



TAMPEREEN  
AMMATTIKORKEAKOULU

TUTKINTOTYÖRAPORTTI

**OPTIMAALISEN CD-ROM-ESITYKSEN  
SUUNNITTELU JA TOTEUTUS**

**Satu Rusi**

Tietojenkäsittelyn koulutusohjelma  
marraskuu 2005  
Työn ohjaaja: Petri Heliniemi

TAMPERE 2005



---

<b>Tekijä(t)</b>	Satu Rusi	
<b>Koulutusohjelma(t)</b>	Tietojenkäsittely	
<b>Tutkintotyön nimi</b>	Optimaalisen cd-rom-esityksen suunnittelu ja toteutus	
<b>Työn valmistumis- kuukausi ja -vuosi</b>	marraskuu 2005	
<b>Työn ohjaaja</b>	Petri Heliniemi	<b>Sivumäärä: 37</b>

---

## TIIVISTELMÄ

Tutkintotyössä on perehdytty optimaalisesti toimivan cd-rom-multimediaesityksen suunnitteluun ja toteutukseen. Työssä käsitellään Macromedia Director MX -ohjelmalla luotavia cd-rom-esityksiä. Tutkintotyö esittelee keinoja erityisesti isokokoisten ja tietokoneen prosessorilta suurta tehoa vaativien cd-rom-esitysten optimointiin. Myös kevyemmät esitykset hyötyvät esimerkiksi mediaelementtien optimoinnista. Tutkintotyö auttaa hahmottamaan, mitkä seikat vaikuttavat cd-romin toimivuuteen ja miten paljon ongelmia voidaan ennaltaehkäistä suunnittelemalla ja toteuttamalla cd-rom-esitys huolellisesti. Valmis, optimoitu cd-rom-esitys toimii moitteettomasti muotoutuen käyttäjän laitteistolle sopivaksi ja on tarvittaessa helposti päivitettävissä.

Cd-rom-esityksen huolellinen suunnittelu ja ongelmakohtien huomioon ottaminen jo alkuvaiheessa säästää aikaa ja vaivaa projektin lopussa. Cd-rom-projekti käynnistyy idean, tavoitteen ja kohderyhmän sekä projektin resursien ja aikataulun suunnittelulla. Esityksen asetukset tulee määrittellä kohdekäyttäjien laitteistoille sopiviksi. Mediaelementeistä erityisesti äänet ja videot vaativat suunnittelua käyttöongelmiensa takia. Myös Directorin Lingo-ohjelmointikielen ominaisuudet, muistin tarkkailu ja esityksen esilataaminen kannattaa huomioida jo cd-romin suunnitteluvaiheessa.

Esityksen huolellinen suunnittelu mahdollistaa sen, että projektin toteutusvaiheessa vältetään suurimmilta ongelmilta. Yksi tärkeimmistä seikoista cd-rom-esityksen optimoinnissa on esityksen multimediaelementtien optimointi. Erityisesti ääni-, video- ja 3D-elementit tulee optimoida, jotta esitys toimisi ongelmitta peruskoneillakin. Mediaelementtien optimoinnin jälkeen voidaan aloittaa esityksen koostaminen. Directorin työympäristön hallinta ja tunteminen nopeuttaa työskentelyä. Esitystä tulee testata koostamisen eri vaiheissa. Testauksessa sekä virheiden jäljittämässä ja korjaamisessa voi käyttää apuna Directorin omia työkaluja sekä Lingo-ohjelmointikieltä. Virheettömäksi todettu cd-rom-esitys voidaan julkaista. Julkaistavaa versiota tehtäessä tulee ottaa huomioon, että esitys käynnistyy nopeasti ja cd-romin tiedostot on suojattu.



---

<b>Author(s)</b>	Satu Rusi	
<b>Degree Programme(s)</b>	Business Information Systems	
<b>Title</b>	The design and implementation of an optimized cd-rom presentation	
<b>Month and year</b>	November 2005	
<b>Supervisor</b>	Petri Heliniemi	<b>Pages: 37</b>

---

### ABSTRACT

This final thesis examines how to design and implement an optimized cd-rom presentation. It is based on presentations created with Macromedia's Director MX -multimedia authoring tool. The thesis presents ways to optimize especially large and complex cd-rom presentations and it helps to outline which facts affect the functioning of the cd-rom. The completed, optimized cd-rom presentation performs faultlessly, adjusts to the user's hardware, and is simple to update.

It is worth while to design the cd-rom presentation carefully because it will save time and trouble at the end of the project. The cd-rom project starts by planning the idea, objective and user group, as well as the project's resources and timetable. The cd-rom presentation must be designed for the average target machine. Sound and video elements require planning due to their operational problems. The design phase also includes recognizing the possibilities of Director's programming language Lingo, memory handling, and preloading the presentation or its elements.

One major way to optimize the cd-rom is to trim the media elements. Especially sound, video and 3D-elements have to be trimmed as much as possible to reduce file size which enables the cd-rom to load and display quickly. After gathering up the media elements, it is possible to start to build the presentation. Mastering the Director interface and its tools speeds up the work Cd-rom presentation must be tested regularly. Director's tools and Lingo help to test the presentation and handle and debug errors. The bug-free cd-rom presentation is ready to be published. The final version of the optimized cd-rom presentation includes protected files and an fast launching projector file.

# Sisällysluettelo

Director-termistöä.....	6
1 Johdanto.....	7
1.1 Tutkintotyön taustaa ja tavoite .....	7
1.2 Aiheen rajausta.....	7
1.3 Termien käyttö ja tutkintotyön lähdemateriaali .....	8
2 Cd-rom-esityksen suunnittelun optimointi .....	9
2.1 Suunnittelun lähtökohdat .....	9
2.1.1 Idea, tavoite ja kohderyhmä.....	9
2.1.2 Resurssit ja aikataulu .....	10
2.2 Multimediaesityksen asetusten määrittely .....	10
2.2.1 Resoluutio .....	10
2.2.2 Värimääritykset .....	11
2.2.3 Fontit.....	11
2.3 Mediaelementtien käytössä huomioitavaa .....	12
2.3.1 Ääni .....	12
2.3.2 Video .....	12
2.4 Lingo-ohjelmointi .....	14
2.4.1 Lingon lauserakenne.....	15
2.4.2 Lingon kirjoittaminen .....	15
2.4.3 Koodin kommentointi.....	16
2.4.4 Koodin nopeuden optimointi .....	16
2.5 Muistin tarkkailu .....	17
2.5.1 Cast memberien muistista poistaminen .....	17
2.5.2 Memory Inspector.....	18
2.5.3 Muistin tarkkailu Lingolla .....	18
2.6 Esilataaminen .....	19
2.6.1 Cast memberien esilataaminen .....	19
2.6.2 Preloader-esilataus.....	20
2.6.3 Splash Screen.....	20

3 Cd-rom-esityksen toteuttamisen optimointi .....	21
3.1 Työympäristön tehokas hallinta .....	21
3.1.1 Näppäinkomennot.....	21
3.1.2 Ohjauspaneeli .....	21
3.1.3 Cast-ikkuna.....	22
3.1.4 Scoren framet.....	23
3.1.5 Muita vinkkejä.....	23
3.2 Mediaelementtien optimointi .....	24
3.2.1 Teksti .....	24
3.2.2 Kuva.....	25
3.2.3 Ääni .....	26
3.2.4 Video .....	26
3.2.5 Flash.....	27
3.2.6 3D-elementti .....	28
3.3 Esityksen testaaminen .....	29
3.4 Virheiden jäljittäminen ja korjaaminen.....	29
3.4.1 Debugger .....	30
3.4.2 Object Inspector.....	30
3.4.3 Message-ikkuna .....	31
3.4.4 Trace-ominaisuus.....	31
3.5 Cd-rom-esityksen julkaisu .....	32
3.5.1 Projektori .....	32
3.5.2 Stub-projektori.....	33
3.5.3 Tiedostojen päivittäminen ja suojaaminen .....	33
3.5.4 Cd-rom tallennusformaattina.....	34
3.5.5 Cd-romin käyttöohjeistus ja jakelu.....	34
4 Yhteenveto.....	36
Lähteet .....	37

## Director-termistöä

Behavior	Directorin Lingo-ohjelmointikielen skriptityyppi. Directorin kirjastopalletti sisältää paljon valmiita behavior-skriptejä ja niitä on mahdollisuus luoda itse lisää.
Cast member	Yksittäinen cast-ikkunassa oleva elementti, joka voi olla esimerkiksi teksti, kuva, ääni tai video.
Codec	Compressor/Decompressor -menetelmä, jota käytetään digitaalisen videon, kuvan ja äänen pakkaamiseen.
DTS	Direct to Stage -ominaisuus, jota voidaan käyttää esimerkiksi video-spriteissa. Ominaisuuden avulla mediaelementin grafiikka näyttää paremmalta ja se voidaan esittää sulavammalla esitysnopeudella.
Frame	Directorissa käytettävä yksikkö, joka kuvaa scoren aikaa sekä spritejen mitta. Yksi frame on yksi pieni ruutu scorella.
Marker	Osoitin, jonka avulla framet voidaan nimetä scoren valkoisessa osoitekentässä.
Näyttämö (Stage)	Näyttämö esittää elementit ja tapahtumat siten, miten ne lopullisessa esityksessä näkyvät.
Projektor (Projector)	Director-esityksestä tehty exe-päätteinen itsekäynnistyvä tiedosto.
Property Inspector	Apuväline kaikessa Director-työskentelyssä. Esittää tietoja avatusta ikkunasta tai valitusta elementistä.
Score	Osa Directorin käyttöliittymää. Esityksen ”digitaalinen käsikirjoitus”. Scorella ohjataan mitä ja milloin näyttämöllä tapahtuu.
Skripti-ikkuna (Script Window)	Skripti-ikkunassa luodaan Lingo-komennot, joiden avulla esitys voidaan ohjelmoida.
Sprite	Yksittäinen scorella oleva elementti.
Xtra	Directoriin liitettävä lisäominaisuus tai toiminto, joka voi olla esimerkiksi parannettu videotuki tai joukko uusia Lingo-komentoja.

# 1 Johdanto

## 1.1 Tutkintotyön taustaa ja tavoite

Tutkintotyön aihe sai alkunsa toimeksiannosta, joka oli Tampereen ammattikorkeakoulun yrittäjyyden yksikön proAkatemian cd-rom-esityksen suunnittelu ja toteutus. Kyseinen cd-rom-esitys oli alun perin tarkoitus liittää osaksi tutkintotyötä. Cd-romilla ei kuitenkaan voitu hyödyntää tarpeeksi tutkintotyössä käsiteltäviä asioita, koska se oli sisällöltään melko suppea. Koska cd-rom ei antanut lisäarvoa tutkintotyölle, se rajattiin tutkintotyön ulkopuolelle.

Cd-rom-esityksen optimointi oli luonteva valinta aiheeksi, sillä toimeksiannoksi saadun cd-romin yksi vaatimus oli päivitettävyyttä. Koska cd-rom-esitys ei ole helposti päivitettävissä, tulee esityksen suunnitteluun ja toteuttamiseen kiinnittää erityistä huomiota. Esityksen päivittäminenkin helpottuu, kun esitys suunnitellaan huolellisesti ja sen rakenne toteutetaan loogisesti. Cd-rom-tuotantoon liittyy paljon muitakin ongelmia, joiden ehkäisyyn tutkintotyö esittelee ratkaisuja.

Tutkintotyön tavoitteena on tutkia, miten cd-rom-esityksestä saadaan optimaalinen. Tässä työssä optimoidulla cd-romilla tarkoitetaan sitä, että valmis cd-rom toimii moitteettomasti ja tehokkaasti Windows-ympäristössä ja tarvittaessa muotoutuu käyttäjän laitteiston vaatimusten mukaisesti. Optimaalisen esityksen mediaelementtien eli esimerkiksi video- ja äänitiedostojen tulee olla mahdollisimman pienikokoisia, mutta kuitenkin tarpeeksi hyvälaatuisia, ettei esityksen näytettävyys kärsi. Optimoitu cd-rom on tarkasti testattu ja löytyneet virheet on jäljitetty ja korjattu. Cd-rom myös on julkaistu siten, että se käynnistyy nopeasti ja sen tiedostot on suojattu.

Cd-romin sisällön lisäksi myös projektin suunnittelu- ja toteutusvaihe voidaan optimoida. Suunnitteluun kannattaa sijoittaa aikaa ja huolellisuutta, mikä palkitaan projektin loppuvaiheessa. Toteutusvaiheen optimointiin liittyy cd-romin tekoon käytettävien ohjelmien tehokas hallinta, mikä säästää myös aikaa ja vaivaa.

## 1.2 Aiheen rajaus

Tutkintotyö on rajattu koskemaan vain Windows-käyttöjärjestelmissä Directorilla luotavia ja esitettäviä cd-rom-esityksiä. Macintosh-tietokoneella Directorin käytössä on pieniä eroavaisuuksia. Cd-rom-esityksen voi rakentaa sekä PC:llä että Macintoshilla toimivaksi hybrid-cd-romiksi. Näistä aiheista löytyy lisätietoja esimerkiksi Pasi Kainulaisen Director MX -kirjasta.

Tutkintotyössä käsitellään Macromedian Director MX -ohjelmaa, joka ilmestyi Suomessa vuoden 2003 alkupuolella. Director on valittu ohjelmaksi, jolla toimeksiannoksi saatu cd-rom rakennettiin ja jota käsitellään tutkintotyössä, koska varsinkin laajemmissa multimediatoteutuksissa Directorin sanotaan olevan alan paras ohjelmisto.

Director on hyvin laaja ja monipuolinen ohjelma. Tutkintotyö keskittyy Directorin osalta cd-rom-esityksen tehokkaaseen rakentamiseen, multimediaelementtien optimointiin, virheiden korjaamiseen ja esityksen julkaisuun. Esimerkiksi Lingo-ohjelmointikieltä ja sen komentoja tutkintotyössä esitellään melko pintapuolisesti aiheen laajuuden takia. Lisäinformaatiota voi etsiä tekstin yhteydessä mainituista lähteistä tai muusta Director-kirjallisuudesta.

### 1.3 Termien käyttö ja tutkintotyön lähdemateriaali

Directorista ei ole olemassa suomenkielistä versiota, joten tutkintotyössä käytetään monista asioista englanninkielisiä termejä. Useimmat Directorin englanninkieliset termit ovat kuitenkin vakiintuneet ohjelman suomalaisten käyttäjien keskuudessa. Termien väkinäinen suomentaminen todennäköisesti vain sekoittaisi lukijaa enemmän. Tutkintotyön alussa selitetään tekstissä useasti esiintyvät Director-termit ja niihin kannattaa kokemattoman Director-käyttäjän tutustua ennen tutkintotyön lukemista.

Kaikki Directorin käyttöön liittyvät komennot, kuten esimerkiksi tietyn työkalun avaaminen näppäinkomennolla tai Directorin yläosan valikosta, on kirjoitettu kursiivilla. Myös Lingo-ohjelmointikielen komennot ja muut termit on merkitty poikkeavasti. Tekstissä esiintyvät Lingo-komennot ja -termit on kirjoitettu kursiivilla ja omalla erillisellä rivillä olevat esimerkkikomennot on kirjoitettu erilaisella fontilla.

Cd-romien optimoinnista ei ole olemassa erityistä kirjallisuutta, artikkeleita tai muuta ohjeistusta. Tutkintotyö on kirjoitettu tarjolla olevan kirjallisuuden sekä omien havaintojen ja kokemusten pohjalta. Tutkintotyön päälähteenä on käytetty Pasi Kainulaisen vuonna 2003 ilmestynyttä Director MX -kirjaa. Kainulaisen teos on kattava ja hyvin Directorin käyttöä havainnollistava. Teos perehtyy myös cd-romin optimointiin liittyviin seikkoihin. Gary Rosenzweigin Using Macromedia Director MX -kirja on vaikeaselkoisempi, joten se on toiminut lähinnä Kainulaisen kirjan tukena. Muina lähteinä on käytetty muun muassa Directorin Help-opasta sekä multimediaan ja sisältötuotantoprojektin hallintaan liittyvää kirjallisuutta. Tutkintotyössä on perehdytty Directorin MX-versioon, joten lähteenä on käytetty pääosin vuoden 2003 Director-kirjallisuutta.



## 2 Cd-rom-esityksen suunnittelun optimointi

### 2.1 Suunnittelun lähtökohdat

Multimediaesityksen huolellisella ennakkosuunnittelulla ja ongelmakoh-tien huomioon ottamisella jo alkuvaiheessa säästyy aikaa ja vaivaa pro-jektin lopussa. Kaupalliset tilaustyöt alkavat suunnittelulla työn tilaajan kanssa. Tilaajalla on yleensä idea, jonka toteuttamiseen hän tarvitsee me-dia-ammattilaisen apua.

Multimediaesityksen tilaajalla ei välttämättä ole minkäänlaista käsitystä, miten hänen tilaamansa tuote teknisesti toteutetaan ja millaisia resursseja suunnittelu ja tuotanto vaatii. Tähän seikkaan on kiinnittänyt huomiota myös Kainulainen (2003: 429), joka toteaa, että media-ammattilaisten on tällöin tarjottava omia näkemyksiään siitä, miten esitetty idea käännetään multimediasi ja millaisessa muodossa ja millaisella laitteistolla sitä kan-nattaa esittää.

Kun projektin lähtökohdista on sovittu, luodaan sisältösuunnitelma, jonka pohjalta suunnitellaan esityksen rakenne. Rakenteen suunnittelun ja esi-tystavan valinnan jälkeen suunnitellaan esityksen graafinen ulkoasu. Tä-män jälkeen aloitetaan multimediaelementtien luominen ja esityksen koostaminen. Esitystä testataan tuotantovaiheen edetessä ja esityksen val-mistuttua. Valmis cd-rom-esitys viimeistellään julkaisukelpoiseksi ja toi-mitetaan jakeluun.

#### 2.1.1 Idea, tavoite ja kohderyhmä

Kuten kaikki sisältötuotanto, cd-rom-multimediankin tekeminen lähtee liikkeelle ideasta tai tarpeesta. Idean synnyttyä kootaan lopullinen projek-tiryhmä, jonka kesken määritellään projektille selkeät tavoitteet ja sovi-taan työnjako. Ideaa ei kannata lähteä työstämään ensimmäisten ajatusten pohjalta, vaan alkusuunnitteluun kannattaa varata riittävästi aikaa, jolloin idea ehtii hioutua toimivaksi kokonaisuudeksi. Projektin ensimmäinen kirjallinen dokumentti, synopsis, syntyy tästä ennakkosuunnittelusta.

Projektin alkuvaiheessa tulee määritellä ideoidun tuotoksen, esimerkiksi esittely-cd-romin kohderyhmä. Kohderyhmän rajaaminen mahdollistaa sen, että esityksen sisältö voidaan tarkentaa juuri oikealle kohdeyleisölle, eikä mitään ylimääräistä tarvitse suunnitella ja toteuttaa. Kohderyhmän määrittely auttaa myös luomaan cd-romille sopivat laitteistovaatimukset, sillä eri käyttäjäryhmillä voi olla hyvinkin suuria eroja tietokonelaitteis-toissa.

### 2.1.2 Resurssit ja aikataulu

Kaupallisten sisältötuotantoprojektien toteuttaminen vaatii resursseja eli työntekijöitä, ohjelmia ja laitteistoja sekä aikaa ja rahaa. Ennen projektin perustamista tulee selvittää, mitä resursseja tulevan projektin toteuttaminen vaatii, mitä resursseja on jo olemassa ja mitä tarvitaan lisää sekä onko tarvittaessa varaa lisäresursseihin. Aikaisemmista projektikokemuksista on hyötyä aikataulun ja resurssien arvioinnissa. Projektiin osallistuvien työntekijöiden on hyvä olla mukana aikatauluneuvotteluissa, sillä he osaavat parhaiten määritellä oman työvaiheensa vaatiman ajan ja tarvitsemansa resurssit.

Ulla Lukkari (2004: 51) toteaa, että digitaalisen tuotantoprojektin budjetointi tarkoittaa enemmänkin projektiin kuluvan ajan arvioimista kuin rahojen laskemista. Hän jatkaa, että sekä tuotantoryhmän että asiakkaan tulisi tietää toistensa työprosessit ja aikataulut, jolloin tuotantoaikatauluun saadaan realistinen pohja ja vältetään ikäviltä yllätyksiltä työmääräarvioiden venyessä.

Projekti kannattaa jakaa pienempiin kokonaisuuksiin, jolloin resurssien arviointi ja projektinhallinta helpottuu. Aikataulussa on myös huomioitava työtehtävien väliset riippuvuudet, sillä tiettyjen vaiheiden on valmistuttava ennen seuraavan vaiheen käynnistymistä. Aikataulussa on myös otettava huomioon ne vaiheet, jotka vaativat työn tilaajan hyväksynnän. Projektiin kannattaa varata aikaa vähintään 30 prosenttia yli arvioidun työmäärän, sillä tuotanto voi viivästyä monenlaisten ongelmien takia. (Keränen, Lamberg & Penttinen 2005: 38.)

## 2.2 Multimediaesityksen asetusten määrittelyminen

Optimoidun multimediaesityksen asetukset on kohdistettu loppukäyttäjien tietokonelaitteistoille sopiviksi. Vaikka kohderyhmä olisikin tarkasti rajattu, voi yksittäisten käyttäjien koneiden tehokkuudessa olla silti suuria eroja. Multimediaesitystä ei tarvitse rajoittaa vain keskivertolaitteille, vaan raskaammassa ja näyttävämmässä esityksessä kannattaa hyödyntää Directorin Lingo-komentoja.

Lingo-ohjelmointikielen avulla esityksen resoluutio tai näyttötila voidaan muuttaa automaattisesti riippuen käyttäjän käyttöjärjestelmästä ja ohjelmista. Erilaisten käyttöjärjestelmien aiheuttamiin fonttiongelmiiin voi vaurautua ennalta, joten esityksessä voi käyttää harvinaisempiakin fontteja.

### 2.2.1 Resoluutio

Kainulainen (2003: 432) toteaa, että vaikka cd-rom-esityksissä suositeltava näyttötarkkuus on pitkään ollut 800 x 600 pikselin resoluutio, tietokoneiden nykyiset näytönohjaimet pystyvät paljon tarkempaan suorituk-

seen, joten esityksen kooksi voi asettaa esimerkiksi 1024 x 768 pikseliä. Hän kuitenkin muistuttaa, että jos käyttäjällä on heikompi resoluutio koneessaan, jää osa esityksestä ruudun ulkopuolelle. Cd-rom-esityksen resoluution voi kuitenkin muuttaa loppukäyttäjän tietokoneelle sopivaksi, mutta tämä täytyy tehdä käsin Directorissa Lingo-ohjelmointia käyttäen.

Loppukäyttäjän resoluutio on mahdollista tarkastaa esityksen alussa *the deskTopRectList* -komennolla. Komentoon voidaan liittää ilmoitus, joka näytetään käyttäjälle, jos hänellä on puutteellinen resoluutio. Edellinen komento voidaan liittää projektoriin yhdessä *stage-*, *rect-* ja *drawRect-* komentojen kanssa, mikäli näyttämön koko halutaan muuttaa käyttäjälle sopivaksi ennen esityksen käynnistymistä. (Kainulainen 2003: 433.)

### 2.2.2 Värimääritykset

Värejä voi muokata Directorissa työkalupaneelin tai Property Inspectorin kautta. Property Inspector -ikkuna on kätevä apuväline, joka antaa tarpeellista tietoa Director-esityksen perusasetuksista ja yksittäisten mediaelementtien ominaisuuksista. Mikäli ikkuna ei ole näkyvillä, sen saa esille *Window*-valikosta tai näppäinyhdistelmällä *Ctrl+Alt+S*.

Tarvittaessa cast-membereille eli yksittäisille cast-ikkunassa oleville elementeille voi asettaa omia väripaletteja. Palettikanavan avulla voi koko esitykselle vaihtaa uuden väripaletin. Normaalikäytössä väripaletteihin ei tarvitse tehdä muutoksia, mutta paletteja saattaa joutua vaihtamaan, mikäli cd-rom-esitystä katsotaan vanhemmissa käyttöjärjestelmissä ja tietokoneissa (Kainulainen 2003: 106).

Lingon *the colorDepth* -komennolla voidaan tarkastaa loppukäyttäjän tietokoneen väriasetukset ja ilmoittaa tarvittaessa käyttäjälle, ettei näyttötila ole riittävä. On kuitenkin suositeltavaa päästää käyttäjä katsomaan esitystä, vaikkei hänen tietokoneensa suorituskyky vastaakaan laitevaatimuksia. (Kainulainen 2003: 436.)

### 2.2.3 Fontit

Mikäli cd-rom-esityksessä halutaan käyttää harvinaisia fontteja, tulee ne liittää mukaan esitykseen. Erikoisfontteja varten kannattaa luoda oma cast-ikkuna. Jos esitys on koottu erillisistä Director-tiedostoista, on hyödyllistä tehdä cast-tiedostosta ulkoinen, jolloin jokainen DIR-tiedosto voi viitata yhteiseen fonttitiedostoon. (Kainulainen 2003: 437.)

Erilaiset käyttöjärjestelmät voivat aiheuttaa ongelmia fonttien esittämisessä (Kainulainen 2003: 438). Tämä voidaan välttää muuttamalla tekstit kuviksi, jolloin teksti näyttää aina samalta käyttöjärjestelmästä huolimatta. Tekstin voi muuttaa kuvaksi esimerkiksi valitsemalla editointitilassa

*Modify*-valikosta *Convert to bitmap*. Tulee kuitenkin muistaa, että kuvaksi muutettuja tekstejä on huomattavasti hankalampi päivittää.

## 2.3 Mediaelementtien käytössä huomioitavaa

### 2.3.1 Ääni

Ääniä voidaan käyttää esityksissä monella tavalla, kuten esimerkiksi taustamusiikkina, puheena tai erikoisefekteinä. Äänet voidaan tuoda Directoriin valitsemalla *File*-valikosta *Import* tai linkittämällä ulkoisina tai sisäisinä äänitiedostoina. Sisäiset äänet, jotka Director lataa muistiin ennen äänen soittamista, sopivat lyhyisiin efekteihin ja ulkoiset äänet ovat yleensä isokokoisia ulkosten cast-tiedostojen sisään tallennettuja äänitiedostoja (Kainulainen 2003: 131-132).

Directorin tukema yleisin ääniformaatti on Windows-ympäristössä Wave (WAV). Muita käyttökelpoisia ääniformaatteja ovat RealAudio, Shockwave, Audio, Beatnik, QuickTime ja MP3. WAV-äänien laatu on hyvä ja tiedostokoko suuri, joten se sopii parhaiten käytettäväksi lyhyissä ääniefekteissä. MP3-ääniformaattia kannattaa käyttää pitkissä musiikkikappaleissa. MP3-musiikki on pakattu pieneen tilaan, mutta äänen laatu on silti kohtuullisen hyvä. (Kainulainen 2003: 128-129.)

Kainulainen (2003: 438) toteaa, että varsinkin vanhemmat Windows-käyttöjärjestelmät saattavat olla ongelmallisia, sillä ne osaavat soittaa vain yhdentyypisiä ääniä kerrallaan ilman viiveitä tai muita ongelmia. Hän jatkaa, että Directorin 8-versiosta lähtien on ollut mahdollista ”mik-sata” erityyppisiä ääniä samanaikaisesti ilman viiveitä. Esimerkiksi scoren ääni- tai sprite-kanavien ääniä voi käyttää yhtä aikaa Lingon *puppetSound*- ja *sound playFile* -komennoilla soitetujen äänien kanssa.

Tietyissä tilanteissa voi olla pakko käyttää Lingoa äänen ohjaamisessa. Tällöin voidaan käyttää esimerkiksi juuri *puppetSound*- ja *sound playFile* -komentoja. *puppetSound*-komennolla soitetään esityksen sisäisiä ääniä ja *sound playFile* -komennolla ulkoisia äänitiedostoja. (Kainulainen 2003: 297-298.)

### 2.3.2 Video

Vaikka videot ovat useimmiten kiinteä osa multimediatuotantoa, tulee niidenkin käyttöä harkita, sillä monesti valokuvien avulla voidaan esittää sama asia. Videon tuotanto vie usein runsaasti aikaa ja resursseja, joten videomateriaalin käyttö vaatii suunnittelua.

Keränen ym. (2005: 227) antavat seuraavia vinkkejä videoleikkeiden käyttöön multimediassa:

- Hyvä video on havainnollinen.
- Liikkumaton kamera ja lähikuvat soveltuvat parhaiten esitettäväksi pieneen ruutukokoon.
- Videoleikkeen pituus kannattaa minimoida, sillä jo alle minuutissa voidaan kertoa paljon.
- Pidemmän videon voi jakaa osiin esimerkiksi lisäämällä väliin interaktiivisuutta.

#### Videoformaattit

Director MX tukee QuickTime- (MOV) ja Video for Windows- (AVI) videoformaatteja. AVI-tiedostoja voi katsoa esimerkiksi Windows Media Playerillä ja MOV-tiedostoja Applen kehittämällä QuickTime Playerillä. Macintosh-tietokoneella luodun QuickTime-videon käytössä tulee huomioida, ettei videotiedosto välttämättä toimi PC-koneella.

AVI-tiedostot toimivat cd-rom-esityksissä hyvin, jos ne pakataan sopivalla codecilla (Kainulainen 2003: 136). Erikseen asennettavien Xtrojen avulla Director hallitsee myös muita videoformaatteja, kuten esimerkiksi RealMedia- ja MPEG-videotiedostoja. AVI-videotiedostot ovat moninkertaisesti suurempikokoisia kuin esimerkiksi MPEG-tiedostot (Keränen ym. 2005: 229), koska niiden laatu on parempi.

QuickTime-videoita käyttäessä kannattaa huomioida seuraavat seikat. Kainulainen (2003: 143) huomauttaa, että on yleinen virhe olettaa, että loppukäyttäjän tietokoneessa on QuickTime asennettuna. Hän jatkaa, että jokaisen Directorilla koostetun esityksen mukaan tulisi laittaa QuickTime-asennustiedosto sekä automaattinen QuickTime-tunnistin, joka tarvittaessa asentaa ohjelman ennen esityksen käynnistämistä. Tällöin QuickTime tuodaan Directoriin normaalisti *File*-valikon *Import* -komennolla, jolloin se linkittyy cast-ikkunaan. Asennuksen voi myös ohjata verkkoon, mutta tulee huomioida, ettei kaikilla loppukäyttäjillä välttämättä ole verkkoyhteyttä.

#### Videoelementit

Videot tuodaan Directoriin valitsemalla *File*-valikosta *Import*. AVI-muodossa oleva video voidaan avata myös QuickTime-muotoisena, mutta on suositeltavaa valita lopullinen videoformaatti jo tallennusvaiheessa. QuickTimen etuna on omien ohjauspainikkeiden ohjelmoimisen helppous. Käytännössä Director käyttää videomediaelementtejä aina ulkoisesti, vaikka ne luotaisiinkin sisäisiksi elementeiksi. Esitystä cd-romille poltettaessa tulee siis muistaa polttaa myös videotiedostot mukaan. Videotiedostojen fyysistä sijaintia ei kannata jälkikäteen muuttaa, sillä ne pitää aina linkittää uudelleen Directoriin. (Kainulainen 2003: 137.)

Videoelementin asetuksia pääsee tarkkailemaan Property Inspectorissa, kun elementti on aktivoituna. Videoissa on suositeltavaa käyttää Direct to Stage (DTS) -toimintoa esityksen suorituskyvyn parantamiseksi. DTS-

toimintoa ei voi käyttää, jos videota liikutetaan esityksen aikana tai jos videoiden päälle tulee graafisia elementtejä, kuten kehykset. DTS mahdollistaa videoissa paremman kuvanlaadun ja sulavamman esitysnopeuden. (Kainulainen 2003: 141.)

Videosprite kannattaa laittaa yhden framen pituiseksi ja asettaa framen scriptikanavalle pysäytyskoodi Lingolla, jolloin esitystempo ei voi muuttua esityksen aikana (Kainulainen 2003: 143). Videon päättyttyä voidaan tehdä automaattinen siirtyminen tai antaa käyttäjälle mahdollisuus siirtyä esityksessä eteenpäin napsauttamalla jotakin painiketta.

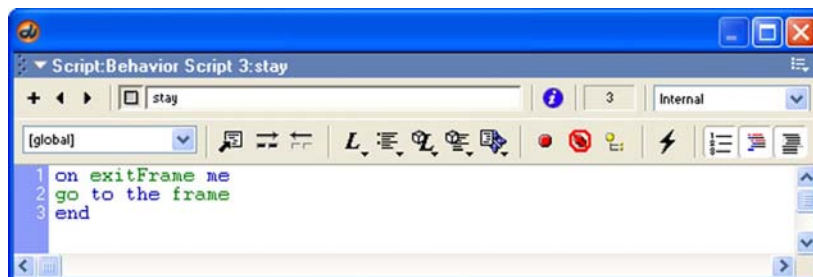
## 2.4 Lingo-ohjelmointi

Directorin perustyökaluilla voidaan rakentaa yksinkertaisia lineaarisia esityksiä, mutta Lingo-komentojen avulla sovelluksista voidaan luoda monimutkaisia interaktiivisia esityksiä, animaatioita tai pelejä. Lingo on samankaltainen olio-ohjelmointikieli kuin esimerkiksi C tai Java, mutta Lingoa ei voi käyttää muissa ohjelmointiympäristöissä kuin Directorissa (Kainulainen 2003: 192). Directorin kirjastopaletti tarjoaa valmiita helpokäyttöisiä Lingo-skriptejä, joita kutsutaan behavior-skripteiksi. Kirjastopaletti löytyy Directorin oikeasta laidasta Code-ikkunan Library-välilehdeltä.

Lingon kirjoittamista voi harjoitella ja testata Message-ikkunassa, jonka saa esiin *Window*-valikosta tai näppäinkomennolla *Ctrl+M* (Kuva 1). Varsinainen ohjelmointi tapahtuu skripti-ikkunassa, jonka saa esille napsauttamalla Directorin ylälaidan *Script Window* -painiketta, valitsemalla sen *Window*-valikosta tai näppäinyhdistelmällä *Ctrl+O* (Kuva 2).



Kuva 1 Message-ikkunassa voi testata Lingo-komentojen toimintaa



Kuva 2 Skripti-ikkunassa tapahtuu esityksen ohjelmointi

### 2.4.1 Lingo lauserakenne

Lingo-komentoja voidaan kirjoittaa sana- tai pistemuotoisena. Sanamuotoinen lauserakenne muistuttaa englannin kieltä ja on siten helpompi oppia. Seuraavan esimerkin komento on lausemuodossa:

```
Set the visible of sprite 1 to FALSE
```

Pistemuotoinen lauserakenne eli pistenotaatio on rakenteeltaan lyhempi ja muistuttaa muita ohjelmointikieliä. Edellinen komentoesimerkki olisi pistenotaationa seuraavanlainen:

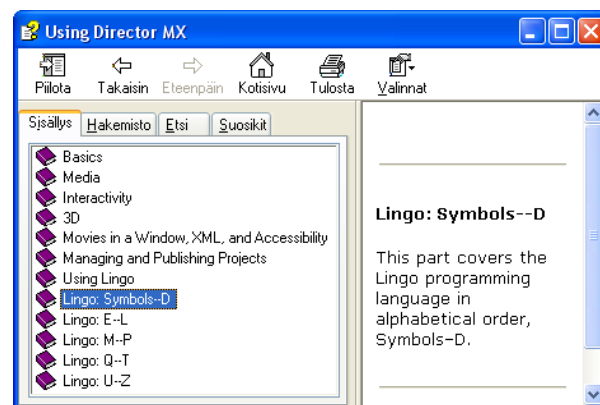
```
Sprite(1).visible=FALSE
```

On suositeltavaa käyttää vain yhtä kirjoitustapaa samaan aikaan samassa esityksessä. Lingo perusteiden oppimisen jälkeen kannattaa siirtyä pistenotaation käyttöön. (Kainulainen 2003: 193.)

### 2.4.2 Lingo kirjoittaminen

Monilta Lingo-ohjelmoinnin ongelmilta voi välttyä koodaamalla Lingoilla hyvän ohjelmointitavan mukaisesti. Lingo kannattaa kirjoittaa pienissä osissa ja jokainen koodin osa kannattaa testata saman tien (Armstrong, Brown, Gowin & Statler 2002: Good scripting habits). Tämä helpottaa ongelmien huomaamista ja paikantamista. Skripti-ikkunan pistemuotoisia esimerkkikomentoja kannattaa käyttää koodaamisen apuna, jolloin komentoja ei tarvitse muistaa ulkoa. Komennot vaativat usein lisäparametreja, jotka kirjoitetaan esimerkkikomennossa olevien sulkeiden sisälle (Kainulainen 2003: 225).

Directorin apuoppaan saa esiin painamalla näppäimistön *F1*-painiketta tai valitsemalla *Help*-valikosta *Director Help*. Avautuvan ikkunan vasemmassa valikosta löytyy paljon hyödyllistä tietoa Lingoista sekä Lingo-ohjelmointikielen sanakirja (Kuva 3).



Kuva 3 Director Help -opas sisältää tietoa muun muassa Lingo-ohjelmoinnista

Sanakirjaan pääsee kätevästi suoraan skripti-ikkunasta pitämällä *Shift*-näppäintä pohjassa ja klikkaamalla haluamaansa komentoa. Tällöin Directorin Help-opas avautuu uuteen ikkunaan ja esittää ohjeita kyseisen komennon käytöstä.

Erilaisten tapahtumankäsittelijöiden, kuten *on MouseUp*, nimeämisessä kannattaa käyttää kuvaavaa nimeä. Käsitteijöiden ja tapahtumaluokkien nimet merkitään yleisen käytännön mukaan siten, että ensimmäinen sana alkaa pienellä kirjaimella ja sitä seuraavat sanat kirjoitetaan jokainen isolla alkukirjaimella (Kainulainen 2003: 383). Käsitteijä voi olla nimeltään esimerkiksi *rightMouseUp* tai *beginSprite* ja tapahtumaluokka *getBehaviorDescription*.

Muuttuja on eräänlainen muistipaikka, jolle ohjelmoija on antanut nimen. Muuttujien nimessä kannattaa käyttää g- tai p-kirjainta. Jos kyseessä on globaali muuttuja, jonka tieto säilyy koko esityksen ajan, voi muuttujan nimi olla esimerkiksi *gNimi*. Lokaalissa muuttujassa, joka voisi olla nimeltään vaikkapa *pOmaMuuttuja*, tieto säilyy vain kyseisen skriptin suorituksen ajan. Usein käytetyt tiedot kannattaa asettaa muuttujien arvoiksi.

### 2.4.3 Koodin kommentointi

Cd-rom-esityksen monimutkaisemmat koodit kannattaa varustaa kommenteilla, varsinkin jos esitystä päivitetään myöhemmin. Tällöin kommentointi auttaa ymmärtämään päivittäjää, miten koodi toimii. Koodin kommentointia edeltää kaksi viivaa (--). Kommentin voi kirjoittaa heti selitettävän komennon perään tai yhdelle tai useammalle omalle riville. Useammalle riville kirjoitetussa kommentissa tulee olla kommentointia edeltävät viivat jokaisen rivin alussa.

Esityksen koodien hyvin runsas kommentointi kasvattaa tiedostojen kokoa. Esitys voidaan tallentaa suojatussa DXR- tai pakatussa DCR-muodossa (Shockwave), jolloin kommentit poistetaan tiedostoista pakkausprosessissa (Armstrong 2002: Lingo syntax, Comments).

Kommentointitoimintoa voi myös hyödyntää Lingo-koodin testauksessa ja virheiden etsimisessä. Koodin voi muuttaa kommentiksi, jolloin virheellistä koodia ei tarvitse poistaa. Kommentointi käy kätevimmin valitsemalla kommentiksi halutun tekstin aktiiviseksi ja napsauttamalla skripti-ikkunan yläosan *Comment*- tai *Uncomment*-painiketta.

### 2.4.4 Koodin nopeuden optimointi

Lingo-ohjelmoinnissa on mahdollista päästä samaan lopputulokseen monella eri tavalla. Yksi tapa saattaa kestää kauemmin ja käyttää enemmän resursseja kuin toinen. Lingo-optimointi ei ole yksinkertaista, mutta eri



vaihtoehtojen testaaminen käy melko helposti. Yksinkertaisissa multimediaesityksissä koodin optimoinnista ei ole juurikaan hyötyä, mutta vaikkapa monimutkaisissa tietokonepeleissä nopeus on hyvin merkittävä tekijä (Kainulainen 2003: 384).

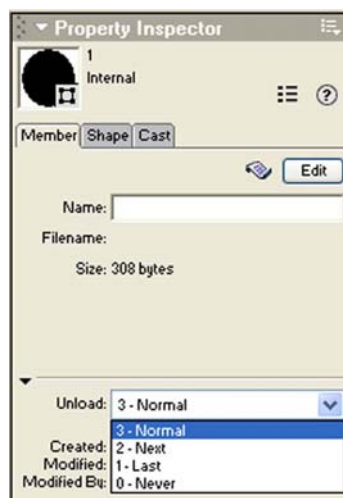
Lingo-ohjelmoinnin ammattilaiset käyttävät benchmark-testausta selvittääkseen nopeimman tavan suorittaa tehtäviä (Rosenzweig 2003: 717). Benchmark-testauksessa vertaillaan eri tapoja parhaan tavan löytämiseksi. Testauksessa käytetyt koodit voivat muuten olla identtisiä, mutta toisessa voidaan käyttää esimerkiksi *if*-komentoa ja toisessa *case*-komentoa. Kun testikoodien nopeusero on pieni, tulee testi suorittaa useita kertoja tarkan keskiarvon saamiseksi.

## 2.5 Muistin tarkkailu

### 2.5.1 Cast memberien muistista poistaminen

Director lataa automaattisesti esityksen jokaisen cast memberin muistiin ennen niiden esittämistä. Kun esimerkiksi jokin tietty kuva ei enää näy esityksessä, poistuu se automaattisesti muistista vapauttaakseen tilaa muille cast membreille. Esityksen suoritusnopeus voi kärsiä jatkuvan muistiin lataamisen ja muistista poistamisen takia. Nämä automaattisesti tapahtuvat toiminnot voi myös säätää käsin, mutta tämä on suositeltavaa vain esityksissä, joissa pienikin nopeushyöty on tarpeen (Kainulainen 2003: 446).

Kullekin cast memberille voidaan erikseen määritellä tapa, jolla se poistuu muistista. Aktivoimalla cast memberin cast-ikkunasta ja siirtymällä Property Inspectorin Member-välilehdelle voi valita tavan, jolla cast member poistuu muistista (Kuva 4).

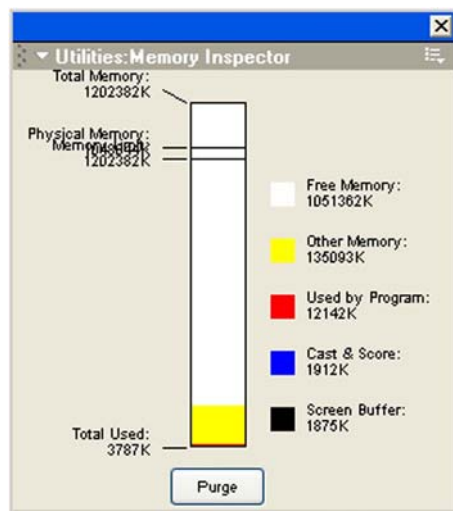


Kuva 4 Property Inspectorissa voi muuttaa cast memberiin liittyviä muistiominaisuuksia

Normal-arvolla varustettu cast member poistuu muistista, kun sitä ei enää tarvita esityksessä ja Director tarvitsee vapaata muistia. Seuraavaksi piimpää pysyvät muistissa Next-arvoiset cast memberit, jotka poistetaan muistista vasta sitten, kun Normal-arvoiset on poistettu. Kauimpaan muistissa pysyvät Last-arvon saaneet cast memberit, jotka poistetaan muistista Next-arvoisten jälkeen. Never-arvolle asetetut cast memberit eivät poistu muistista ollenkaan.

### 2.5.2 Memory Inspector

Directorin Memory Inspector on työkalu, jolla voi tarkkailla esityksen muistia (Kuva 5). Työkalun saa esiin valitsemalla *Window*-valikosta *Memory Inspector*. Memory Inspector näyttää, kuinka paljon muistia on yhteensä, kuinka paljon sitä on vielä vapaana ja miten käytetty muisti on jakautunut. Vapaan muistin loppuessa voi ikkunan *Purge*-painikkeella tyhjentää muistin. Jos Directorin käyttö takkuilee muistin takia pahemmin, kannattaa käynnistää Director uudelleen (Kainulainen 2003: 51).



Kuva 5 Memory Inspector tarkkailee esityksen muistia

### 2.5.3 Muistin tarkkailu Lingolla

Lingon avulla voidaan tarkkailla loppukäyttäjän tietokoneen muistia. Jos halutaan, että myös pienemmällä muistilla varustetuissa tietokoneissa esitys pyörii sulavasti, voidaan esimerkiksi video- tai äänitiedostosta muokata kaksi erikestoista ja -laatuista versiota. Lingon avulla voidaan selvittää, kuinka paljon vapaata muistia on tiedoston käsittelyyn ja esittää sen mukaisesti joko pienempi tai isompi tiedosto.

Vapaan muistin määrä voidaan palauttaa *freeBytes()*-komennolla ja cast memberien koko saadaan selville *size*-parametrin avulla (Kainulainen 2003: 449-450). Ääniä on helppo soittaa yksinkertaisen *puppetSound*-komennon avulla.

Koodi voisi olla seuraavan esimerkin mukainen:

```
if freeBytes() > member ("isoMusiikkitiedosto").size then
  puppetsound "isoMusiikkitiedosto"
else
  puppetsound "pieniMusiikkitiedosto"
end if
```

## 2.6 Esilataaminen

### 2.6.1 Cast memberien esilataaminen

Yksinkertaisen Lingo-komennon avulla Director voidaan pakottaa lataamaan haluttu cast member muistiin etukäteen. Kun sitä myöhemmin tarvitaan, voidaan se esittää suoraan muistista. Esilataamisesta on hyötyä tapauksissa, jossa esimerkiksi isokokoisella scorella olevan kuvan tulisi vaihtua toiseen cast member -kuvaan kesken esityksen (Kainulainen 2003: 450). Jos uusi isokokoinen kuva joudutaan lataamaan kesken esityksen muistiin, voi se aiheuttaa viiveen esityksessä.

Jokainen yksittäinen cast member -elementti voidaan ladata etukäteen muistiin Lingin *preloadMember*-komennolla. Tällöin esitys ei hidastu elementin esittämisen aikaan, sillä elementti on jo ladattu muistiin. Komento voi olla esimerkiksi:

```
preloadMember "taustakuva"
```

Useampia cast membreitä voi ladata muistiin esimerkiksi seuraavalla komennolla, joka lataa muistiin kyseiset kaksi kuvaa sekä kaikki kuvat niiden väliltä cast-ikkunassa:

```
preloadMember "taustakuva1", "taustakuva2"
```

Pelkkä *preloadMember*-komento lataa cast-ikkunan elementtejä niin kauan kunnes muisti loppuu.

Jos haluaa esiladata pelkästään scorella sijaitsevia cast membreitä, voi käyttää Lingin *preLoad*-komentoa. Komento ei siis esilataa cast-ikkunassa olevia, eikä Lingin avulla scorelle vaihtuvia elementtejä. Mikäli *preLoad*-komennon perään ei laiteta parametreja, komento yrittää esiladata kaikki elementit kyseisestä frame-kohdasta eteenpäin. Esilataus voidaan suorittaa halutulta väliltä antamalla komennolle frame-kohtien parametrit. Seuraava esimerkkikomento esilataa scoren cast memberit framesta 10 frameen 50 asti:

```
preLoad 10, 50
```

### 2.6.2 Preloader-esilataus

Koko esityksen esilataaminen tapahtuu *preLoadMovie*-komennolla. Komento lataa *go to movie* tai *play movie* -komennolla avattavan esityksen kaikki ensimmäisen framen elementit muistiin. Isokokoisten esitysten esilataaminen saattaa olla hidasta, mutta lataamisen jälkeen esitys toimii sulavasti. Esimerkiksi alkuanimaatio-nimisen tiedoston esilataaminen suoritettaisiin komennolla:

```
preLoadMovie "Alkuanimaatio"
```

### 2.6.3 Splash Screen

Cd-romilla julkaistavien esitysten projektoreista eli käynnistystiedostoista saattaa tulla suurikokoisia tiedostokooltaan. Tällöin on suositeltavaa luoda projektorista pieni Splash Screen -latauskuva. Latauskuvan luomiseen voi käyttää esimerkiksi Directorin SplashScreenX-Xtraa, mutta latauskuvan voi tehdä ilman apuohjelmiakin. Latauskuvan tulee olla BMP-päätteinen ja se tulee sijoittaa samaan paikkaan projektoritiedoston kanssa. Kuvan nimeksi tulee sama kuin projektorin nimi on eli jos projektorin nimi on *projektor.exe*, tulee kuvan nimen olla *projektor.bmp*. (Kainulainen 2003: 451).

## 3 Cd-rom-esityksen toteuttamisen optimointi

### 3.1 Työympäristön tehokas hallinta

Macromedia Director MX:n käyttöliittymä on malliltaan samanlainen kuin muutkin Macromedian MX-sarjan ohjelmat. Näyttöruudulla aiemmin vanhemmissa versioissa vapaasti liikuteltavissa olleet ikkunat on MX-versioissa lukittu kiinteäksi osaksi muita ikkunoita ja valikoita, mikä saattaa hankaloittaa työskentelyä. Kainulainen (2003: 28) huomauttaa, että Directorin hyvä puoli on se, että käyttöliittymän voi jokainen muokata itselleen sopivaksi.

Käyttöliittymämalleista on käytettävissä sekä uusi että vanha ja myös molempien yhdistelmä. Oletusarvoisen MX-käyttöliittymän, jossa ikkunat ovat alkuperäisillä paikoillaan, voi asettaa valitsemalla *Window*-valikosta *PanelSets, Default*. Vanhan käyttöliittymämallin voi valita käyttöön valitsemalla *Window*-valikosta *PanelSets, Director 8*. Uutta käyttöliittymää käytettäessä on suositeltavaa ainakin cast-ikkunan ja scoren erottaminen toisistaan, jolloin niiden kokoa voi muuttaa.

#### 3.1.1 Näppäinkomennot

Directorin näppäinkomentoja käyttämällä voi nopeuttaa työskentelyä, joten ainakin useimmin käytettävät komennot kannattaa opetella suorittamaan näppäinkomennoilla. Esimerkiksi seuraavia näppäinkomentoja voi joutua käyttämään usein:

- Ctrl+1            Näyttämö
- Ctrl+2            Ohjauspaneeli
- Ctrl+3            Cast-ikkuna
- Ctrl+4            Score
- Ctrl+7            Työkalupaneeli
- Ctrl+Alt+S        Property Inspector

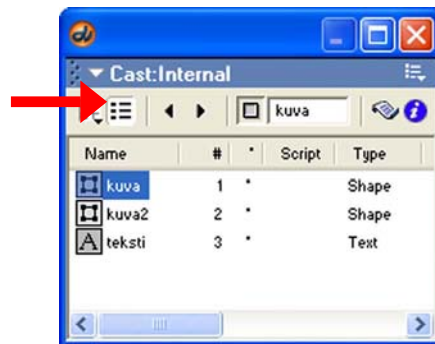
#### 3.1.2 Ohjauspaneeli

Ohjauspaneelilla ohjataan esityksen tempoa eli esitysnopeutta. Esityksen nopeus riippuu kuitenkin myös käyttäjän tietokoneen tehokkuudesta, joten heikkotehoisella tietokoneella monimutkaista grafiikkaa katsottaessa esitys hidastuu vaikka esitysnopeus olisi määritetty nopeammaksi. Kainulainen (2003: 32) toteaa, että myös hyvin tehokkaat tietokoneet voivat aiheuttaa ongelmia, mikä on kuitenkin harvinaista. Esitysnopeuden voi asettaa sopivaksi tehokkaammille tietokoneille valitsemalla *Modify*-valikosta *Movie, Playback* ja asettamalla *Lock Frame Durations* -kohdan aktiiviseksi, jolloin esityksen nopeus ei voi kasvaa haluttua suuremmaksi.

### 3.1.3 Cast-ikkuna

Cast-ikkuna arkistoi kaiken esityksessä käytetyn materiaalin. Cast-ikkunan mediaelementit määrittävät lähes koko Director-esityksen tiedostokoon. Jos cast-ikkunan elementti on linkitetty, se ei vie tilaa Director-esityksessä. Kokonainen cast-ikkuna voidaan myös linkittää esitykseen. (Kainulainen 2003: 33.)

Cast-ikkunan näkymän voi valita kahdesta vaihtoehdosta. Lista- eli list view -näkyvässä elementit ovat listattuina allekkain (Kuva 6). Thumbnail view -näkyvässä elementit näkyvät pienikokoisina kuvina (Kuva 7). Listanäkymää kannattaa käyttää ainakin silloin, jos cast-ikkunassa on paljon elementtejä. Listanäkymässä elementtejä mahtuu enemmän näkyville ikkunaan ja ne voi järjestellä esimerkiksi tyyppin tai nimen mukaisesti.



Kuva 6 Cast-ikkunan elementit listana



Kuva 7 Cast-ikkunan elementit kuvina

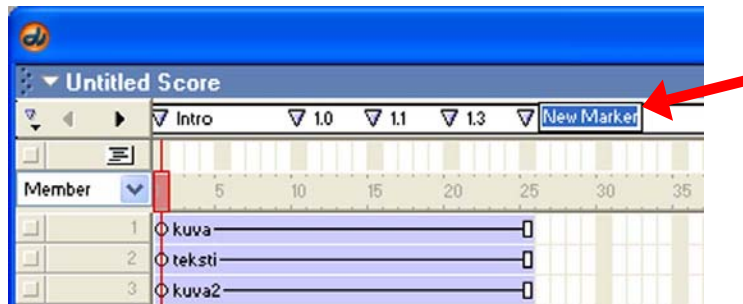
Cast-ikkunan näkymää voi muokata valitsemalla *Edit*-valikosta *Preferences, Cast*. Thumbnail-näkymää käytettäessä kannattaa asettaa *Row Width* -kenttään arvoksi *Fit to Window*, jolloin cast-ikkuna sovitaa ikkunaan mahtuvien ruutujen määrän ja jättää turhat vierityspalkit pois. (Kainulainen 2003: 34-38.)

Cast-ikkunan jokaisella cast memberillä tulisi olla muista elementeistä erottuva nimi. Nimettyihin elementteihin on helppo viitata Lingolla, vaikkakin viittaus voidaan tehdä myös järjestysnumeron perusteella (Kainulainen 2003: 37).

Laajemmassa esityksessä jokaiselle elementtityypille kannattaa luoda oma cast-ikkuna, jotka voi nimetä esimerkiksi Grafiikka, Ääni, Video ja Teksti. Tällöin elementit löytyvät nopeasti ja ne pysyvät hyvin arkistoituna. Cast-ikkunoiden nimet kannattaa pitää lyhyinä ja yksinkertaisina Lingon viittausten takia. Sisäiset (internal) cast-ikkunat ovat Director-esityksen kiinteitä osia. Ne kulkevat esityksen mukana automaattisesti, eikä niitä voi hyödyntää muissa Director-tiedostoissa. Ulkoisia (external) cast-tiedostoja voi sen sijaan käyttää useissa eri Director-esityksissä. (Kainulainen 2003: 39.)

### 3.1.4 Scoren framet

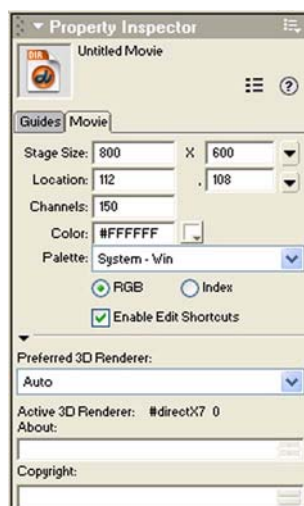
Scoren frameja kannattaa nimetä marker-merkinnöillä, joihin voidaan viitata Lingolla siirryttäessä esityksessä tiettyyn kohtaan. Loogisia nimiä on helpompi muistaa kuin framen järjestysnumeroita (Kuva 8). Lisäksi frame-numerointia muutettaessa joudutaan viittaukset muuttamaan myös koodiin, kun taas nimetyt framet toimivat aina muutoksista huolimatta. (Kainulainen 2003: 44-45.)



Kuva 8 Scoren frameja voi nimetä napsauttamalla yläreunan valkoisella alueella

### 3.1.5 Muita vinkkejä

Uuden esityksen tekeminen alkaa yleensä näyttämön asetusten määrittelyllä. Kun näyttämö on aktivoitu, voidaan asetuksia muuttaa Property Inspectorin Movie-välilehdellä (Kuva 9). Channels-kohdan arvo kertoo scoren kanavien lukumäärän, jota voi kasvattaa tarvittaessa. Kanavien määrää voi myös pienentää esityksen optimoimiseksi, mutta tällöin saatutettu hyöty on käytännössä pieni (Kainulainen 2003: 58). About ja Copyright -kohtiin voi kirjoittaa yleisiä tietoja sekä tietoja kopiointisuojauksesta, jos esitys menee levitykseen.



Kuva 9 Property Inspectorin Movie-välilehdellä voi muokata esityksen asetuksia

Director-esityksen maksimipituutta ei rajata ennalta. Cd-rom-projektin loppuvaiheessa saattaa olla vaikea tutkia varsinkin pidemmän esityksen mahdollisia ongelmakohtia, joten laajemman esityksen osiin pilkkominen kannattaa suunnitella jo etukäteen (Kainulainen 2003: 44-45). Esimerkiksi alkuintrosta, varsinaisesta esityksestä ja loppukohtauksesta voi tehdä omat Director-tiedostot.

Directorin ylälaidan *Edit*-valikosta löytyy *Undo*-toiminto (*Ctrl+Z*), jolla voi peruuttaa viimeisimmän tehdyn asian. Koska sitä enempää ei voi peruuttaa taaksepäin, kannattaa tiedot tallentaa usein ja laajemmissa esityksissä kannattaa käyttää tiedostojen nimeämisessä versionumerointia. Valitsemalla *File*-valikosta *Revert*, voi palauttaa esityksen siihen tilaan, mihin se jäi viimeisen tallennuksen jälkeen. (Kainulainen 2003: 52.)

Director-tiedostot kannattaa nimetä siten, että ne kuvaavat mahdollisimman hyvin kulloistakin esityksen versiota. Esityksen päivityksessä, erityisesti silloin, kun joku toinen henkilö suorittaa päivityksen, on kuvaavista nimistä paljon apua.

Director-esitystä tallentaessa kannattaa muistaa välillä tallentaa esitys *File*-valikon *Save and Compact* -komennolla. Komento järjestelee castikkunan mediaelementit, poistaa tuhottujen elementtien muistiin jääneitä viittauksia, jotka kasvattavat tiedostokokoa ja kääntää kaikki castikkunan koodit ilmoittaen mahdollisista virheistä. (Kainulainen 2003: 56.)

## 3.2 Mediaelementtien optimointi

Koska cd-levyjen tallennuskapasiteetti on rajallinen, tulee erityisesti isokokoisten esitysten multimediaelementit optimoida. Elementtien optimoinnilla pyritään saavuttamaan paras mahdollinen laatu mahdollisimman vähäisellä digitaalisella informaatiolla. Äänien ja videoiden tiedostokoot ovat yleensä suurimpia verrattuna muihin multimediaesityksen elementteihin, joten niiden optimointiin tulee kiinnittää erityisen paljon huomiota.

### 3.2.1 Teksti

Multimediaesityksen tekstisisältöä laatiessa tulee muistaa, että tekstin luettavuus muuttuu, kun se siirretään paperilta näytölle. Näytöltä luettava teksti on parhaimmillaan hyvin tiivistettyä ja lyhyisiin kappaleisiin jaoteltua sekä typografialtaan selkeää. Erilaiset vierityspalkit heikentävät tekstin luettavuutta.

Directorin käyttämät tekstityypit ovat perustekstit (*text member*) ja tekstikentät (*field member*). Perusteksteillä ei ole mitään erityisominaisuuksia, mutta tekstikentät voidaan ohjelmoida Lingolla. Tekstikentät vievät vähemmän tilaa kuin normaalit tekstit, sillä esimerkiksi perusteksti



”This is a test” vie tilaa n. 1,6 kilotavua, kun taas sama teksti tekstikentässä vie tilaa vain n. 0,07 kilotavua (Rosenzweig 2003: 53). Ero korostuu huomattavasti, kun käytetään antialisoitua tekstiä. Kainulainen (2003: 125) kuitenkin huomauttaa, että koska myös tavallinen teksti vie vähän tilaa, ei tällä ole yleensä suurta käytännön merkitystä.

Kainulainen (2003: 118-122) toteaa, että tekstit ja Lingo-skriptit voidaan kirjoittaa valmiiksi esimerkiksi Wordissa tai Notepadissa. RTF- tai TXT-muotoon tallennetut tekstit voidaan tuoda Directoriin *File*-valikon *Import*-komennolla. Tällöin Director kysyy, halutaanko teksti avata tavallisena tekstielementtinä vai Lingo-skriptinä. Tavalliset tekstit, joissa on käytetty perusmuotoiluja, kannattaa tuoda Directoriin RTF-muodossa (Rosenzweig 2003: 47). Tekstin ominaisuuksia voi tarkastella Property Inspectorissa. DTS-ominaisuutta käyttämällä esityksen ruudunpäivitys nopeutuu. DTS asettaa tekstin näkymään kaikkien muiden spritejen päälle.

Antialisointi on tekniikka, jota käytetään bittikarttagrafiikassa muun muassa fonttien reunojen pehmentämiseen. Ilman pehennystä suurikokoisten fonttien reunat näyttävät sahalaitaisilta. Antialisoidun tekstin ruudunpäivitys on hitaampaa kuin normaalin tekstin, joten pehennystä kannattaa käyttää harkitusti ja vain suuremmissa teksteissä kuten esimerkiksi otsikoissa (Kainulainen 2003: 123).

### 3.2.2 Kuva

Multimediaesityksessä käytettävät kuvat kannattaa optimoida esityksen toiminnan nopeuttamiseksi ja tiedostotilan säästämiseksi. Esityksen kuvamateriaalia suunniteltaessa kannattaa miettiä, kuinka tärkeässä osassa kuvat esityksessä ovat ja kuinka paljon niitä tulee yhteensä. Mikäli kuvat ovat tärkeä informaatiolisä cd-romille ja niitä on suhteellisen vähän, kannattaa kuvien laatuun panostaa.

Bittikarttakuva muodostuu kuvapisteistä, pikseleistä, joita jokaisessa kuvassa on tietty määrä pysty- ja vaakasuunnassa. Kuvan pikselikoko vaikuttaa tiedostokokoon ja kuvan tarkkuuteen. Kuvan resoluutiolla ei ole merkitystä multimediaesityksissä, sillä ne näkyvät tietokoneen ruudulla siten, että yksi kuvapiste vastaa yhtä kuvapistettä näytöllä (Keränen, Lamberg & Penttinen 2000: 99). Cd-rom-esityksen kuvien resoluutioksi riittää 72 ppi (pixels per inch).

Kuvan värisyvyys eli bittien määrä kuvapistettä kohti valitaan kuvan ominaisuuksien ja käyttökohteen mukaan. RGB (Red/Green/Blue) -väritilaa kannattaa käyttää aina, kun värikuvaa tullaan näyttämään tietokone-näytöllä (Keränen ym. 2000: 93). RGB-kuva voidaan esittää näytöllä ilman värimuutoksia, joten kuvan esittäminen on nopeaa (Keränen ym. 2005: 90). Kainulainen (2003: 77) suosittelee, että kuvat luodaan tarvittaessa oikeaan bittimäärään kuvankäsittelyohjelmassa ennen Directoriin tuomista. Hän myös toteaa, että nykyisille tietokoneille suunnitelluissa

tuotannoissa ei ole mitään syytä käyttää alle 16-bittisiä kuvia, sillä esimerkiksi 32-bittiset kuvat näyttävät paljon paremmilta ja voivat jopa viedä vähemmän tiedostotilaa kuin vastaavat 8-bittiset kuvat.

Bittikarttakuvan tiedostokokoa voidaan pienentää valitsemalla sopiva tiedostomuoto tallennettaessa. Häviötöntä pakkaustapaa, kuten esimerkiksi TIFF-tiedostomuotoa käytetään yleensä logoille ja grafiikalle. Häviöttömässä pakkauksessa kuvan laatu ei heikenny. Häviöllinen pakkaus, kuten esimerkiksi JPEG-tiedostomuodon käyttö heikentää hieman kuvan laatua, sillä se pudottaa kuvan yksityiskohtien ja värien määrää. Häviöllinen pakkaustapa sopii hyvin valokuville, sillä niiden laadussa ihmissilmä tuskin huomaa muutosta.

### 3.2.3 Ääni

Taustamusiikin tiedostokokoa voi pienentää toistamalla lyhyttä taustaa ääntä niin, ettei käyttäjä huomaa äänen alkaneen uudestaan. Mikäli esityksessä toistuu sama äänitiedosto useammassa samaan esitykseen kuuluvassa Director-tiedostossa, kannattaa ääni sijoittaa yhteen jaettuun ulkoiseen cast-tiedostoon, josta se voidaan linkittää myöhemminkin muihin esityksen osiin (Kainulainen 2003: 128).

Äänen laadulla tarkoitetaan äänen näytteenottotaajuutta, resoluutiota ja kanavien määrää. Näytteenottotaajuus ilmaistaan kilohertseinä ja resoluutio bitteinä. Kanavien määrä voi olla mono tai stereo riippuen käyttötilanteesta. Äänitiedoston kokoa voidaan pienentää muuttamalla näytteenottotaajuutta pienemmäksi, mutta tällöin myös äänen laatu kärsii. Näytteenottotaajuus valitaan äänen tyyppin mukaan, jolloin puheen esittämiseen riittää yleensä 11 kHz, perusmusiikille 22 kHz ja cd-tasoiselle musiikille 44 kHz (Kainulainen 2003: 130). Kainulainen (2003: 130) suosittelee äänille 16-bittistä resoluutiota, vaikka se viekin kaksi kertaa enemmän tilaa kuin 8-bittinen ääni. Hän kehottaa pudottamaan tiedostokokoa mieluummin näytteenottotaajuutta pienentämällä. Stereoääni voidaan muuntaa monoääneksi, jolloin tiedoston koko puolittuu, mutta tällöin menetetään stereotoiston edut.

Äänten optimoinnissa kannattaa myös muistaa kiinnittää huomiota siihen, että käytettävät äänet ovat tiiviitä ja ”trimmattuja” eli kaikki ylimääräinen, kuten tyhjät kohdat alussa ja lopussa on poistettu. Tällöin äänet alkavat oikeasta kohdasta ja käyttävät vain tarvittavan määrän tiedostotilaa. (Rosenzweig 2003: 715.)

### 3.2.4 Video

Videon laatumääritykset eli videon koko, kuvataajuus ja pakkausmenetelmä määritellään jakelukanavan ja esitysmuodon mukaisiksi. Videotiedoston optimoinnilla pyritään säilyttämään kuvan ja äänen laatu hyvänä

huolimatta tiedostokoon ja kaistanleveyden pudottamisesta. (Keränen ym. 2005: 232.)

Kainulainen (2003: 136) suosittelee esimerkiksi 800 x 600 pikselin kokoiseen esitykseen 320 x 240 pikselin kokoisia videoesityksiä. Pienikokoinen esitys pyörii hyvin hitaammallakin näytönohjaimella. Mikäli esitystä näytetään suurelta valkokankaalta, kannattaa videon olla suurempi. Esitysruudun koko päätetään digitointivaiheessa tai kuvaa pakattaessa esitysmuotoon. Keränen ym. (2005: 233) toteavat, että paras tapa pienentää videoesityksen kokoa on ruudun puolittaminen, jolloin joka toinen kuvapikseli häviää ja kuvan terävyys säilyy mahdollisimman hyvänä.

Toinen tapa keventää videon tiedostokokoa on pudottaa sekunnissa esitettävien ruutujen määrää. Kuvataajuuden ollessa pienempi kuin 18 kertaa sekunnissa, videon liike saattaa näyttää nykivältä. Videota kannattaa kuitenkin digitoida suuremmalla nopeudella, jos mahdollista, sillä kuvataajuutta voidaan pudottaa myöhemmässäkin vaiheessa (Keränen ym. 2005: 234).

Videotiedoston kokoon vaikuttaa myös kuvan laatu. Kuvan laatuun taas vaikuttaa videokuvan pakkaamisessa käytetty codec ja pakkaussuhde. Codecien pakkaussuhdetta voidaan useimmiten muuttaa. Koska myös videon kesto vaikuttaa videotiedoston kokoon, pakkauksesta puhutaan usein tiedonsiirtonopeutena. (Keränen ym. 2005: 234-235.)

Cd-rom-asetat pystyvät teoriassa siirtämään tietoa 50-kertaisella nopeudella, mutta käytännössä tämä pitää paikkansa vain levyn ulkoreunaan tallennetussa tiedossa monistetuissa levyissä. Myös levyn ja aseman liikkaisuus voi hidastaa tiedon lukemista. Ohjeistuksena voidaan pitää sitä, että cd-rom-aseman siirtonopeus on maksimissaan aseman normaali lukunopeus (150 kt/s) kaksinkertaisena (n. 300 kt/s eli n. 2400 kbit/s) ja tähän nopeuteen sisältyy myös videon ääni. Uusimmilla codeceilla tämä mahdollistaa erittäin hyvän kuvanlaadun 320 x 240 pikselin kokoisessa videoesityksessä. Mikäli halutaan käyttää hyvälaatuisia ääntä, joudutaan kuvanlaadusta hieman tinkimään. (Keränen ym. 2005: 236.)

### 3.2.5 Flash

Directorissa voi hyödyntää monipuolisesti Flashilla luotuja esityksiä, sillä molemmat ohjelmat kuuluvat Macromedian tuoteperheeseen. Flashilla luotu grafiikka on vektorigrafiikkaa, mikä mahdollistaa pienen tiedostokoon sekä vektorielementtien hyvän skaalattavuuden.

Kainulainen (2003: 144-145) toteaa, että luultavasti suosituin tapa käyttää Flashia Directorin osana on tehdä esityksen painikkeet Flashissa ja tuoda ne sitten Directoriin. Hän jatkaa, että toinen suosittu tapa on käyttää Flashilla luotuja pieniä splash screen -animaatioita osana Directorin Shockwave-esityksiä. Tällöin nopeasti käynnistyvä Flash-animaatio viih-

dyttää käyttäjää varsinaisen esityksen vielä latautuessa. Flashin tuomisen Directoriin mahdollistaa Flash Asset -Xtra. Jokainen Flash-animaatio vie osansa Xtran toimintakyvystä, joten samaan aikaan suoritettavat animaatiot voivat heikentää esityksen suorituskykyä (Kainulainen 2003: 145).

Flash-elementtien ominaisuuksia voi muuttaa Property Inspectorissa. Flash-spriteille kannattaa asettaa Paused-ominaisuus, jos kyseinen Flash-esitys on tarkoitettu vain paikallaan pysyväksi grafiikaksi. Tällöin Flash-esitys ei animoidu ja Director-esityksen suorituskyky paranee. Kun Property Inspectorissa aktivoi DTS-ominaisuuden, Flash-esityksen toimivuus paranee, mutta elementti näkyy näyttämöllä aina päällimmäisenä.

Director tukee Flash-spriteissa vain Copy-, Transparent-, Background Transparent- ja Blend-peittovärejä. Esityksen suorituskyvyn parantamiseksi kannattaa suurikokoisissa Flash-spriteissa käyttää vain Copy Ink-peittoväriä. Jos elementin DTS-ominaisuus on aktivoituna, spritejen painovärimääritykset eli Ink-efektit eivät toimi. (Kainulainen 2003: 146.)

### 3.2.6 3D-elementti

Director-esityksen 3D-elementtien optimointi onnistuu samankaltaisilla Lingo-komennoilla ja työkaluilla kuin muidenkin mediaelementtien optimointi. Directorin kirjastopaletin valmiita 3D-skriptejä voi hyödyntää 3D-elementtien optimoinnissa. Esimerkiksi paletista löytyvän Actionskohdan *Level of Detail* -behaviorilla voi poistaa 3D-mallista yksityiskoh-  
tia (Kainulainen 2003: 443).

Suurin yksittäinen keino pienentää 3D-Shockwave-esityksen tiedostokoko-  
koa lienee tekstuurien eli 3D-kappaleiden pintakuvion määrän huomatta-  
va raja. Tekstuurien fyysisen koon pitäisi olla luvun kaksi potenssi eli  
tekstuureiksi haluttujen kuvien tulisi muodostua luvuista: 2, 4, 8, 16, 32,  
128 jne. Tekstuurin koko kannattaa olla alle 256 x 256 pikseliä. Pieni ko-  
ko on hyödyllistä erityisesti silloin, kun 3D-objekti on tarkoitettu liikku-  
vaksi tai animoitavaksi, koska suuremmilla tekstuureilla ei saavutettaisi  
hyötyä. Myös tekstuurien värimäärään kannattaa kiinnittää huomiota, sil-  
lä esimerkiksi 32-bittiset kuvat vievät paljon näytönohjaimen muistia. Jos  
kuvissa ei tarvita alfa- eli maskikanavaa, se kannattaa ehdottomasti pois-  
taa tekstuurista. (Kainulainen 2003: 444.)

Antialisoinnilla voidaan pehmentää tekstin lisäksi myös 3D-kuvioiden  
reunoja. On suositeltavaa käyttää antialisointia vain sellaisissa esityksis-  
sä, joissa esitysnopeus ja 3D-grafiikan sulavuus ei ole ratkaisevaa. Lin-  
gon avulla voidaan tarkistaa, tukeeko käyttäjän tietokone 3D-grafiikkaa  
ja antialisointia, mutta loppukäyttäjälle kannattaa mieluummin antaa  
mahdollisuus asettaa antialisointi päälle tai pois päältä. (Kainulainen  
2003: 445-446.)

### 3.3 Esityksen testaaminen

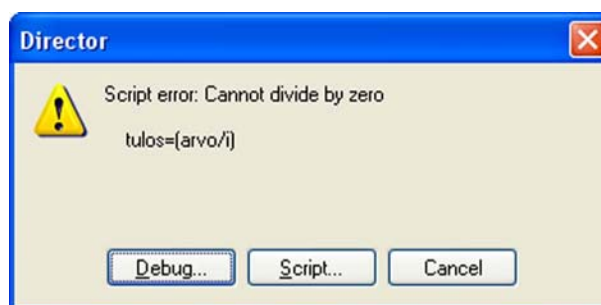
Mitä laajempi ja monimutkaisempi multimediaesitys on, sitä aikaisemmin tulee testaus aloittaa toteutusvaiheen aikana. Testauksen tarkoituksena on selvittää esityksen mahdolliset virheet, tarkistaa tekninen toimivuus erilaisilla laitteistokokoonpanoilla sekä testata esityksen selkeyttä ja käytettävyyttä. Testauksessa on hyvä olla mukana esityksen tuotantoon osallistuneita, toimeksiantajan edustajia sekä projektin ulkopuolisia, kohderyhmään kuuluvia henkilöitä.

Multimediatuotantojen testaus luokitellaan usein kahteen vaiheeseen: alfa- ja beta-testaukseen. Alfa-testauksella tarkoitetaan yleensä sovelluksen testausta oman työryhmän kesken erilaisilla laitteistokokoonpanoilla. Beta-testaus suoritetaan ulkopuolisilla mieluiten kohderyhmään kuuluvilla henkilöillä. Beta-testauksessa voidaan kiinnittää enemmän huomiota esityksen ulkoisiin seikkoihin ja toiminnallisuuteen, koska alfa-testaus karsii suurimmat tekniset virheet pois. Lukkari (2004: 21) lisää master-testauksen yhdeksi vaiheeksi. Hän esittää, että master-versiolle tehdään toiminnallisuuden ja sisällön lopputarkistus sekä pakkauksen, kansien tai ulkoasun tarkistus.

Multimediaesityksen testiversioiden tallennusformaattina kannattaa käyttää cd-rw-levyjä, sillä ensimmäiset testiversiot sisältävät usein virheitä, joten näin säästyy cd-romien hankintakustannuksia (Kainulainen 2003: 184).

### 3.4 Virheiden jäljittäminen ja korjaaminen

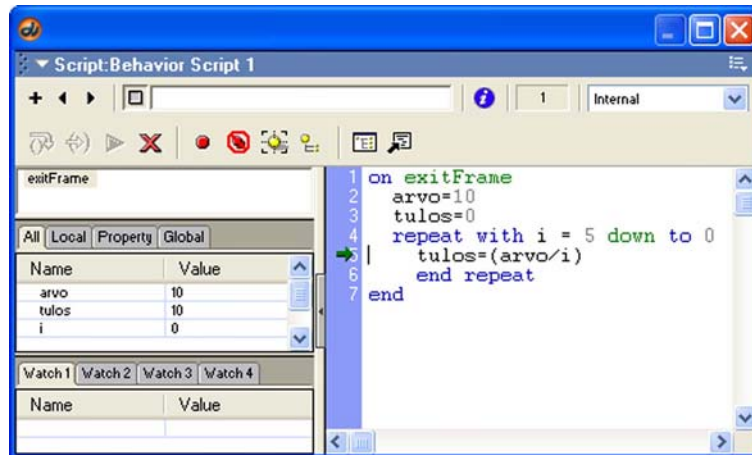
Directorin Lingo-ohjelmointia käytettäessä on todennäköistä, että virheitä tulee enemmän tai myöhemmin. Pienet kirjoitusvirheet erottaa komenon värityksestä, koska Directorin skripti-ikkuna värittää esimerkiksi tietyt määrittelyt ja kommentoinnit eri väreillä. Kun skripti-ikkunaan jää virheellinen komento, antaa Director virheilmoituksen avaamalla Script error -ikkunan (Kuva 10). Ikkuna kertoo virheen syyn ja näyttää virheellisen kohdan koodista. Virheellistä koodia pääsee tutkimaan Debuggerityökaluun napsauttamalla *Debug*-painiketta. *Script*-painiketta napsauttamalla pääsee skripti-ikkunaan muuttamaan koodia.



Kuva 10 Skripti-ikkuna ilmoittaa virheellisestä komennosta

### 3.4.1 Debugger

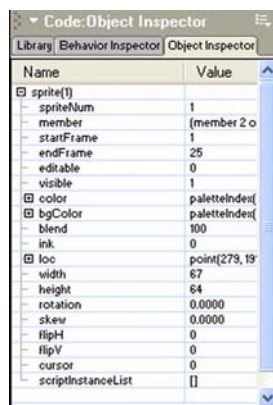
Silloin kun suoraan virheilmoituksesta ei näe, missä vika koodissa on, kannattaa siirtyä Debugger-työkaluun (Kuva 11). Debuggerissa koodia suoritetaan rivi kerrallaan ja tarkkaillaan muuttujien arvoja. Director MX-versiosta alkaen Debugger-työkalu on edistyneempi aikaisempiin versioihin nähden. Debugger käyttää samoja ominaisuuksia kuin Object Inspector, mutta Debuggerissa esitys on aina pysäytettynä, kun taas Object Inspector näyttää tiedot reaaliajassa (Kainulainen 2003: 466).



Kuva 11 Debugger-työkalun avulla selvitetään esityksen virheitä

### 3.4.2 Object Inspector

Object Inspector -työkalu on ollut käytössä Directorissa MX-versiosta alkaen. Työkalu on sijoitettu Directorin oikeaan laitaan Property Inspectorin alapuolelle. Object Inspector mahdollistaa erilaisten mediaelementtien ja objektien ominaisuuksien ja arvojen tarkkailun reaaliajassa (Kuva 12). Esimerkiksi 3D- ja Flash-elementtien muuttujien arvoja voi tutkia ilman ohjelmointiosaamista. Object Inspectoria voi kätevästi hyödyntää Lingo-skriptien testauksessa, koska sen kautta voi muuttaa arvoja koskematta itse Lingo-skriptiin (Kainulainen 2003: 466).



Kuva 12 Object Inspector näyttää erilaisten objektien ominaisuuksia

### 3.4.3 Message-ikkuna

Message-ikkunaa voi käyttää esimerkiksi muuttujien ja komentojen testaamiseen (Kuva 13). Message-ikkunan saa esiin näppäinkomennolla *Ctrl+M*, napsauttamalla Directorin yläaidan *Message Window* -painiketta tai valitsemalla *Window*-valikosta *Message*. Ikkunaan kirjoitetut komennot häviävät, kun ikkuna suljetaan, joten Message-ikkunaa ei voi käyttää varsinaiseen ohjelmointityöhön. Lingo-komennot, joiden toiminnasta ei ole varma, voi testata Message-ikkunan avulla. Ikkunan yläosasta löytyy Lingo-komennot listattuina samoin kuten skripti-ikkunassa.



Kuva 13 Message-ikkunaa käytetään Director-esitysten testaamiseen

### 3.4.4 Trace-ominaisuus

Message-ikkuna on ensisijainen työkalu trace-virheenjäljitykseen (Rosenzweig 2003: 675). Edistyneempään virheenkäsitelyyn tarkoitettu trace-ominaisuus tulostaa jokaisen suoritettavan komennon Directorin Message-ikkunaan. Ominaisuuden voi ottaa käyttöön napsauttamalla Message-ikkunan *Trace*-painiketta tai kirjoittamalla Message-ikkunaan seuraavan Lingo-komennon:

```
the trace = 1
```

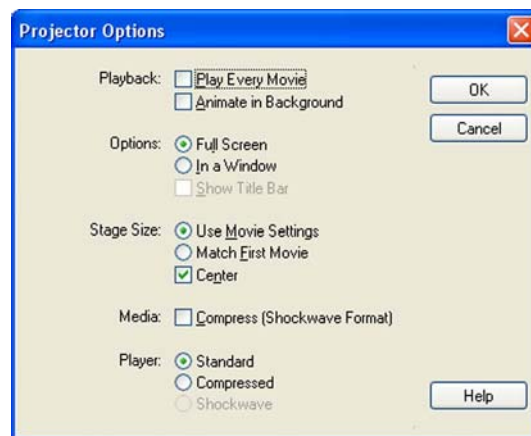
Trace-komentoja on kolme erilaista: *the trace*, *the traceLoad* ja *the TraceLogFile*. Käyttämällä *the trace* -komentoa Message-ikkunaan tulostuu jokainen suoritettava koodirivi sekä muuttujien arvojen muutokset. Debugger-työkalu toimii hyvin pitkälti samoin tavoin kuin *the trace* -komento. *the traceLoad* -komento tulostaa Lingo-komentojen sijasta tietoja cast membreistä. *the traceLoad* -komennosta on kuitenkin vähän käytännön hyötyä, sillä Lingon *put*-komennolla saa tulostettua samat tiedot. Mikäli trace-komennoista on kuitenkin ohjelmoinnissa hyötyä, kannattaa käyttää myös *the TraceLogFile* -komentoa. Sen avulla voi tallentaa *the traceLoad* -tulostukset ulkoiseen tekstitiedostoon. (Kainulainen 2003: 467-469.)

## 3.5 Cd-rom-esityksen julkaisu

### 3.5.1 Projektori

Cd-rom-esityksen julkaisua varten tulee Directorilla luoda esityksestä exe-päätteinen projektoritiedosto. Kun Director-esitys on julkaisuvalmis ja tallennettu, projektorin luominen tapahtuu valitsemalla *File*-valikosta *Create Projector*. Rosenzweig (2003: 716) toteaa projektorien suurimman ongelman olevan niiden hidas käynnistyminen. Projektorien käynnistymistä voi nopeuttaa esimerkiksi poistamalla projektorista ylimääräiset Xtrat. Projektorit toimivat tehokkaammin loppukäyttäjän kovalevyllä kuin cd-romilta, joten jos suorituskyvyn tulee olla optimaalinen, voi projektoriin liittää automaattisen asennuksen.

Create Projector -ikkunan *Options*-painiketta napsauttamalla voi määrittellä projektorin asetukset (Kuva 14). Jos pakkausmenetelmäksi ei valitse mitään (Standard), esitys käynnistyy nopeasti, mutta on tiedostokooltaan suurin. Compressed-pakkaustapa pienentää tiedoston kokoa, mutta pidentää käynnistysaikaa muutamalla sekunnilla. Verkkojulkaisuissa voi käyttää Shockwave-pakkausta, jolloin Director luo huomattavasti pienemmän projektoritiedoston. Loppukäyttäjä tarvitsee tällöin Shockwave-ohjelman esityksen katsomiseen. (Kainulainen 2003: 175.)



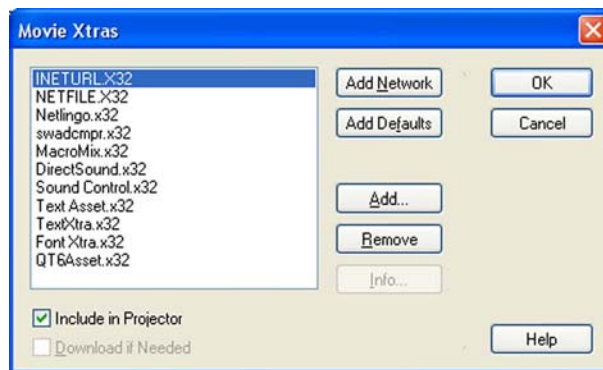
Kuva 14 Projector Options -ikkunassa voi muuttaa projektorin ominaisuuksia

Shockwave-projektori käynnistyy nopeasti, koska kaikki Xtrat tallennetaan ulkoiseen xtras-nimiseen kansioon ja lopullisen projektoritiedoston koko jää pieneksi. Projektorin käynnistymisnopeuteen vaikuttaa myös se, että normaalin projektorin käynnistyessä sen sisään tallennetut Xtrat puretaan väliaikaisesti tietokoneen temp-kansioon. Shockwave-tallennuksessa Xtrat on valmiiksi purettu ulkoisiksi tiedostoiksi, mikä lyhentää projektorin käynnistymisaikaa. (Kainulainen 2003: 176.)

Xtrat ovat ohjelmistoihin asennettavia lisäpaketteja, joilla voi tuoda uusia tai parannettuja lisäominaisuuksia Directoriin. Xtrat kasvattavat projekto-



rin tiedostokokoa ja siten latautumiseen kuluvaan aikaan. Käynnistymistä voidaan nopeuttaa käyttämällä ulkoisia Xtroja ja poistamalla esitykseen mahdollisesti jääneet ylimääräiset Xtrat. Cd-rom tuotannoissa käytetyt Xtrat tulisi sijoittaa esityksen mukaan Xtras-nimiseen alikansioon (Kainulainen 2003: 478). Esityksen tarvitsemat Xtrat voi tarkastaa Movie Xtras -ikkunasta (Kuva 15), jonka saa esille valitsemalla *Modify*-valikosta *Movie, Xtras*.



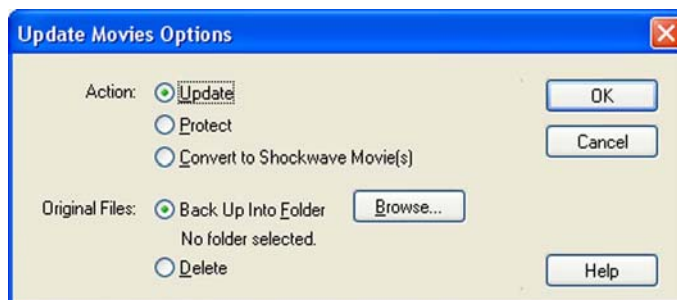
Kuva 15 Movie Xtras -ikkunassa voi tarkastella esityksen lisäominaisuuksia

### 3.5.2 Stub-projektori

Stub-projektori kannattaa hyödyntää etenkin esityksissä, joita päivitetään. Stub-projektori ei sisällä muuta kuin yksinkertaisen Lingo-komennon, joka käynnistää varsinaisen esityksen. Kun esitystä myöhemmin päivitetään, ei stub-projektorille tarvitse tehdä muutoksia. Kainulainen (2003: 176) suosittelee, että stub-projektoriin liitetään myös QuickTime-tunnistin. Hän jatkaa, että koska tällainen tunnistin luodaan Lingolla, ei projektorin koko kasva juuri ollenkaan. Ja kun projektorista jätetään vielä kaikki Xtrat pois, saadaan erittäin pienikokoinen ja nopeasti käynnistytävä tiedosto. QuickTime-tunnistin tulee tietenkin luoda vain, jos esityksessä käytetään MOV-päätteisiä videoita.

### 3.5.3 Tiedostojen päivittäminen ja suojaaminen

Directorin päivitysominaisuuden voi ottaa käyttöön valitsemalla *Xtras*-valikosta *Update Movies*, jolloin avautuu kuvan 16 kaltainen ikkuna. Ikkunan *Update*-toiminto päivittää automaattisesti kaikki vanhat DIR-päätteiset tiedostot uutta Director-versiota vastaaviksi tiedostoiksi. *Protect*-toiminto suojaa julkaistavat tiedostot niin, ettei loppukäyttäjä pääse koodeihin tai mediaelementteihin käsiksi. *Convert to Shockwave Movie(s)* pakkaa tiedostot, jolloin niiden lataaminen nopeutuu. Suojattuja tai pakattuja tiedostoja ei voi enää avata Directorissa, joten alkuperäiset tiedostot kannattaa pitää tallessa. (Kainulainen 2003: 182.)



Kuva 16 Update Movies Options -ikkunan avulla voi päivittää ja suojata tiedostoja

Suojattujen Director-tiedostojen päätteeksi tulee DXR ja suojattujen cast-tiedostojen päätteeksi CXT. Suojattu cd-rom-esitys saattaa toimia nopeammin kuin Shockwave-esitys, koska sen ei tarvitse olla pakattu. Pakkaamattoman suojatun esityksen käyttö on suositeltavaa, jos levytilaa riittää. Suojatut esitykset eivät sisällä muokkaukseen liittyvää informaatiota eivätkä ohjelmaa, joka toistaa esitystä. (Armstrong 2002: About distribution formats.)

### 3.5.4 Cd-rom tallennusformaattina

Cd-rom-levyille tallennetaan tiedostomuotoista tietoa, jota voidaan lukea cd-rom- tai dvd-aseamalla varustelluilta tietokoneilta. Standardilevyille mahtuu tietoa noin 650 megatavua. Kainulainen (2003: 183) toteaa, että cd-rom-esitysten tilaukset ovat vähentyneet nopeiden verkkoyhteyksien takia, sillä ennen vain cd-romille suunniteltuja esityksiä voidaan toteuttaa myös verkkoon. Kainulainen jatkaa, että cd-rom on kuitenkin suositeltava ja laajasti tuettu tallennusformaatti. Cd-romin monistaminen ja jakelu on helppoa ja edullista.

Kuntaliiton Efeko Oy:n tutkimuksen mukaan Suomi on edelläkävijä Internet-yhteyksissä, mutta silti joka neljäs 18-70-vuotias suomalainen ei pääse Internetiin kotona, työssä tai vapaa-ajan harrastuksissa (Tolvanen 2005: A5). Cd- tai dvd-asema sen sijaan löytyy nykyään lähes jokaisesta tietokoneesta.

Lopullisia cd-rom-levyjä poltettaessa ei tule käyttää multisession-polttotyyppiä, sillä se saattaa aiheuttaa virheitä cd-romille eikä levyille muutenkaan ole tarvetta myöhemmin lisätä mitään, kun esitys on jo valmis. Polttoon kannattaa varata hieman laadukkaampia tyhjiä levyjä. Lopullisia painoon vietäviä cd-romeja ei tule polttaa nopeimmilla polttonopeuksilla. (Kainulainen 2003: 184.)

### 3.5.5 Cd-romin käyttöohjeistus ja jakelu

Kainulainen (2003: 188) suosittelee, että jokaisen julkaistavan ja levitetävän multimediaesityksen mukaan liitetään readme.txt-tyylinen tiedosto, jossa kerrotaan esityksen käynnistämiseen liittyvät ohjeet, esityksen ver-

sionumero, laitteistovaatimukset ja kopiointisuojaustiedot. Ohjeistukseen voi myös laittaa vaikkapa johdannoksi haluamaansa tietoa cd-romin sisällöstä.

Testaus- ja korjausvaiheiden jälkeen cd-rom on valmis luovutettavaksi toimeksiantajalle jatkotoimenpiteisiin. Cd-romilla jaettavasta laajempaan levitykseen tarkoitettusta multimedialta teetetään ohjelman masterointi ja monistus. Valmiille esitykselle suunnitellaan kannet ja pakkaus sekä kootaan käyttöohjeet. Esityksen tilaajalle annetaan tarvittaessa käyttäjäkoulutusta tuotteen käyttöön ja päivitykseen.

## 4 Yhteenveto

Tutkintotyöstä kehittyi Directoriin syventynyt, monipuolinen tietopaketti optimaalisen cd-rom-esityksen kehitysvaiheista suunnittelun lähtökohdistuotantoon ja julkaisuun asti. Oli yllättävää huomata, kuinka moni asia vaikuttaa cd-romin toimivuuteen ja miten paljon ongelmia voidaan ennaltaehkäistä cd-romin huolellisella suunnittelulla ja toteutuksella. Cd-rom-esityksen optimointi liittyy myös oleellisesti esityksen päivitettävyyteen. Kompaktin kokoista ja huolellisesti tuotettua cd-romia on vaivatonta päivittää.

Cd-romin optimoinnilla saavutetaan huomattavia hyötyjä etenkin laajoissa, monimutkaisissa ja tiedostokooltaan isoissa esityksissä. Pienemmissäkin esityksissä kannattaa panostaa vähintään multimediaelementtien optimointiin, sillä ne määräävät suurimmaksi osaksi kuinka kevyt tai raskas esityksestä tulee. Nopeasti ja sulavasti toimiva cd-rom-esitys ei ehdi pitkästyttää loppukäyttäjää. Valitettavan usein kaupallistenkin cd-romien joukossa on hitaasti toimivia multimediaesityksiä, joiden käyttö jää vähiin negatiivisten käyttökokemusten takia.

Tutkintotyö tarjoaa paljon vinkkejä erityisesti isokokaisen cd-rom-esityksen optimointiin, jonka suorituskyvyltä vaaditaan paljon ja jossa pienikin nopeushyöty on tarpeen. Tällaisessa esityksessä pienemmätkin seikat, kuten esimerkiksi cast memberien muistista poistamisen säätäminen tai projektorin asentaminen loppukäyttäjän kovalevyllä voivat vaikuttaa merkittävästi lopputulokseen.

Jos cd-romia päivitetään tai hyödynnetään muuten myöhemmin, kannattaa esityksen rakenteesta tehdä mahdollisimman selkeä. Rakenteen koostamisessa voi hyödyntää Directorin tarjoamia työkaluja, joita tutkintotyö esittelee. Lingon käytössä tulee olla huolellinen ja monimutkaisemmat koodit kannattaa aina selventää kommentteilla, jolloin esityksen muokkaaminen nopeutuu ja helpottuu.

Vaikka tutkintotyöhön kaavailtu cd-rom jäi tutkintotyöstä pois, ei sen tekeminen tutkintotyön ohella ollut yhdentekevää. Työt tukivat toisiaan; cd-romista tuli vähäisenkin tutkintotyöteorian soveltamisen avulla todella pienikokoinen, nopeasti käynnistyvä ja sulavasti toimiva esitys, ja tutkintotyön tekemistä taas auttoi huomattavasti aidon cd-rom-projektin suunnittelu ja toteuttaminen sen rinnalla.

## Lähteet

Armstrong, Jay, Brown, George, Gowin, Stephanie & Statler, Tim 2002. [Director Help] [viitattu 23.10.2005]. Macromedia Director MX. Using Director MX. USA: Macromedia, Inc.

Kainulainen, Pasi 2003. Director MX. Jyväskylä: Docendo.

Keränen, Vesa, Lamberg, Niko & Penttinen, Jukka 2005. Digitaalinen media. Jyväskylä: Docendo.

Keränen, Vesa, Lamberg, Niko & Penttinen, Jukka 2000. Multimedia. Jyväskylä: Teknolit.

Lukkari, Ulla 2004. Digitaalisen sisältötuotantoprojektin hallinta. Helsinki: Edita.

Rosenzweig, Gary 2003. Special Edition Using Macromedia Director MX. USA: Que Publishing.

Tolvanen, Kristiina 2005. Internet pitäisi olla kaikkien saatavilla sanomalehden tapaan. Aamulehti 23.10.2005, A5.