



TAMPEREEN
AMMATTIKORKEAKOULU

KONSERTTIVALAISTUKSEN OHJELMOINTI GRANDMA2-YMPÄRISTÖSSÄ

Esko Ansami

Opinnäytetyö

Toukokuu 2016

Elokuvan ja television koulutusohjelma

Teatterin ja tapahtumien audiovisuaalinen suunnittelu



TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Elokuvan ja television koulutusohjelma
Teatterin ja tapahtumien audiovisuaalinen suunnittelu

ANSAMI ESKO

Konserttivalaistuksen ohjelmointi GrandMA2-ympäristössä

Opinnäytetyö 37 sivua, joista liitteitä 0 sivua

Toukokuu 2016

Teatteri- ja estradivalaisussa käytettävät valaisimet ovat kehittyneet jo vuosia erittäin nopeasti. Tämän seurauksena myös suunnittelijoiden ilmaiskeinot ovat lisääntyneet laitteiden mahdollisuuksien myötä. Samaan aikaan varsinkin viihdeteollisuuden tuotantojen mittakaava on kasvanut merkittävästi yleisön vaatimusten kasvaessa ja teknisen kaluston kustannusten laskiessa. Valaistuksen ohjelmointi on työvaihe, jonka aikana järjestelmästä saadaan ulos taiteellisen suunnitelman mukainen lopputulos.

Suuri osa valojen ohjelmointityöstä tehdään virtuaalisessa ympäristössä, jossa operaattori voi työskennellä ilman oikeaa kalustoa ja siitä aiheutuvia kustannuksia. Siirryttäessä todellisen laitteiston pariin on operaattorilla usein varsin vähän aikaa saada kaikki toimimaan halutulla tavalla. Työn onnistuminen vaatii taidokasta valopöydän käyttöä, mutta ennen kaikkea hyviä työtapoja ja oikeita valintoja sen suhteen, miten eri asiat on konsolissa tehty.

Opinnäytetyössä käydään läpi valojen ohjelmointia konserttivalaisussa. Aiheen rajaamiseksi keskitytään GrandMA2-tuoteperheen parissa toimimiseen, mutta monet prosessit, työtavat ja tekniset seikat ovat sovellettavissa myös muihin laitteisiin. Opinnäytetyössä käsitellään etukäteisohjelmointia sekä sen siirtämisestä varsinaiseen ympäristöön, kerrotaan myös ongelmista, joita tähän joskus liittyy. Lukija saa työstä käsityksen ja kuvan yhdestä toimintatavasta, sekä sen haitoista ja hyödyistä.

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Degree Programme in Film and Television
Audiovisual Design of Theatre and Events

ANSAMI, ESKO
Programming of Concert Lighting in the GrandMA Environment

Bachelor's thesis 37 pages, appendices 0 pages
May 2016

Lighting fixtures used in theatre and event technology have been undergoing a rapid change over recent years. As a result, the ways designers can express themselves has been on increase as well. At the same time, the scale of entertainment productions such as television and concerts has grown massively as audience's expectations are growing simultaneously with the decreasing cost of technical equipment. Bringing artistic vision to a visible form requires skilled lighting programming.

Major part of lighting programming is done in a virtual environment, where an operator can work without the cost of having actual gear with him/her. When moving to the actual rig, everything has to be functional very quickly. Accomplishing such a task requires good knowledge of the console, but especially the right ways of working.

This thesis deals with the work of programming concert lighting. The focus is on GrandMA2-consoles, but many parts of the technical work flow described here are applicable to other consoles as well. The discussion includes pre-programming and moving it into the actual rig, potential problems that may occur along the way. The reader will get to know one way of working with its ups and downs.

Key words: lighting design, pre-programming, intelligent lighting, grandma2

SISÄLLYS

1	JOHDANTO.....	6
2	GrandMA2.....	7
2.1	GrandMA2-valokonsolin taustaa.....	7
2.2	Patch.....	7
2.3	Views	10
2.4	Programmer	11
2.5	Preset.....	12
2.6	FX	14
2.7	Cue.....	15
2.8	Sequence	16
2.9	Executor.....	16
2.10	Grand MA3D	16
3	ÄLYKKÄIDEN HEITINTEN OMINAISUUKSISTA.....	18
3.1	Ominaisuuksien jaottelusta	18
3.2	Attribuutit.....	19
3.3	Dimmer	20
3.4	Position	21
3.5	Gobo.....	22
3.6	Color	22
3.6.1	Color wheel	22
3.6.2	RGB ja CMY	22
3.7	Beam	23
3.7.1	Shutter	23
3.7.2	Prism ja Iris	24
3.8	Focus ja Zoom	24
3.9	Control ja Shapers.....	24
4	CASE: TAKOMO	26
4.1	Takomo	26
4.2	MA2-ympäristö valonohjauksen perustana	27
5	TAKOMO VIRTUAALISTI JA PROSESSINA	29
5.1	Ennakko-ohjelmoinnin lähtökohdat.....	29
5.2	Ennakko-ohjelmointi prosessina.....	29
5.2.1	Patch.....	29
5.2.2	Presetit.....	30
5.2.3	Sequence	31
5.2.4	Tracking	32

5.3	Livetilanne	33
5.4	Yhteenvedoa ennakko-ohjelmoinnin ja livetilanteen suhteesta.....	34
6	POHDINTA.....	36
	LÄHTEET.....	37

1 JOHDANTO

Opinnäytetyöni käsittelee valojen ohjelmointia. Lähtökohta on konserttivalaistuksessa, mutta samoilla tiedoilla ja taidoilla voi menestyksekkäästi ohjelmoida valaistuksen mi- hin tahansa esitykseen. Tarvittavien asioiden painotus voi olla toinen, mutta osaaminen on pätevää läpi esittävän taiteen kentän.

Valaisinten ohjaamiseen tarkoitettut valopöydät ovat nykypäivänä tietokonepohjaisia järjestelmiä, joiden laajuus saattaa yllättää käyttäjän. Useimmat konsolit tarjoavat mit- tavan valikoiman työkaluja erilaisten tarpeiden kattamiseen. Eri toimintojen hyödyntä- minen ja ymmärtäminen ulkoa muistamisen sijaan ovat usein avain tehokkaaseen ja onnistuneeseen työhön.

Kerron opinnäytetyössäni GrandMA2 konsolin toiminnoista, sekä niiden käytöstä omassa työssäni. Samaan lopputulokseen voi päästä monin tavoin, joten tekstin perusta- na on enemmän oma näkökulmani aiheeseen kuin totuus suuntaan tai toiseen. Tietope- rusta joidenkin teknisten seikkojen puolesta perustuu alan kirjallisuuteen. Konsolin käyttöön liittyvissä seikoissa viitataan tarpeen tullen verkosta löytyvään käyttöohjee- seen. Suurin osa työssä esitetyistä asioista perustuu kuitenkin omaan kokemukseeni ja mielipiteeseeni.

Opinnäytetyön case-esimerkkinä käytetään Takomo-yhtyeen konsertteja Electric Carne- val -konserttikiertueella. Baltic Queenilla ja Baltic Princessilla tehdyt neljä esiintymistä toimivat hyvinä esimerkkeinä ”peruskeikoista” niin ennakkotyön kuin varsinaisten keikkojenkin osalta.

Opinnäytetyöni on suunnattu kaikille asiasta kiinnostuneille, niin ammattilaisille kuin opiskelijoille. Se ei kuitenkaan ole minkään laitteen yksityiskohtainen käyttöohje, vaan yksittäisen käyttäjän näkökulmasta tehty läpileikkaus valojen ohjaukseen yhdellä tietyl- lä valopöydällä.

2 GrandMA2

2.1 GrandMA2-valokonsolin taustaa

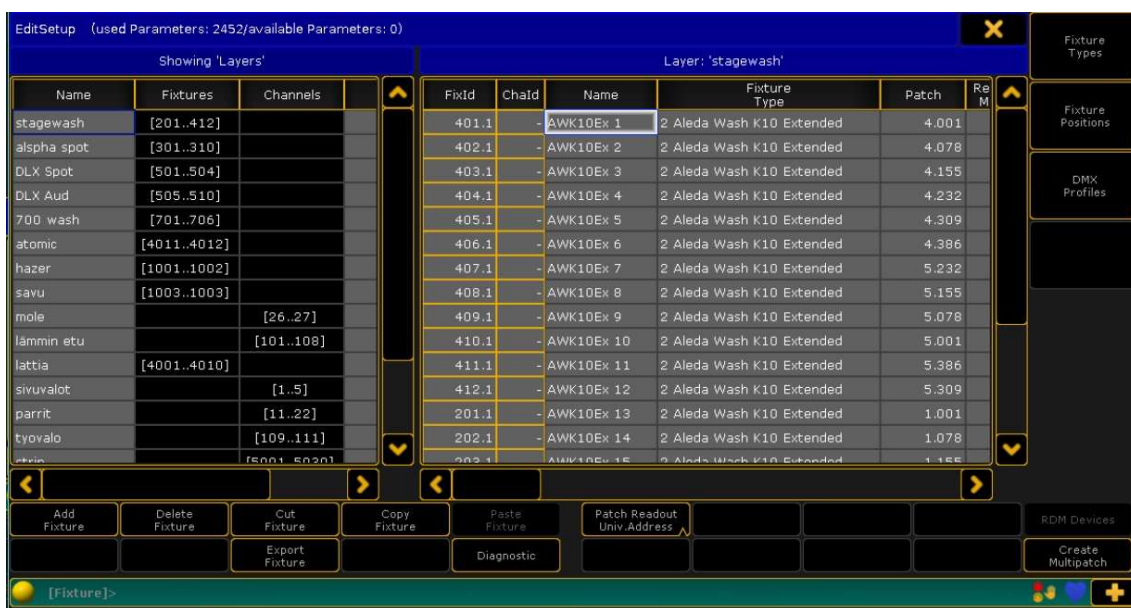
Estradivalaisussa valon hallintaan liittyvien työkalujen, kuten liikkuvien heittimienkin, historia on hämmästyttävän pitkä. Nykyaikaisten valopöytien historian voi kuitenkin sanoa alkaneen vuonna 1981, jolloin Genesis-yhtyeelle rakennetut ensimmäiset Vari-Lite-heittimet esiteltiin yleisölle (Cadena 2010, 18). Näitä heittämiä ohjanneessa konsolissa oli jo paljon yhtäläisyyksiä nykyisiin valopöytiin. Toinen vastaava merkkipaalu saavutettiin vuonna 1986, jolloin United States Institute of Theatre Technology (USITT) loi ensimmäisen DMX-standardin, yleisen ja avoimen protokollan ohjaimen ja valaisimen väliseen kommunikointiin. Standardin yleistyttyä valopöydät ja valaisimet kommunikoivat keskenään yhteisellä kielellä valmistajista riippumatta. Tämä edisti osaltaan laitteiden kehitystä eri yritysten keskittyessä kapeampaan alaan tuotekehityksessään. (Cadena, 2010, 277)

Tietokoneiden kehityksen myötä myös valopöydät siirtyivät nopeasti toimimaan tietokoneaudan päälle rakennetuilla järjestelmillä. Näin saatiin käyttöön laadukkaita, edullisia komponentteja ja paljon laskentatehoa. GrandMA2 on vuonna 1983 Saksassa perustetun MA Lighting-yrityksen luoma valo-ohjain. Se edustaa alan tämän hetken viimeisintä sukupolvea ja nauttii edeltäjänsä GrandMA:n tapaan suurta suosiota alan kentällä. (MA Lighting 2016)

2.2 Patch

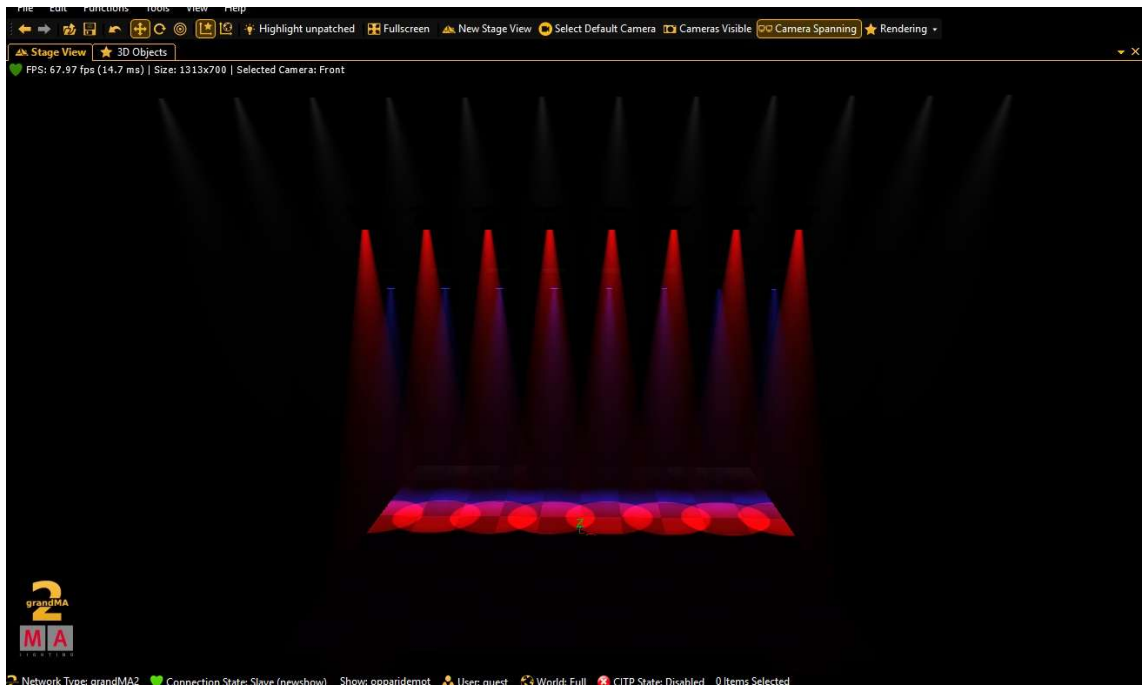
Patch & Fixture Schedule on paikka, jossa käyttäjä kertoo valopöydälle käytössä olevien laitteiden mallit ja DMX-osoitteet (Patch & Fixture Schedule: grandMA2 Help). MA hallinnoi Patch-näkymää Layer-tyyppisellä ajattelulla (kuva 1), joka antaa käyttäjälle runsaasti vaihtoehtoja käytössä olevan laitteiston organisointiin. Patch-näkymässä laitteelle annetaan myös Fixture ID, Channel ID tai molemmat. ID-numeroa voidaan verrata heittimien nimeen, sillä sen avulla heittämiä voidaan kutsua ohjattavaksi. ID-numerointi on oleellinen osa operaattorin työskentelyä ja perustuu monella tapaa jokai-

sen yksilöllisiin mieltymyksiin. Myöskään konsoli ei rajoita ID-numeroiden antoa juurikaan. Lähtökohtaisesti kuitenkin laitteille, joiden fixture type on himmennin, annetaan Channel ID, kaikille muille Fixture ID. Jotkut edistyneet heittimet pitävät sisällään ”useita heittämiä”. Esimerkiksi tässä opinnäytetyössä esillä olevat Clay Pakyn A.leda Washit koostuvat varsinaisesta heittäimestä, sekä 19 erikseen ohjattavasta RGB-pikselistä. Näin ollen valopöytä antaa Pääheittimelle ID-numeroksi xxx.1. RGB-pikselit seuraavat järjestyksessä perässä. GrandMA määrittää kunkin laitteen ohjaamiseen käytettävän heitinkirjaston laitteen sekä valmistajan mallin mukaan. Heitinkirjasto toimii tulkkina, jonka avulla käyttäjän antamat komennot ohjautuvat automaattisesti oikeille DMX-kanaville ilman, että käyttäjä joutuu asiaa erikseen ajattelemaan. Heitinkirjasto pitää sisällään tiedon heittimen tarvitsemista kanavista, niiden järjestyksestä ja kunkin laitteen ominaisuuksista. Kirjastot ovat käyttäjän hallittavissa, joten niitä voidaan muunnella tilanteen vaatimalla tavalla vaikkapa highlight-arvojen osalta. Erittäin käytökelpoinen on myös snap-valinta, joka estää käyttäjän asettamien fade-aikojen vaikutuksen valittuun attribuuttiin.

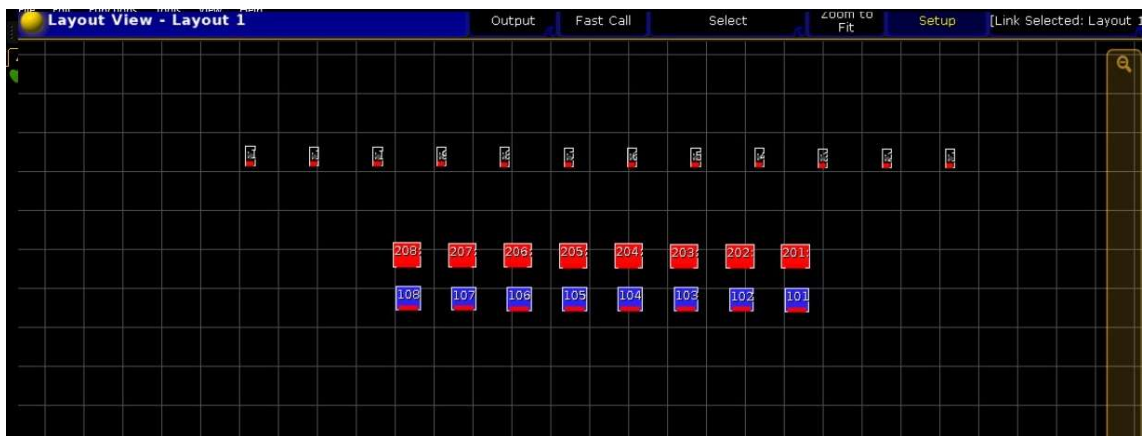


KUVA 1. Kuvassa käytössä olevat heittimet on jaettu eri layereihin heitintyyppin tai sijainnin mukaan. Heitinten ID-numerot on asetettu vastaamaan keikkapaikalta saatua valokarttaa. Kuvasta nähdään myös heittimien DMX-osoitteet. (Kuva: Esko Ansami)

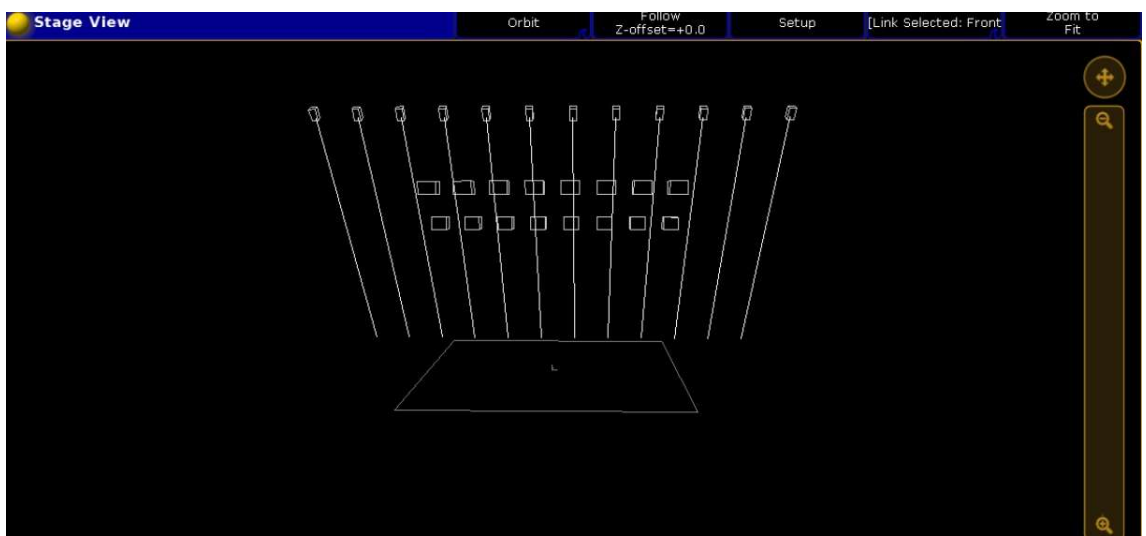
Fixtrue Schedule-näkymässä käyttäjä voi myös halutessaan asettaa laitteet todellisuutta vastaaville paikoilleen kolmiulotteisessa virtuaaliavaruudessa (kuvat 2a-c). Kerran tehty määrittäminen on myöhemmin käytettävissä monissa konsolin osissa, kuten MA 3D-visualisaattorissa, layout-näkymissä sekä Pixel Mapperissa.



KUVA 2a.



KUVA 2b.



KUVA 2c.

KUVAT 2a-c: Heitinten sijainnit 3D-visualisaattorissa sekä Stage- ja layout-näkymissä (Kuvat: Esko Ansami)

2.2.1 DMX-kanava ja parametri

Saadakseen valaistusjärjestelmän hallintaansa tarvitsee operaattori vaihtelevan määrän DMX-kanavia. Jos kanavien määrä on yli 512, tarvitaan uusi DMX-universumi. Kanavien kokonaismäärä riippuu järjestelmässä olevien heitinten tyypistä ja määrästä. Valopöydät pitävät monesti sisällään rajoituksia sitä koskien, montako universumia laitteesta on mahdollista ottaa ulos.

Lähes kaikki liikkuvat heittimet vaativat nykypäivänä joidenkin ominaisuuksien, kuten pysty- ja vaakaliikkeen toteuttamiseen kaksi DMX-kanavaa, joista toista kutsutaan tyyppillisesti fine-kanavaksi. Kahden kanavan käytöllä haetaan mahdollisimman tarkkaa hallintaa eri ominaisuuksiin. Suurissa järjestelmissä fine-kanavia saattaa olla monta sataa.

GrandMA2:ssa laskennan rajoitus ei perustu uloslähteviin DMX-kanaviin, tai DMX-universumeihin, vaan käytössä oleviin parametreihin. Tämä tarkoittaa käytännössä sitä, että vaikkapa liikkuvan heittimen pystysuuntainen liike on yksi parametri, vaikka sen hallintaan tarvitaankin useampi DMX-kanava. Esimerkkinä liikkuvasta heittäimestä voidaan ottaa vaikkapa Robe MMX Spot. Se tarvitsee toimiakseen laajimmillaan 40 DMX-kanavaa, joista fine-kanavia on 9 (Robe Lighting 2016). GrandMA2 Light-konsoli laskee ulos 4096 parametria. MMX-heittäimiä saadaan siis hallintaan pyöristettynä 132. Jos rajoitus olisi 4096 DMX-kanavaa, vastaava luku olisi 102.

2.3 Views

GrandMA2:n suuri vahvuus on käyttäjälle annettu mahdollisuus vaikuttaa itse siihen, mitä, missä ja miten on nähtävillä. Näin käyttäjä voi tehdä konsolista itselleen soveltuvan sen sijaan, että kaikki totuttelisivat samoihin näkymiin. Useat operaattorit ovat luoneet käyttöönsä käyttäjäprofiilin, jonka avulla on mahdollista tuoda omat, tutut näkymänsä mihin show fileen tahansa.

Näkymiä (Views) on konsolissa monenlaisia ja ne löytyvät kuuden eri välilehden alta. Sheets-välilehdeltä löytyvät näkymät muistuttavat värillisiä Excel-dokumentteja. Niiltä

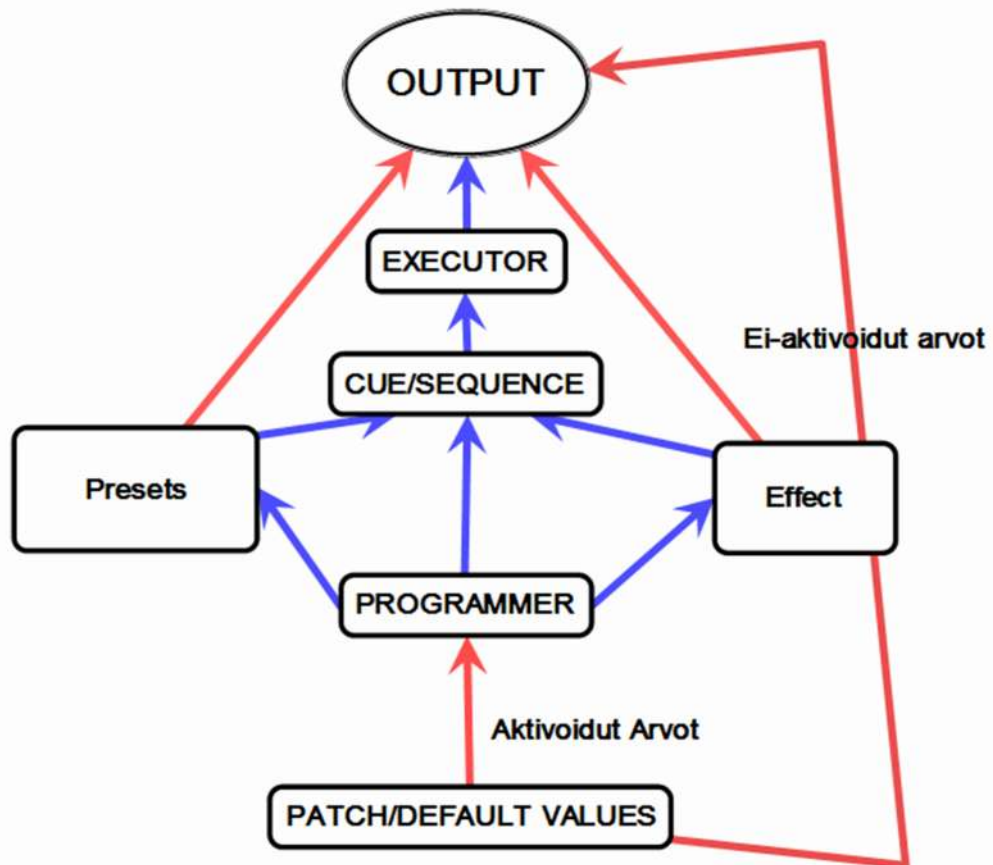
voidaan nähdä yksittäisen heittimen hetkelliset DMX-kanavat arvoineen millä tahansa hetkellä. Presets-ikkunat pitävät sisällään tiedot preseteistä, ja playback-ikkunoiden avulla voidaan hallita vaikkapa toisella executor-sivulla olevia kontrolleja kosketusnäytöltä. Kaikkia näkymiä yhdistelemällä ja tallentamalla operaattori voi nähdä tarvittavat asiat itselleen sopivalla tavalla, sopivassa järjestyksessä.

2.4 Programmer

Programmeri (Programmer) on valo-ohjainten oleellisimpia käsitteitä. Se tarkoittaa lyhyesti määriteltynä konsolin osaa, jossa heittimille voidaan syöttää arvoja ilman, että ne tulevat mistään tallennetusta valotilanteesta (Programmer: grandMA2 Help). Lähtökohtaisesti kaikki, mitä valopöydällä tehdään, kuuluu programmeriin. Jotta voidaan tallentaa valotilanne, on se mitä tahansa, se täytyy luoda ensin programmerissa.

Programmerilla käyttäjä tekee heittämiä koskevat valinnat, ja lähtökohtaisesti tallentaa lopputuloksen johonkin. Heittimien ominaisuudet siirtyvät programmeriin vasta, kun ne aktivoidaan. Ominaisuuksia voidaan aktivoida ilman, että valaistuksessa tapahtuu nähtäviä muutoksia.

Lähtökohtaisesti programmeriin aktivoidut arvot ja muutokset näkyvät näyttämöllä välittömästi. Niitä ei ole kuitenkaan tallennettu vielä mihinkään. Kuva 3 havainnollistaa arvojen kulkua konsolista heitimeen. Siniset viivat osoittavat tyypillisen tallennusjärjestyksen. Punaiset puolestaan sen, mitkä arvot milloinkin näkyvät näyttämöllä. Clear-toimintoa käyttämällä programmer tyhjenee ja suoraan heitinkirjastoista tulevat arvot palaavat voimaan. Clear-toiminto on kolmiportainen: ensimmäinen painallus poistaa heitinvalinnat ja toinen tyhjentää tallennukseen menevät asiat. Vasta kolmas painallus saa heittimet lavalla palaamaan lähtötilanteeseen.



KUVA 3. Arvojen kulku konsolista heittimeen (Kuva: Esko Ansami)

MA2-järjestelmässä avain programmerista eteenpäin on tyypillisesti store-näppäin, joka aktivoi tallennuksen käyttäjän valitsemaan kohteeseen. Tallentaa voidaan vaikkapa pre-settejä, efektejä, cue-tilanteita ja näkymiä. Perustilanteessa kaikki aktivoidut arvot lähtevät tallennukseen, mutta käyttäjä voi halutessaan rajata tallennuksen sisältöä eri keinoin.

2.5 Preset

Programmerissa aikaansaatuja asioita voidaan tallentaa preset-muotoon, jolloin pääsy kerran asetettuihin arvoihin on yhden napin painalluksen päässä. Vaikkapa positio, jossa jokainen näyttämön liikkuva heitin osoittaa lavan keskelle etureunaan, vaatii usein jokaisen lampun yksittäistä hallintaa ja on sitä kautta melko työläs toteuttaa. Kun tämä työ on kerran tehty ja valmis positio tallennetaan preset-elementiksi, päästään samaan näyt-

tämökuvaan tulevaisuudessa vaivattomasti aktivoimalla preset. GrandMA:ssa presettejä voidaan luoda mistä tahansa attribuutista tai useampien attribuuttien yhdistelmästä.

Kun preset-tietoa syystä tai toisesta muutetaan, muuttuvat kaikki sitä käyttävät valotilanteet automaattisesti mukana. Preset-elementin tarkoituksena onkin toimia viittauksena määrättyihin arvoihin sen sijaan, että arvot olisi kirjoitettu suoraan valotilanteeseen (Presets: grandMA2 Help). Lähtökohtaisesti on viisasta pyrkiä luomaan valotilanteita, jotka sisältävät mahdollisimman paljon preset-pohjaisia arvoja ja vastaavasti vähän suoraan tilanteeseen tallennettuja arvoja. Näin toimittaessa keikkapaikalla tulee tarkistaa vain preset-tiedon oikeellisuus. Tätä kautta valotilanteet tulevat näyttämään suunnitelluilta ilman, että niitä on tarpeen käydä läpi ja yksi kerrallaan muuttaa.

MA2:ssa lähtökohtaisesti päällä on preset-filter, eli vaikkapa väriarvoja ei voi tallentaa positio-attribuuttien preset-elementiksi. Halutessaan tämän ominaisuuden saa pois päältä. Vaikkapa focus-preset voi pitää sisällään useiden attribuuttien arvoja, kuten zoom ja focus, vaikka filter olisi päälläkin, koska focus ja zoom kuuluvat ohjaimen logiikassa saman preset-typen alle. Jos kuitenkin presetissä on alun perin aktivoitu vaikkapa vain focus-arvo, ja siihen myöhemmin lisätään mukaan zoom-arvo, ei uutena arvona mukaan tullut zoom-attribuutti päivity valotilanteisiin. Oikeat arvot tulee siis aktivoida jo presetiä luotaessa, vaikka ne myöhemmin tulisikin muuttaa.

Selective-presetit (kuva 4) voidaan aktivoida vain niille heittimille, joille kyseinen preset on alun perin tehty. Global Preset puolestaan voidaan aktivoida miten monelle heittimelle tahansa, kunhan preset on tehty vähintään yhdelle saman heitinkirjaston heittimelle. Universal preset voidaan aktivoida mille heittimille tahansa, mutta lopputulos voi olla odottamaton. Presettien avulla voidaan hallita myös highlight-ja default-arvoja. Asettamalla vaikkapa positio default-presetiksi pidetään huolta siitä, etteivät heittimet palaudu tyyppilliseen, yleensä epäedulliseen oletusasettoonsa kun programmer tyhjentään.



KUVA 4. Preset-ikkuna. Kuvassa selective presetit, kuten positiot, voidaan erottaa s- kirjaimesta presetissä. Global-presetissä vastaava kirjain on G. All-preseteissä ”high-beam”-presetti on embedded preset, eli sen teossa on käytetty muita presettejä. Kun käytettyjä presettejä päivitetään, päivitty embedded preset samalla. (Kuva: Esko An-sami)

2.6 FX

Valo-operoinnissa efektien luonne on moninainen ja tarkkaa määritelmää efektin käsit-teelle on hankalaa antaa. Se voisi kuitenkin olla esimerkiksi seuraavan lainen:

- Efekti on käyttäjän tahdon mukaisesti tapahtuva valitun attribuutin DMX-arvojen variointi
- Efekti kohdistetaan yhteen tai useampaan attribuuttiin, jonka jälkeen konsoli au-tomaattisesti alkaa muunnella vastaanottavien heittimien saamia DMX-arvoja.

Efektejä voidaan tallentaa omaan paikkaansa, Effect-pooliin, tai niitä voidaan ohjelmoi-da suoraan valotilanteisiin programmerilla. Effect-pooliin voidaan tallentaa kahden tyyppisiä efektejä, selektiivisiä ja template-efektejä. Selektiivisessä efektissä on sisään-kirjoitettuna tieto siitä, mille heittimille efekti on tehty, template-efekti taas voidaan kohdistaa millaiseen heitinvalintaan tahansa. Jos efektejä käytetään valotilanteissa, se-lektiivinen efekti käyttäytyy presetin tavoin, eli sen muuttaminen vaikuttaa automaatti-

sesti valotilanteeseen. Template-efekti puolestaan tulee aktivoida uudelleen ja valotilanne päivittää, jotta muutos tulee voimaan.

Efektin editointi-näkymän avulla käyttäjä pystyy hallitsemaan efektin toimintaa eri muuttujien, kuten korkeimman ja matalimman raja-arvon sekä nopeuden osalta. Oheisessa kuvassa oikealla nähtävät Phase From-, Phase to- ja Width, muokkaimet hallinnoivat sitä, millä tavalla efektin aaltoliikettä ohjataan useamman heittimen ryhmään. Efekti käsittelee heittämiä määrättyssä järjestyksessä. Järjestys on käyttäjän muokattavissa ja vaikuttaa visuaaliseen lopputulokseen monesti paljonkin. Tarvittaessa efektin käsittelyssä olevat heittimet voidaan jaotella eri tavoin ryhmiin, jolloin ryhmässä olevat valaisimet reagoivat identtisesti efektiin. Myös nämä työkalut löytyvät oheisesta näkymästä.



KUVA 5. Efektin editointi-näkymä (Kuva: Esko Ansami)

2.7 Cue

Cue on programmereihin syötettyjen arvojen yhdistelmä, joka tallennetaan saataville yhdellä yksinkertaisella komennolla. Periaatteellisesti ajatellen Cue on se valo-eroinnin lopputuote, joka lopulta näkyy yleisölle. Cuen sisältöä tai kestoa on vaikea käsitteellisesti määrittellä, mutta sen voidaan sanoa olevan yksi stabiili valaistuksellinen tilanne näyttämöllä. Cue voi pitää sisällään monenlaisia ajastuksia, eli siirryttäessä cuesta toiseen valotilanne voi elää. Kuitenkin, kun Cue on lopullisesti tullut voimaan, ei valaistuksessa tapahdu muutoksia. Poikkeuksen tähän luovat kuitenkin efektit, jotka saattavat tehdä cue-tilanteesta hyvinkin eloisan.

Esitys, on se mikä tahansa, vaatii monesti käyttöönsä useampia valotilanteita. Näin luodaan useamman cuen mittainen cuestack, valotilanteiden sarja jossa edetään esityksen sanelemassa tahdissa eteenpäin. Esimerkiksi rock-konsertti sisältää useampia kappaleita, joita varten valosuunnittelija voi luoda useita cuestackeja, jokaiselle kappaleelle omansa.

2.8 Sequence

Kun käyttäjä tallentaa uuden cuen mihin tahansa muualle kuin jo olemassa olevan cuelistan jatkeeksi, MA2 luo uuden sequencen. Se voi pitää sisällään miten monta cue-tilannetta tahansa. sequencet ovat yksilöllisesti numeroidut ja käyttäjä pääsee niihin käsiksi sequence-poolissa. Valmistajan käyttöohje vertaa sequencen ja cue-tilanteen välistä eroa cd-levyn ja levytä löytyvän kappaleen suhteeseen. Siinä, missä cue on yksi kappale, on sequence koko levy. (Cues & Sequences: grandMA2 Help.)

2.9 Executor

MA2:ssa valotilanteiden hallintaan käytettävät napit ja liu`ut koostuvat executoreista. Executorien määrä vaihtelee valopöydän mallin mukaan. Lähtökohtaisesti niitä on kahdenlaisia: toinen tyyppi pitää sisällään yhden liu`un ja kolme nappia, toinen tyyppi vain yhden napin. Käyttäjä voi käyttää executoreita mielensä mukaan, mutta tyypillisimmin niihin osoitetaan sequenceja. Normaalisti tämä tapahtuu jo siinä vaiheessa, kun käyttäjä luo sequencen. Jokaiseen executor-nappiin tai liukuun on mahdollista asettaa erilaisia toiminnallisuuksia. Lisäksi valitun sequencen hallintaan voidaan tarvittaessa osoittaa useampia Eecutoreita. Näin sequencen toimintaa saadaan hallittua hyvin tehokkaasti. Sequencejen lisäksi Eecutoreihin voidaan osoittaa vaikkapa Macroja ja Efektejä.

2.10 Grand MA3D

GrandMA 3D on MA2-tuoteperheeseen kuuluva osa, jonka voi ladata ilmaiseksi internetistä. Sen avulla voidaan ohjelmoida teoriassa miten isoja laitemääriä tahansa, eivätkä

valmistajan asettamat parametrirajoitukset päde siihen. Visualisaattori kytketään MA2-konsoliin MA2-Net-protokollan avulla.

Poiketen omina ohjelminaan myytävistä valosuunnitteluohjelmista, MA3D toimii itse asiassa konsolin lisäosana. Verrattuna varsinaisiin valosuunnitteluohjelmiin, joissa käyttäjä rakentaa valosuunnitelmansa pala palalta ohjelmaan, MA 3D toimii paljon suora-
viivaisemmin. Tiedot heittimistä, niiden sijainnista ja hallinnasta syötetään suoraan konsoliin, josta se peilautuu visualisaattoriin. Toisaalta isojen järjestelmien rakentaminen konsolin kontroleilla on monesti hitaampaa kuin perinteisissä valosuunnitteluohjelmissä.

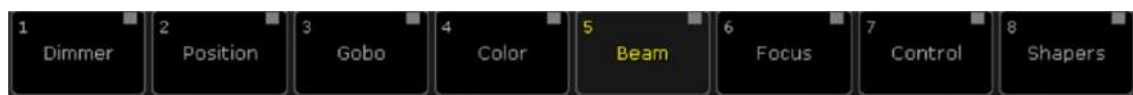
MA3D:n luotettavuuden suhteen kriittisessä asemassa ovat heitinmallit, jotka visualisaattorissa kullekin laitteelle osoitetaan. Ohjelma ei pidä sisällään kirjastoa eri heittimistä vaan käyttää muutamaa perusmallia kaikkien heitinten mallintamiseen. Niin ikään tieto DMX-kanavista kloonautuu suoraan valopöydästä.

3 ÄLYKKÄIDEN HEITINTEN OMINAISUUKSISTA

3.1 Ominaisuuksien jaottelusta

Älykkään heittimen määritelmänä pidetään yleisesti sitä, ettei heitin ole konventionaalinen, eli sitä hallitaan suoraan oman sisäisen koneistonsa avulla ulkoisen himmentimen sijaan. Nykyaikaiset valojärjestelmät perustuvat vahvasti liikkuviin heittämiin, mutta mukana on myös staattisia led-laitteita, stroboskooppeja sekä erilaisia matriisi-heittämiä, joiden hallinta perustuu yksittäisten pikselien ohjaukseen. DMX-kanavien tarve on nopeassa kasvussa, ja videon käyttö myös valojen hallinnassa lisääntyy jatkuvasti. Tämä kehitys haastaa myös valo-ohjainten suunnittelua uudella tavalla. Entistä suurempi määrä mahdollisia muuttujia on pidettävä järjestyksessä ja helposti hallittavissa niin, että laitteiden tehokas käsittely visuaalisen ilmaisun välineinä pysyy yhä työn keskiössä.

Älykkäiden heittimien ominaisuudet jaotellaan käytännössä kaikissa valopöydissä eri kategorioiden alle. Tämä helpottaa heitinten hallintaa ja varsinkin presettien (monet valopöytävalmistajat käyttävät nimitystä paletti) tekemistä. Tyypillisin tapa on neliosainen jako, jossa toisistaan eriteltyinä ovat Intensiiteetti, positio, väri sekä beamshape. Näin toimittaessa positiopaletteihin tallennetaan positiodata ja väripaletteihin väridata. Intensiiteeteille ei kaikissa konsoleissa ole palettimahdollisuutta lainkaan. Beamshape-paletteihin voidaan puolestaan tallentaa mitä tahansa beamshape-attribuutteihin kuuluvia arvoja. Näin ollen samassa paletissa saattaa olla arvoja vaikkapa strobo-, gobo-, prisma-, zoom- ja focus-kanavilta. GrandMA:ssa nelijaon tilalla on jako kahdeksaan eri preset-tyypeen (kuva 6). Kunkin preset-typen alta löytyy yksi tai useampia attribuutteja, joiden arvoja muuttamalla laite vastaa.



KUVA 6. GrandMA2:n preset-type-valikko (Kuva: Esko Ansami)

GrandMA2:ssa kuvan 7 osoittaman paletin tekeminen vaatisi preset filter-toiminnon ottamista pois päältä tai All-presetin tekoa. Siinä, missä kaikki kuvassa aktivoituneet arvot ovat Chamsysissa yhtä ja samaa beamshape-osiota heittimen ominaisuuksista, GrandMA:ssa ne löytyvät kolmen eri preset-typen alta. Shutter ja iris ovat samaa, samoin Go-

bo1 ja Gobo2 kuten myös zoom ja focus. Valopöytien paremmuudesta kiisteleminen ei ole tämän opinnäytetyön tarkoitus, joten voitaneen sanoa, että tässäkin tapauksessa sekä GrandMA:n että Chamsysin tapa on tilanteesta riippuen parempi tai huonompi. MA:n tavan hyväksi puoleksi voidaan kuitenkin laskea se, että yhden preset-typen alle kasautuu huomattavasti vähemmän attribuutteja, koska beamshape-attribuutit on jaettu monen preset-typen alle. Näin ikään ei-toivottujen arvojen tallentaminen preset-elementteihin lienee hieman harvinaisempaa, jos preset filter on päällä.

Head name	Head type	Hd no	Shutter	Iris	Gobo1	Gobo2	Focus	Zoom
MacViper	MacViper	1	Rnd Strob	Open	Rot Gobo	Open Gob	000	Wide
MacViper	MacViper	2	Rnd Strob	Open	Rot Gobo	Open Gob	000	Wide
MacViper	MacViper	3	Rnd Strob	Open	Rot Gobo	Open Gob	000	Wide
MacViper	MacViper	4	Rnd Strob	Open	Rot Gobo	Open Gob	000	Wide
MacViper	MacViper	5	Rnd Strob	Open	Rot Gobo	Open Gob	000	Wide
MacViper	MacViper	6	Rnd Strob	Open	Rot Gobo	Open Gob	000	Wide

Kuva 7. Beamshape-paletti Chamsys-valopöydässä (Kuva: Esko Ansami)

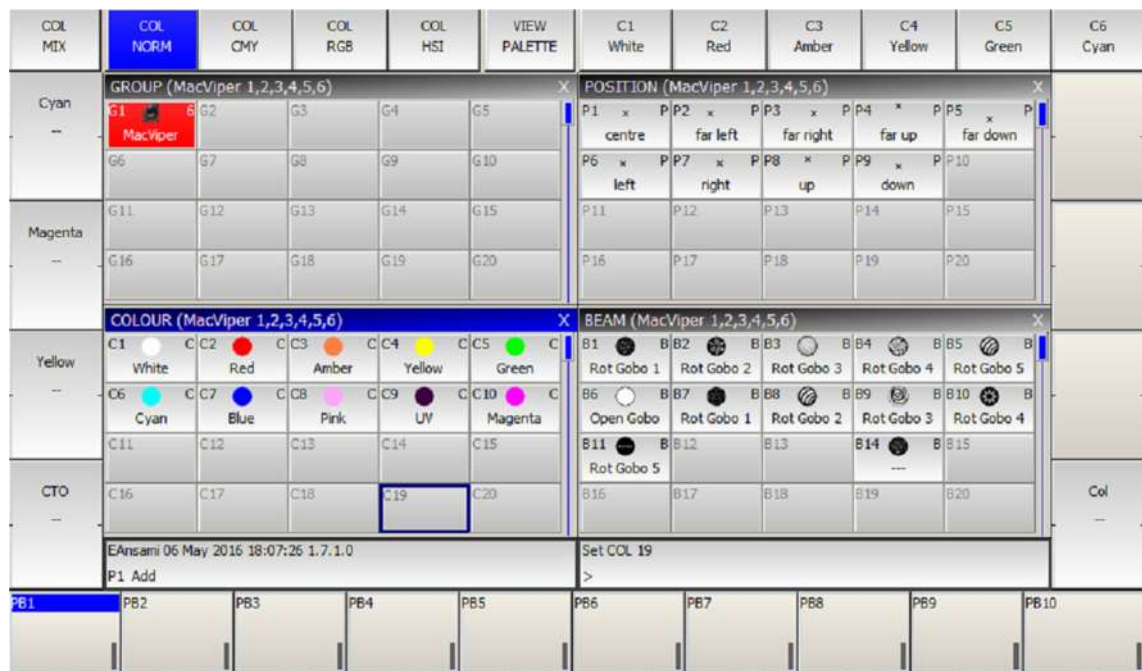
3.2 Attribuutit

Kun preset-type on valittu, saadaan näkyviin attribuutteja. Näitä säätämällä saadaan aikaan muutoksen näkyminen lavalla olevissa heittimissä. Monen preset-typen alla on useampia attribuutteja. Ne voivat olla toisistaan täysin irrallisia tai tarjota useita rinnakkaisia työkaluja vaikkapa valon värin hallintaan (kuva 8).



Kuva 8. Mac Viperia ohjattaessa GrandMA2 antaa Color-preset-typessä hallittavaksi kolme eri attribuuttia. Käsillä oleva värimiksaus vaatii käyttöönsä kolme fyysistä rullaa. (Kuva: Esko Ansami)

Mac Viperista löytyy kolme erilaista työkalua värin hallintaan. GrandMA:sta ne löytyvät kolmelta eri sivulta, Chamsysissa kaikki mahdollisuudet ovat näkyvissä kerralla (kuva 9).



KUVA 9. Chamsysin käyttöpinta saman asian hallintaan (Kuva: Esko Ansami)

Aiemmin mainittujen preset-typejen avulla konsoli jaottelee valaisinten ominaisuudet eri kategorioiden alle. Jako on siinä suhteessa käyttäjälle kiitollinen, että preset-typen alta löytyvät attribuutit useimmiten liittyvät toisiinsa tavalla tai toisella. Vaikkapa Color-preset-type pitää usein sisällään varsin kattavan määrän valintoja, koska värinmuodostus heittimissä tarjoaa monesti useita vaihtoehtoja. Vastaavasti esimerkiksi Dimmer-preset-type pitää sisällään tyypillisesti vain yhden mahdollisen muuttujan.

3.3 Dimmer

Himmennin, eli valon kirkkauden säätö on minkä tahansa laitteen perustavanlaatuinen ominaisuus. Himmennys voidaan toteuttaa varsin vaihtelevin tavoin. Jännitteen hallintaan perustuvan himmentimen käyttö rajoittuu käytännössä konventionaalisiin valonheittämiin ja muutamiin liikkuviin valoihin. Purkauslampuilla varustettua liikkuvaa heittintä ei voi kytkeä tavalliseen himmentimeen, koska sen muu elektroniikka hajoaisi jännitteen laskiessa liian alas. Liikkuvissa valoissa käytetään mekaanista himmennystä, eli valonlähteen ja linssin väliin sijoitettuja metallilevyjä. Näiden asentoa voidaan ohjata

portaittain ja siten hallita ulostulevan valon määrää (Cadena 2010, 242). LED-tekniikassa himmennys tehdään ohjaamalla tätä varten kehitettyä virtalähdettä (Cadena 2010, 217).

Ulostulevan valon kirkkaus on lukemattoman monen tekijän summa. Siihen vaikuttavat polttimon tehon lisäksi sen ikä, kunto sekä kaikki käyttäjän tekemät valinnat koskien heittimen optiikkaa ja väriä. Lisäksi se, miten kirkkaana katsoja valon kokee, riippuu ympäristöstä, käytettävän tehosteusvan määrästä ja muista valaistusolosuhteista. Etukäteisohjelmoinnissa valon intensiteetti voi erota oikeasta tilanteesta radikaalisti.

3.4 Position

Positiolla tarkoitetaan liikkuvan heittimen vaaka- ja pystyliikkeen arvojen yhdistelmää, joka saa valon osumaan määrättyyn kohtaan näyttämöllä. Sankaheittimistä löytyvät askel-moottorit liikkeille vaaka- ja pystysuunnassa. Visualisointiympäristössä position luotettavuuteen vaikuttavat ensisijaisesti kaksi tekijää. Ensimmäinen on laitteen sijoittuminen 3d-avaruudessa suhteessa reaalielämään. Jos heittimen on tarkoitus osoittaa vaikkapa 20 metrin heittomatkalta tarkasti määriteltyyn pisteeseen, vaikuttaa muutamankin sentin muutos heittimen sijainnissa merkittävästi valon osumakohtaan lavan pinnassa. Virhe kasvaa suorassa suhteessa heittomatkan kanssa. Kaiken kaikkiaan positioiden ohjelmoiminen etukäteen tarkalleen oikeiksi on useimmiten mahdotonta, koska yksittäisen heittimen tarkkaa sijaintia ei voi etukäteen määrittää riittävän tarkasti.

Toinen, merkittävästi isompia virheitä aikaansaava elementti on heittimen oletusarvojen ristiriita visualisoinnin ja todellisen heittimen välillä. Tämä tarkoittaa tyypillisesti sitä, että visualisoinnissa heittimen vaakaliikkeen lähtötilanne on väärä. Kyseinen tilanne johtaa siihen, että visualisoinnissa rakennetut positiot ovat yleensä 90 astetta väärin joko oikealle tai vasemmalle. Eli jos valon tulisi osua lavalle, se osuukin lavan sivuseinään.

3.5 Gobo

Gobo on valon tielle asetettava kuvio, jonka avulla saadaan projisointi (Cadena, 2004, 254). Gobot toimivat liikkuvissa valoissa samaan tapaan kuin värikiikko. Kiekkoon on valmistajan toimesta asetettu gobot, joista käyttäjä valitsee mieleisensä. Useissa laitteissa on kaksi gobokiekkoa. Osa goboista on indeksoivia, eli niiden asentoa on mahdollista säätää. Osa puolestaan on pyöriviä, joista saadaan efektiivinen lopputulos.

Värikiikon tapaan GrandMA2:ssa gobojen visualisointia voidaan säätää heitinkirjastossa. Jos vaikkapa teatterissa käytetään erikoisgoboja, jotka on vaihdettu heittimeen, saadaan sama tehty myös konsolissa. Näin virtuaalisessa ympäristössä aikaan saatava lopputulos on luotettavampi.

3.6 Color

3.6.1 Color wheel

Color wheel eli värikiikko on heittimestä löytyvien valmiiden värien kokoelma. Väri ei ole muokattavissa, mutta on monesti laadukkaampi kuin värimiksauksesta saatava lopputulos (Cadena, 2010, 251). GrandMA 3D:ssä värikiikossa olevat värit saadaan heitinkirjastosta, ja ne ovat myös muokattavissa tarpeen mukaan.

3.6.2 RGB ja CMY

LED-tekniikkaan kuuluu olennaisesti RGB-värijärjestelmä, jossa punaista, vihreää ja sinistä yhdistelemällä saadaan aikaan haluttu väri. Kaikkien kolmen ollessa maksimissaan saadaan teoriassa aikaan valkoinen. Koska valkoisen sävy on monesti käyttökelpoton, on kehitetty mm. RGBW ja RGBWA-värimiksaukset, joissa nimen mukaisesti on mukana myös valkoisia ja oransseja LED-elementtejä. Näin sävyistä saadaan tarkempia. CMY-mallissa lähtökohtaisesti värjätään valkoista valoa. RGB-mallissa valo itsessään syntyy värinmuodostuksen yhteydessä.

Virtuaalisten ympäristöjen kannalta LED-laitteiden värinmuodostus on ollut pitkään ongelmallista. Varsinkin MA 3D:ssä intensiteetin ja värin suhde toisiinsa aiheutti paljon tilanteita, joissa valon kiila ei mallintunut tietokoneelle lainkaan. Viime aikoina asiassa on tapahtunut huomattavaa kehitystä ja heittimien mallinnus on muuttunut luotettavammaksi.

CMY-värijärjestelmässä heittimen optiikkaan on rakennettu kiekot, jotka voidaan eri asennoissa ohjata valon eteen. Näin saadaan teoriassa aikaan mikä tahansa kuviteltavissa oleva väri. CMY:n käyttö vaikuttaa heittimen valotehoon vähentävästi erityisesti saturaatioiltaan korkeiden värien, kuten sinisen ja punaisen tapauksessa.

3.7 Beam

Beam pitää sisällään monia heittimen optisia ominaisuuksia käsitteleviä attribuutteja. Muut optiikkaa käsittelevät preset-tyypet, Gobo ja Focus, pitävät sisällään selkeästi rajatun attribuuttivalikoiman. Kaikki gobojen toimintaa koskevat valinnat liittyvät luonnollisesti toisiinsa, samoin zoom ja focus. Beam on preset-type, josta löytyvät edellä mainitun kahden osa-alueen ulkopuolelle jäävät attribuutit.

3.7.1 Shutter

Shutter eli suljin pitää sisällään tyypillisimmin heittimen strobo-ominaisuteen liittyvät valinnat. Tarvittaessa se voidaan myös sulkea, jolloin valoa ei tule läpi himmentimen arvoista riippumatta. Fyysisesti suljin löytyy useimmista sankiheittimistä. Joissain laitteissa himmennin on korvattu kokonaan sulkimella, jolloin valon intensiteettiä ei pysty portaittain säätämään. Joskus shutter-kanavalta löytyvät myös lampun muuhun käyttöön liittyvät komennot, kuten purkauslampun sytytys -ja sammutuskomennot.

3.7.2 Prism ja Iris

Prisma (Prism) on optinen väline, joka levittää valosta tulevan kiilan useampaan osaan, tyypillisimmin kahdeksaan osaan. Monesti prisman lisäksi heittimessä on myös prisman rotaatio, joka on oma attribuuttinsa.

Iris (Iris) on zoomin lisäksi työkalu, jolla kiilaa saadaan kapeammaksi. Tämän toiminnon lisäksi monet heittimet sisältävät erilaisia iiriksen toimintaan liittyviä efektejä.

3.8 Focus ja Zoom

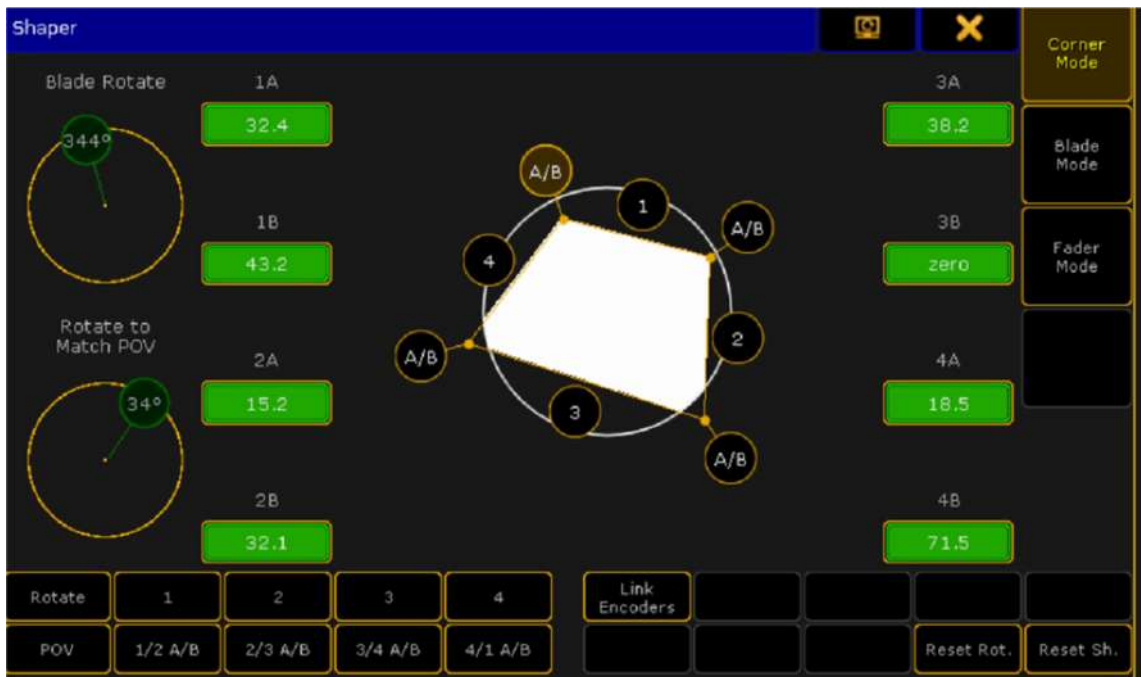
Focus säätelee kiilan teräväreunaisuutta ja myös projisoitavan gobon tarkkuutta. Sen toiminta riippuu suoraan zoom-attribuutin arvoista. Esimerkiksi seinään osoittavan gobo-projisoinnin kuva on siis terävä vain yhdellä tietyllä zoom- ja focus-attribuutin yhdistelmällä. Uudet liikkuvat heittimet tarjoavat monesti AutoFocus-toiminnallisuuden, jonka avulla focus seuraa zoom-kanavan muutoksia. Tällöin tulee kuitenkin tarkasti määrittää haluttu projisointietäisyys. Joissain tilanteissa focus ei toimi visualisointiympäristöissä lainkaan.

Zoom käsittelee kiilan leveyttä. Nykyaikaisissa heittimissä zoom on lineaarinen, eli portaattomasti säädettävissä. Vanhemmissa heittimissä on usein zoom-kiekkoo, jonka tarjoamista vaihtoehdoista käyttäjä valitsee haluamansa zoom-asetuksen.

3.9 Control ja Shapers

Control-attribuutin alta löytyvät laitteen ylläpitoon liittyvät komennot, kuten reset-käskyt. Niin ikään lampun sytytys ja sammutus tapahtuvat control-välilehdeltä.

Shapers hallitsee heittimessä olevien veitsirajainten toimintaa. Veitsien avulla saadaan tehtyä valokiilasta ympyrän sijaan vaikkapa puoliympyrä. Veitsien käyttö vaatii varsin paljon fyysisiä kontroleja, joten niiden käyttöön on luotu myös Special Dialog-näkymä (kuva 9).



Kuva 9. Shapers Special Dialog (Kuva: Esko Ansami)

4 CASE: TAKOMO

4.1 Takomo

Opinnäytetyöni case-esimerkkinä toimii työni Takomo-yhtyeen palveluksessa. Case-esimerkin kautta monet tekniset ohjelmointiin liittyvät valinnat ja käytännöt ovat helpompia esitellä ja perustella. Käyttämäni työtapoja ei ole tarkoitus väittää ainoiksi oikeiksi, eikä niin ikään parhaiksi mahdollisiksi. Kuvaus kuitenkin antaa osviittaa siitä, millaista valo-ohjelmointi oikeissa olosuhteissa on. Tätä kautta monet asiat selittävät itse itsensä, sekä tuovat kunkin ratkaisun hyviä ja huonoja puolia esiin. Case-esimerkissä käsitellään Electric Carnival-kiertuetta, joka piti sisällään neljä risteilyä kevättalvella 2016.

Takomo on vuonna 2008 perustettu yhtye, joka on ajan ja miehistönvaihdosten myötä päättynyt soittamaan elektronista musiikkia. Yhtyeelle ominaista on vahva painottuminen erityyppisiin lyömäsoittimiin. Olen työskennellyt yhtyeen kanssa ensimmäistä kertaa vuonna 2012 ja kiinteämmin vuoden 2015 alusta.

Takomon esiintymisen lähtökohdat poikkeavat jonkin verran tavallisten yhtyeiden vastaavista. Valosuunnittelun näkökulmasta huomattavaa on, ettei yhtyeessä ole ketään yksittäistä henkilöä, jonka läsnäolo toimisi visuaalisen ilmeen pohjana. Monesti konserttivalaisua helpottava ja osin ohjaavakin elementti on solisti, jonka valaisuun liittyvät ratkaisut ovat suuri osa konsertin kokonaisilmettä. Riippumatta siitä, miten demokraattinen ja tasavertainen yhtye on, muodostuu esiintymistilanteessa laulajasta usein keikan kantava voima ja vahvimmin läsnä oleva yksilö. Takomossa esiintyy neljä tasavertaista muusikkoa, ja asetelma on välitettävä tällaisena myös yleisölle. Toisaalta koko konseptin ideana on se, että yhtye soittaa keikalla kuultavan musiikin itse, vaikkakin erilaisia elektronisia välineitä käyttäen. Näin ollen esiintyjien on oltava järkevissä määrin näkyvissä. Tässä suhteessa ollaan lähempänä bändivalaisua kuin tyypillistä konemusiikin visuaalista suunnittelua.

Huomattavaa on myös, että musiikki ei kesken keikan taukoa käytännössä lainkaan. Kappaleiden vaihdot on sulautettu musiikkiin, joten taukoja ei synny. Näin ollen valoaakaan ei voi suunnitella kappale kappaleelta, vaan on kiinnitettävä tarkasti huomiota sii-

hen, miten kappaleiden välillä liikutaan teemasta toiseen. Paikoin keikassa saatetaan palata taaksepäin johonkin jo kuullun kappaleen osaan, niin että välissä on ollut jo jotain aivan muuta. Kokonaisuuden hallinta on siis haastavaa, mutta toisaalta elektronisen musiikin tematiikka vaatii valolliselta ilmaisulta myös näyttävyyttä hyvän draaman kaaren lisäksi.

EDM-musiikki noudattaa maallikkokuulijan tulokulmasta genretyyppejä lainalaisuuksia, joiden myötä musiikin suvannot ja kiivaammat osuudet ovat jokseenkin arvattavissa. Valollisen ilmaisun kannalta tämä johtaa helposti siihen, että tulee toistaneeksi samaa kaavaa noudattavat iskut monta kymmentä kertaa keikan aikana. Kuten todettua, kokonaisuuden hallinta on oleellista. Koko ajan ei voi ajaa täysillä, joten on tarkasti harkittava, millaisen kaaren esiintymiseen haluaa luoda. Vaikkapa tämän case-esimerkin keikoilla kalusto oli varsin laadukasta ja sitä oli riittävästi. Kuitenkin mikä tahansa laitteisto menettää tehonsa nopeasti, jos sitä käytetään liikaa tai väärin. Huomattavaa on myös, että tapahtumaa oli mennyt jo monta tuntia, kun oma esiintymisvuoro koitti. Yleisö oli nähnyt jo kaiken, mitä talon valopöydästä löytyi.

4.2 MA2-ympäristö valonohjauksen perustana

Takomon keikoista melko suuri osa ajetaan SMPTE-aikakoodilla. Tämä on jo havaittu hyväksi järjestelyksi monessa suhteessa. Paitsi, että aikakoodin avulla saadaan tarvittaessa hyvinkin tarkasti synkronoitua valotilanteet musiikkiin, sen edut myös monessa muussa kohtaa ovat kiistattomat. Ennen kaikkea keikan ohjelmointi on paljon nopeampaa, koska minun ei tarvitse välittää siitä, miten ohjelmoidut tilanteet saadaan ajettua ulos. Voin halutessani ohjelmoida päälistan tueksi tarvittavia iskuja tai muuta miten tahansa ja nauhoittaa ne aikakoodiin.

Keikkatilanteessa SMPTE pitää huolta iskujen oikea-aikaisuudesta, joten minun ei tarvitse murehtia muistini ja koordinaatiokykyni riittävydestä. Aikakoodi pitää huolen myös siitä, että tarvittaessa voin osoittaa itselleni tuuraajan, eikä hänen tarvitse tietää esitettävästä musiikista tai ohjelmoidusta show`sta käytännössä mitään. Kunhan hän päivittää presetit tarpeen mukaan oikeiksi, kaikki sujuu kuten pitääkin. Niitä keikkoja, joissa ei ole sitovaa taustanauhaa, ei voida ajaa aikakoodilla. Tällöin olen ohjelmoinut yhden pitkän pää-ajolistan, joka kattaa koko keikan.

Työhöni olen valinnut GrandMA2-järjestelmän, jonka eri versioita käytämme tarpeen mukaan. Kaikki versiot tukevat aikakoodia, joten valinta tehdään sen mukaan, paljonko parametreja pitää saada hallintaan. Muuten järjestelmät ovat toiminnaltaan identtisiä, joten mikä tahansa show saadaan pyörimään millä tahansa versiolla. Tähän mennessä käytössä ovat olleet onPC Command wing, sekä Light-konsoli.

5 TAKOMO VIRTUAALISTI JA PROSESSINA

5.1 Ennakko-ohjelmoinnin lähtökohdat

Koko konsertin mittaisia ajolistoja ei ole mahdollista ohjelmoida keikkapaikalla, joten työ on tehtävä ennakkoon. Jotta lopputulos olisi haluttu, on harkittava tarkasti, miten asiat etukäteen tehdään. Se, että valot saadaan toimimaan oikein tietokoneen näytöllä, ei vielä riitä. Eri elementit on tehtävä ja järjesteltävä niin, että niitä voidaan muuttaa halutulla tavalla myös keikkapaikalla.

Aikaa on lähtökohtaisesti aina vähän, joten on myös osattava ajatella, kuinka suuren määrän asioita ehtii paikan päällä testaamaan ja korjaamaan. Pitää myös varautua siihen, että paikan päällä ilmenee ongelmia. Niiden ratkomiseen menee helposti paljon aikaa, mutta keikka on silti saatava käyntiin ajallaan. Huolellinen ennakko-ohjelmointi ja riskien minimoiminen helpottaa operaattorin työtä monessa käänteessä.

5.2 Ennakko-ohjelmointi prosessina

Takomo teki käsillä oleville keikoille settilistansa uusiksi lähes tyhjästä, joten edessä oli uuden show`n ohjelmointi. Electric Carneval-keikoille valopöydäksi valikoitui Grand-MA2 Light-konsoli, koska odotettavissa oli liian suuria parametrimääriä onPC:n hallittavaksi. Sain melko aikaisin nauhoitteen yhtyeen harjoituksista. Sen avulla pystyin suunnittelemaan valaistuksen päälinjoja. Tekniset tiedot laivan kalustosta olivat varsin pitkään summittaisia ja tarkentuivat loppuun asti vasta viime hetkillä. Lisäksi oli tiedossa, että järjestäjän puolesta paikalle tuotaisiin myös lisäkalustoa, mutta sen yksityiskoh-
tia en saisi luultavasti tietää ennen keikkapäivää.

5.2.1 Patch

Suurin osa valo-ohjelmoinnista tehtiin ilman tarkkoja tietoja käytettävissä olevasta kalustosta. Tarkoitukseni oli tehdä ohjelmointi niin, että voisin tehdä varsinaisen ennakko-työn vain kerran. Koska tapahtumat olivat eri laivoilla, joutuisin joka tapauksessa kään-

tämään show`n kulloinkin käytettävissä oleville heittimille. Näin ollen ohjelmoin keikan alun perin huomattavasti suuremmille heitinmäärille kuin mitä todellisuudessa olisi käytössä. Näin voisin jättää keikkapaikalla patchiin vain sen määrän lamppuja kuin niitä todellisuudessa on ja poistaa loput. Kaikki toimisi tällä tapaa joka tapauksessa oikein.

Joillekin heittimille olin antanut jonkun tietyn tehtävän, kuten olla joissakin tilanteissa yksittäisen soittajan takavalona. Nämä heittimet merkkasin patchiin, niitä en poistaisi millään keikalla vaan katoisin keikkapaikalla, mikä heitin olisi sijainniltaan parhaalla paikalla ja määrittäisin sen oikeaan ID:hen. Tilaajan hankkimaa lisäkalustoa varten loin erilliset layerit, mutta niiden kanssa toimisin samalla tapaa kuin katossa olevienkin heittimien.

5.2.2 Presetit

Kuten todettu, presettien järkevä käyttö on yksi tärkeimmistä osa-alueista, kun valoja ohjelmoidaan etukäteen. Optimitilanteessa käyttäjä tietää tarkalleen, mitä mikäkin preset pitää sisällään, ja miltä sen halutaan näyttävän. Näin presettien käyttö on tehokasta ja isokin show saadaan nopeasti näyttämään oikealta.



KUVA 8. Ennako-ohjelmoinnin Preset-ikkuna (Kuva: Esko Ansami)

Alkuvaiheessa ohjelmoin presetit kuvan osoittamalla tavalla. focus-presettejä ohjelmoin kaksi viiden sarjaa. Jokaiseen presetiin on kirjoitettu zoom- ja Focus-arvo. Toiset on

tarkoitettu käytettäväksi silloin, kun gobo ei ole, toiset taas gobon kanssa. Näin saadaan oikea focus-arvo tilanteesta riippumatta.

Koska en tiennyt heitinten sijoittelusta etukäteen juuri mitään, ohjelmoin positiot hyvin viitteellisesti osoittamaan lavalle päin. Niin ikään tiesin, että gobo- ja focus-presetit tulisi joka tapauksessa päivittää. Lisäksi ennako-ohjelmointi tehtiin alun perin eri heittimillä kuin mitä vastassa tulisi todellisuudessa olemaan. Tein presetit vain sen verran tarkasti, että sain GrandMA 3D:ssä riittävän käsityksen siitä, miltä valo ehkä oikeassa elämässä tulisi näyttämään. Joka tapauksessa virtuaalisessa ympäristössä on pyrittävä pitämään mielessä, ettei mallinnus tarkimmillaankaan anna täydellistä kuvaa siitä, miltä tilanne tulee lopullisissa olosuhteissa näyttämään.

5.2.3 Sequence

Sequencejen määrään vaikuttaa paljolti se, miten kattava pääajolista on. Jos sinne on kirjoitettu aivan kaikki, teoriassa käytössä voi olla vain yksi sequence. Tällainen lähestymistapa on kuitenkin monesti melko epäkäytännöllinen. Jos yksitaiset iskut, usvakooneet, etuvalot ja yleisön huudatukset ohjelmoidaan kaikki samaan ajolistaan, kasvaa ajolistan pituus monikymmenkertaiseksi. Niin ikään vaikkapa etuvalojen intensiteettiä joutuu monesti säätämään lennossa muun valotilanteen mukaan.

Sequence pool on paikka, johon kaikki Sequencet tallentuvat, lähtökohtaisesti luomisjärjestyksessä. Sequence voi pitää sisällään yhden tai useamman valotilanteen, se voi olla myös osoitettu yhden tai useamman executorin hallittavaksi. Yksi sequence voi pitää sisällään vain yhdenlaista dataa, joten jos sama sequence on osoitettu kahteen executoriin, ne pitävät sisällään samat valotilanteita koskevat tiedot. Jos käyttäjä muuttaa sequencessa olevia tietoja, tiedot päivittyvät automaattisesti kaikkialle, missä sequence on käytössä. Jos sequence kopioidaan toiseksi sequenceksi sequence-poolissa, toimivat alkuperäinen ja kopioitu sequence jatkossa toisistaan riippumatta. Kuvassa 10 sequence-poolista löytyvät pää-ajolistan lisäksi yksittäisiä iskuja sisältävät sequencet, sekä työvalot.



KUVA 10. Sequence-pool (Kuva: Esko Ansami)

Itse pidän lähtökohtaisesti etuvalot kokonaan ulkona muita heittämiä koskevista ajolis-toista. Laitan ne aina johonkin oman liukunsa taakse. Kun takaa tulevan valon määrä vaihtelee radikaalisti, on etuvalo säädettävä sen mukaan. Paras lopputulos toki saataisiin säätämällä etuvalot valmiiksi kuhunkin tilanteeseen sopivaksi, mutta siihen ei ollut mitenkään mahdollisuuksia. Jos kuitenkin jossain vaiheessa keikkaa tarvitaan vaikkapa nopea blackout, sen kirjoitan pää-ajolistaan. Näin saan haluamani pimeyden yhdellä napin painalluksella, enkä unohda etuvaloja päälle. Niin ikään omina sequenceinaan pidän strobot ja yleisöön osoittavan blinderit, vaikka niihin liittyviä asioita saattaa olla ohjemoituna myös pääajolistaan.

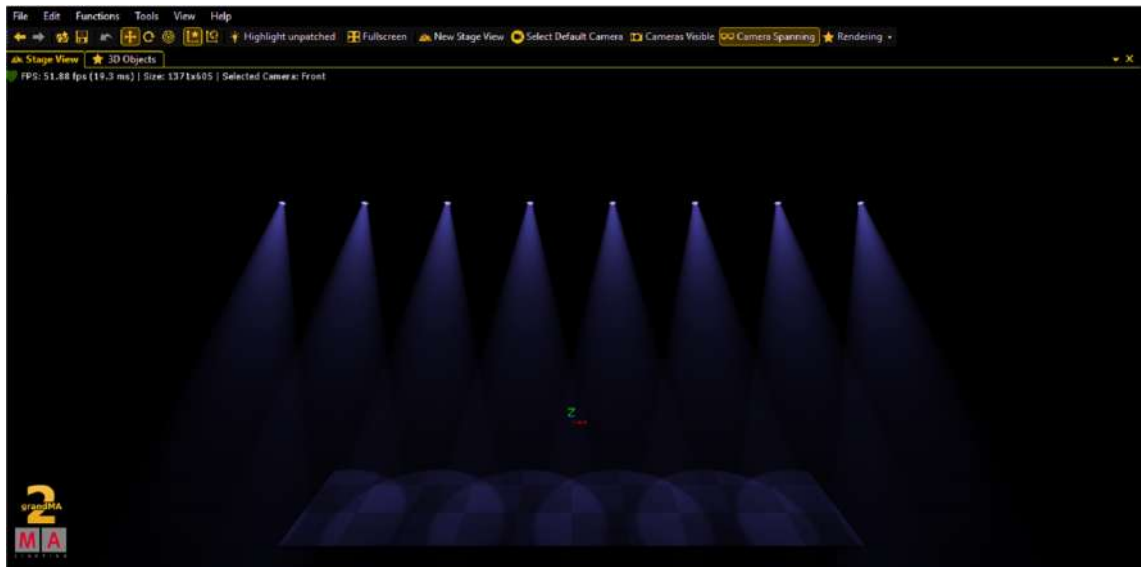
5.2.4 Tracking

Luotaessa pitkää ajolistaa valopöydän tärkeimpiä toimintoja on tracking. Sillä tarkoitetaan tapaa, jolla uusi tilanne tallennetaan vanhan tilanteen perään. Jos tracking on pääl-lä, ohjelmoidaan uuteen tilanteeseen vain se, minkä halutaan lavalla muuttuvan. (Schil-ler 2011, 12.) Kun edellisestä tilanteesta siirrytään seuraavaan, jäävät voimaan sellaiset arvot joihin uusi tilanne ei ota kantaa. Tämä vähentää ohjelmoinnin määrää valtavasti, mutta kysyy myös taitoa ja ymmärrystä trackingin toiminnasta.

Jotta tracking toimisi tehokkaasti ja loogisesti, on syytä ohjelmoida niin vähän kuin mahdollista, eli antaa trackingin toimia mahdollisimman paljon. Haasteellisuus ilmenee viimeistään silloin, kun yksittäisiä tilanteita pitää muuttaa tai siirtää paikasta toiseen. Tällöin on oltava tarkasti perillä siitä, mitä arvoja mihinkin on tallennettu (kuvat 11a-b).

Tracking		E: 1. 210		S: Sequ 1		Edit CueOnly		Readout Natural		Link Encoders		Selection Only		Link Selected		Auto Scroll		Feature Sort		Fixture Sort	
Number	Name	1: Dim	1: Pan	1: Tilt	1: C1	1: R	1: G	1: B	2: Dim	2: Pan	2: Tilt	2: C1	2: R	2: G	2: B	3: Dim	3: Pan	3: Tilt			
1	Cue 1		center	-36.2						center	-36.2										
2	Cue 2		center	-36.2	open	14.4	41.3	max		center	-36.2	open	14.4	41.3	max						
3	Cue 3	open	center	-36.2	open	14.4	41.3	max	open	center	-36.2	open	14.4	41.3	max	open	center	-36.2			

KUVA 11a.



KUVA 11b.

KUVAT 11a-b. Kuvassa Tracking-sheet ja visualisointi Cue-tilanteesta 3. Kuten nähdään, Cue 3 pitää sisällään vain intensiteetti-arvot, mutta trackingin ansiosta myös väri ja positio edellisistä tilanteista ovat voimassa. (Kuvat: Esko Ansami)

5.3 Livetilanne

Keikkapaikalle päästäessä ensimmäisenä kytketään oma pöytä talon valojärjestelmään. Tämä voi tilanteesta riippuen olla helppoa tai vaikeaa. Nykyaikaiset, verkkoprotokollaan perustuvat signaalinsiirtoformaatit ovat keskimäärin vaikeampia kuin fyysisten DMX-kaapelien kytkeminen pöydän ulostuloihin. Signaalien saaminen valoihin asti voi olla joskus haastavaa, ja vielä tämänkin jälkeen saattavat DMX-universumit mennä väärin paikkoihin. Jos paikalla on valmiiksi valopöytä, johon oma showfile voidaan ladata, osa ongelmista poistuu. Talon valopöydät voivat nekin olla kuitenkin hankalia. Ohjelmistoversiot saattavat poiketa siitä, jolla showfile on alun perin luotu, tai talon valopöydässä voi olla kiinni heittämiä, joita ei omassa tiedostossa ole. Tällaisissa tilanteissa MA:n partial show read-toiminnolla on mahdollista pitää talon pohjatiedostossa olevia elementtejä, vaikka lataakin laitteeseen omat tiedostonsa.

Kun oikea data saadaan oikeisiin lampuihin, yleensä nähdään varsin nopeasti, onko kaikki kunnossa vai ei. Oikeaa dataa saadessaan järjestelmä yleensä heilahtaa oletusarvoihinsa. Jos ongelmia on, sen yleensä huomaa vilkkumisesta tai heitinten epämääräisestä liikehdinnästä. Tällöin edessä on monesti työläs ja turhauttava vianetsintä. Ongel-

ma saattaa olla signaalin reitityksessä, omassa patchissa, heittimissä itsessään tai jossain muussa.

Jos ohjelmointi on tehty oikein presettejä älykkäästi käyttäen, ei ohjelmoinnin korjaus oikeaan tilanteeseen tuota juuri yllätyksiä. On sinänsä merkityksetöntä, kuinka lähelle tai kauas lopullista muotoa presetit on ennakkoon tehty, koska ne joka tapauksessa käydään läpi. Huomattavasti tärkeämpää on, että operaattori on presettejä luotaessa aktivoinut oikeat arvot, ja että oikeat presetit on tallennettu oikeisiin valotilanteisiin.

Itselleni, kuten ennalta saattoi olettaa, suurimpia haasteita aiheuttivat järjestäjän tilaama lisäteknikka, joka piti jollain keinolla saada omaan ohjelmointiin mukaan. Näistä kuitenkin selvittiin kunnialla. Muutamat MA2-järjestelmän toiminnot ovat verrattain harvoin käytössä mutta osoittautuvat erityisen käteviksi, kun keikkapaikalla pitää tehdä yllättäviä muutoksia. Erinomainen esimerkki on park-työkalu, jolla heitin saadaan nopeasti lukittua yhden tai useamman attribuutin osalta. Search & Replace-toiminnolla puolestaan voidaan nopeasti selvittää esimerkiksi, missä tilanteissa jokin yksittäinen preset on käytössä.

5.4 Yhteenvetoa ennako-ohjelmoinnin ja livetilanteen suhteesta

Huomattavasti vaikeampaa, kuin saada ohjelmointi menestyksekkäästi siirrettyä tietokoneen näytöltä oikeaan elämään toimivana, on itse visuaalisen suunnittelun tekeminen ja sen arvioiminen, miltä mikäkin tilanne tulee todellisuudessa näyttämään. Erityisen haastavaksi osoittautui pixelmapping-efektien käyttö itse keikoilla. Kyseisiä efektejä ajettiin Clay Pakyn A.Leda-washeisin. Teknisesti efektit näyttivät siltä miltä piti, mutta eri värien summautuminen yleisöön oli odottamattoman vaikeaa ennakoida. Pixelmapping osoittautui muutenkin haastavaksi työkaluksi, koska välillä tätä tekniikkaa hyödyntäviä heittämiä kloonattiin järjestäjän tuomiin lisäheittämiin, jotka olivat tavanomaisia Led Wash- heittämiä. Näin ollen lisäheittämiin piti kloonata sekä itse heitin, että yksi pikseli, jotta värit saatiin toimimaan oikein.

Heitinten käyttäytyminen vastasi ennakoitua kokonaisuutena arvioiden hyvin. Oikeat valot liikkuvat monesti todellisuudessa hitaammin, ja LED-valaisinten fade-aikoja jouuu monesti säätämään vielä paikan päällä. Nämä olivat itselleni ennalta tuttuja asioita, joten yllätyksiä ei niiden osalta ilmennyt.

Etukäteisohjelmointi osoittautui neljän konsertin aikana melko onnistuneeksi työvaiheeksi. Työtä ei tarvinnut koskaan aloittaa aivan täysin alusta, vaan alkuperäinen showfile taittui uuteen järjestelmään ja lopulta melko kivuttomasti. Varsinaisia ongelmia ei oikeastaan ilmennyt. Aikaa valopöydän kanssa meni maksimissaan 20 minuuttia, jota voi pitää kohtuullisena aikana. Tosin on huomioitava, että pohjatyöt, kuten uuden järjestelmän patch tehtiin etukäteen.

6 POHDINTA

Valojen ohjelmointi on prosessi, jossa taiteellisesta visiosta yritetään tehdä totta. Matkalla suunnittelijan päästä yleisön silmiin on monia mutkia, joissa visio voi kariutua tai kärsiä. Heittimet, tietokoneet, tai operaattorit eivät voi tehdä mitä tahansa. Tämä vaatii operaattorilta myös sosiaalisia taitoja. Niin ikään on ymmärrettävä käytössä oleva aika, jota ei yleensä ole rajattomasti. Yhden tilanteen tekemiseen ei voi aina käyttää tuntikausia aikaa, tulisi siitä millainen tahansa. Voikin sanoa, että valopöydän käyttö on työn helpoimpia osia. Kuten tästä opinnäytteestä ehkä käy ilmi, sekään ei välttämättä ole erityisen yksinkertaista. Kuten minkä tahansa teknisen laitteen kanssa toimiessa, tärkeintä on pitää kirkkaana mielessä, mitä on tekemässä ja miksi, mikä on työn lopullinen päämäärä. Usein minkä tahansa monimutkaiselta tuntuvan asian tekemiseen on käyttökelpoisempi tapa, jos maltaa ajatella.

Opinnäytetyössä on käsitelty GrandMA2:n toimintaa omasta näkökulmastani, omiin käsityksiini ja työtapoihini perustuen. Lopputulokseen voi monessa suhteessa olla jокseenkin tyytyväinen, olkoonkin että aihetta on jouduttu rajaamaan paljon. Vaikka koko teksti käsittelee pääsääntöisesti yhtä tietokonetta, on sekin aiheena varsin laaja. Verrattuna alkuperäiseen suunnitelmaan opinnäytetyö on täysin toinen kuin se, mitä lähdin alun perin tekemään. Koen kuitenkin, että näin lopputulos on ansiokkaampi ja paremmin jäsennelty. Tutkimusta aiheen tiimoilta voisi helposti jatkaa moneenkin suuntaan. Asiantuntijahaastattelut antaisivat aiheeseen ainakin uusia näkökulmia ja allekirjoittanutta paljon vahvempaa ammattitaitoa. Niin ikään selvitys siitä, miten visualisoinnin luotettavuutta voisi parantaa, mikä itse asiassa oli alkuperäinen visioini tästä opinnäytetyöstä, toisi alalle paljon uutta, käyttökelpoista tietämystä. Teknisten laitteiden käyttöä käsittelevän kirjallisuuden elinikä on monesti lyhyt, joten tämäkään opinnäyte ei välttämättä ole ajankohtainen kovinkaan montaa vuotta.

Opinnäytetyön tekeminen oli pitkä projekti, jonka sovittaminen muiden kiireiden sekaan tuntui monesti mahdottomalta tehtävältä. Teknisten yksityiskohtien selventäminen kirjalliseen muotoon oli kirjoitusprosessin haastavimpia tehtäviä. Olen kuitenkin tyytyväinen lopputulokseen, ja uskon että se tarjoaa aiheesta kiinnostuneille luettavaa, ehkä myös jotain uutta tietoa.

LÄHTEET

Cadena, R. 2010. Automated Lighting: The Art and Science of Moving Light in Theatre, Live Performance, and Entertainment, 2nd edition. Burlington, Oxford: Focal Press publications.

Robe Lighting. luettu 22.5.2016

http://www.robe.cz/fileadmin/robe/downloads/dmx_charts/Robin_MMX_Spot_DMX_charts.pdf

MA Lighting. 2016 Luettu ja tulostettu 12.5.2016

Patch & Fixture Schedule. grandMA2 Help Luettu ja tulostettu 8.5.2016

<https://help.malighting.com/view/>

Programmer. grandMA 2 Help. Luettu ja tulostettu 8.5.2016

<https://help.malighting.com/view/>

Presets. grandMA2 Help Luettu ja tulostettu 8.5.2016

<https://help.malighting.com/view/>

Cues & Sequences. grandMA2 Help. Luettu ja tulostettu 8.5.2016

<https://help.malighting.com/view/>

Schiller, B. 2011. Automated Lighting Programmer`s Handbook, First Edition. Focal Press publications.