeilla ja koodekkiversioilla. [online] <URL: <http://www.medialab.sonera.fi/demos/>>

[Stenvall 1998] Stenvall, Jani, Hakala, Juha. Dublin Core -formaatin käyttöopas, 21.09.1998 [online] <URL: <http://www.lib.helsinki.fi/meta/dc-opas.html>>

[StreaminMedia 2004] StreamingMedia-verkkolehti. H.264 Video Prepares MPEG-4 For Prime Time. Maaliskuu 2004. <URL: <http://www.streamingmedia.com/article.asp?id=8588>>

[SVG 2004] W3C:n suositus vektorikuvaformaattista verkkokäyttöön [online] <URL: <http://www.w3.org/Graphics/SVG/>>

[Sync-O-Matic 2004] Sync-O-Matic -ohjelma videoleikkeiden ja luentokalvojen synkronoimiseen. [online] <URL: <http://www.syncomat.com/>>

[Timed-Text 2004] W3C:n suositus ajastettujen tekstien käyttöön WWW-sovelluksissa [online] <URL: <http://www.w3.org/AudioVideo/TT/>>

[W3C 2004] W3C Synchronized Multimedia Integration Language [online] <URL: <http://www.w3c.org/AudioVideo/>>

[Web University 2004] Web University [online] <URL: <http://www.cern.ch/webuniversity/>>

[Videnet 2001] Dublin Core Application Profile for Digital Video [online] <URL: [http://www.vide.net/workgroups/videoaccess/resources/vide\\_dc\\_userguide\\_20010909.pdf](http://www.vide.net/workgroups/videoaccess/resources/vide_dc_userguide_20010909.pdf)>

[Windows Media 2004] Windows Media [online] <URL: <http://www.microsoft.com/windows/windowsmedia/>>

[Wired News 1999] RealNetworks Probe Begins. Tarkistettu 23.8.2003 [online] <URL: <http://www.wired.com/news/technology/0,1282,32250,00.html>>

[XHTML+SMIL 2004] W3C:n XHTML+SMIL -profiili. [online] <URL: <http://www.w3.org/TR/XHTMLplusSMIL/>>

[X-Smiles 2004] X-Smiles -soitto-ohjelma [online] <URL: <http://www.xsmiles.org>>

# Liitteet

## Liite 1: SMIL-ohjelmakoodiesimerkki

Seuraava SMIL-merkkaukielen esimerkkilistaus on otettu SIMUBAn yhdestä toimintamallista, johon ei ole lisätty ajastettuna kommentteja. Listauksen layout-osassa soitto-ohjelman soittoalue jaetaan kahteen osaan, joista toiseen (**Title\_Region**) tuodaan body-osassa määritelty tekstimuotoinen RealText-leike ja toiseen (**Video\_Region**) sijoitetaan videoleike. Koska teksti- ja videoleikkeet ovat par-lohkon sisällä, soitto-ohjelma näyttää videon otsikkotiedot sisältävän tekstileikkeen ja videoleikkeen yhtä aikaa.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<smil xmlns="http://www.w3.org/2001/SMIL20/Language">
  <head>
    <meta name="author" content=""/>
    <meta name="title" content="3.2. Siirtoauton tarkistukset"/>
    <meta name="copyright" content=""/>
    <layout>
      <root-layout background-color="#279957" width="368" height="368"/>
      <region id="Title_Region" left="4" top="4" width="360" height="80" z-index="3"/>
      <region id="Video_Region" left="4" top="84" width="360" height="280" z-
index="3"/>

      </layout>
    </head>
    <body>
      <par>
        <video id="video"
src="http://simuba.pkky.fi/simuba/video/taskvideos/pekka.ranta/p32_1079605225.rm" re-
gion="Video_Region" dur="131"/>
        <textstream id="title"
src="http://simuba.pkky.fi/simuba/smil/modelcomment_rt.php?id=339&title=1" re-
gion="Title_Region" dur="131"/>

      </par>
    </body>
  </smil>
```

Seuraavassa listauksessa on esitetty edellä mainitun dynaamisesti PHP-ohjelmalla generoitavan RealText-tiedoston sisältö. Listauksen alussa määritellään tekstileikkeen perusominaisuudet: ajallinen kesto, tekstileikkeen varaaman kuva-alueen koko ja värimääritykset. Loppuosassa on varsinainen tekstileikkeen sisältö, jolle on määritelty ulkoasumäärityksiä.

```
<window
  type="generic"
  bgcolor="#FFFFFF"
  underline_hyperlinks="false"
  width="360"
  height="80"
  duration="131"
  link="#666600"
  version="1.2"
>

<font face="Arial" size="+0" charset="iso-8859-1">
<ul>
Toimintamalli:
<b>3.2. Siirtoauton tarkistukset</b><br/>
Konemerkki:Ponsse<br/>
Maasto:Piha
</ul>
</font>
</window>
```

## **Liite 2: SIMUBAn palvelinalusta ja ohjelmaversiot**

### **Palvelinkone:**

- IBM xSeries 226
- prosessori Intel Xeon 2.66GHz
- muistia 1Gtavu
- levyjärjestelmä 2 x 36 Gtavu SCSI-levyt RAID-1 konfiguraatiossa

### **Ohjelmistot:**

Käyttöjärjestelmä: Linux Redhat 9

WWW-palvelin: Apache 2.0.50 (toimii myös Apache 1.3.x-versioilla)

Tietokanta: MySQL 4.0.20 (toimii myös MySQL 3.x-versioilla)

PHP: PHP 5.0.2 (toimii myös vanhemmilla PHP 4.x-versioilla)

RealPlayer: kaikki RealOnePlayer-versiot sekä uudemmat RealPlayer 10 -versiot

### **Selaimet:**

- Internet Explorer 6 (kaikki toiminnallisuudet)
- Mozilla 1.7, Firefox 1.0 (kommentointilomakkeen JavaScript-rajapinta RealPlayer-pluginin kanssa ei toimi täysin)

### **SMIL-esitykset :**

- SMIL 2.0 (tosin ei käytetä sen uusia ominaisuuksia verrattuna 1.0-versioon)
- RealText 1.2 (kaikki tekstielementit SMIL-esityksissä)
- RealMedia:
  - videokodekki RealVideo 9
  - bittinopeus 512kbit/s
  - kuvakoko 360x280
  - kuvanopeus 25 fps
  - (myös muilla parametreilla tuotettuja leikkeitä käytössä)

## **Liite 3: Matemaattisen mallinnuksen luentotallenteiden palvelinalusta ja ohjelmaversiot**

### **Palvelinkone (CSC):**

- Supermicron SuperServer 6022P-8R
- prosessorit 2 x Intel Xeon
- muistia useita gigatavuja
- erillinen teratavu-kokoluokan levyjärjestelmä

### **Ohjelmistot:**

Käyttöjärjestelmä: Linux Redhat 9

Mediapalvelinohjelmisto: Helix Universal Server

RealPlayer: kaikki RealOnePlayer-versiot sekä uudemmat RealPlayer 10 -versiot, tallenteet toimivat myös vanhemmissa RealPlayer 8 -versioissa

### **Selaimet:**

- graafinen selain, joka pystyy käynnistämään RealPlayerin SMIL-esityksen linkistä, mm. Internet Explorer 5.x tai uudempi, Netscape 4.x tai uudempi, Opera, Mozilla

### **SMIL-esitykset :**

- SMIL 1.0
- RealText 1.2 (kaikki tekstielementit SMIL-esityksissä)
- RealMedia:
  - videokoodekki RealVideo 8
  - videotiedostoformaattina SureStream, sisältää bittinopeusversioita välillä 64 - 512 kbit/s
  - videoleikkeiden kuvakoko välillä 192 x 144 - 320x240
  - kuvanopeus 2 - 25 fps riippuen yhteysnopeudesta
- luentokalvot RealPix-muodossa GIF-kuvina, kuvakoko tapauskohtainen
- tekstimuotoisten tiedostojen luomiseen käytettiin itse tehtyä Perl-ohjelmaa