

Janne Saarela

LIVESTREAM
MONIKAMERATUOTANNOSSA
KASSUN KARNEVAALEILLA

Opinnäytetyö
Tietotekniikan koulutusohjelma


Joulukuu 2011




MIKKELIN AMMATTIKORKEAKOULU

Mikkeli University of Applied Sciences

KUVAILULEHTI

 <p>MIKKELIN AMMATTIKORKEAKOULU Mikkeli University of Applied Sciences</p>		Opinnäytetyön päivämäärä 2.12.2011
Tekijä(t) Janne Saarela		Koulutusohjelma ja suuntautuminen Tietotekniikan koulutusohjelma, mediatekniikka
Nimeke Livestream monikameratuotannossa Kassun karnevaaleilla		
Tiivistelmä 7. - 10.9.2011 järjestettiin Mikkelin ammattikorkeakoulun piha-alueella Kassun karnevaalit. Kassun karnevaalit -tapahtuma oli osa Mikkelin ammattikorkeakoulun 20-vuotisjuhlia. Tässä tapahtumassa toteutettiin opiskelijatyönä monikameratuotanto, josta lähetettiin Campustv.fi-websivustolle livestreamvideo. Monikameratuotannon suunnittelutyö aloitettiin kahta viikkoa ennen varsinaista tapahtumaa. Suunnitteluun kuului muun muassa tuotantokäsikirjoituksen laatiminen. Tämä opinnäytetyö avaa streamtekniikoita sekä monikameratuotannon toteuttamista. Streamtekniikoiden osuus on käsitelty opinnäytteen teoriaosuudessa mm. streamformaattien ja -protokollien kautta. Käytännön osuus taas koostuu monikameratuotannon raportoinnista, sekä siellä käytetyistä tekniikoista ja livestreamin käytännön toteutuksesta. Projektissa kävi ilmi suunnittelun ja tekniikan testaamisen tärkeys. Lopputuloksena tapahtuma ja siitä toteutettu monikameratuotanto sekä livestream onnistuivat hyvin.		
Asiasanat (avainsanat) streaming, videolähetys, monikameratuotanto		
Sivumäärä 33 s.	Kieli suomi	URN http://www.urn.fi/URN:NBN:fi:amk-2011111714751
Huomautus (huomautukset liitteistä) Liite 1		
Ohjaavan opettajan nimi Reijo Vuohelainen		Opinnäytetyön toimeksiantaja Mikkelin ammattikorkeakoulu

DESCRIPTION

 <p>MIKKELIN AMMATTIKORKEAKOULU Mikkeli University of Applied Sciences</p>		Date of the bachelor's thesis 2 December 2011
Author(s) Janne Saarela	Degree programme and option Information Technology, Media Technology	
Name of the bachelor's thesis Live stream video as a part of multi-camera production for the Kassun karnevaali event		
Abstract The Kassun karnevaalit event was held from 7 to 10 September 2011 at the grounds of Mikkeli University of Applied sciences as a part of the Universitys 20th anniversary celebration. As a student project we organised a multi-camera production. A Part of the production was a live video stream which was shown on a web page called Campustv.fi. The planning of the multi-camera production began two weeks before the Kassun karnevaali event. The final plan for the multi-camera production included, for example, a manuscript for the production. The purpose of this bachelor's was to survey the techniques behind a live stream video and the execution of a multi-camera production. The techniques were introduced in the theory part of the bachelor's, including, for example, the streaming protocols and formats. The more practical part of the thesis was a report on the multi-camera production of the Kassun karnevaali event and the techniques used during the production as well as the practical execution of the live stream video. Project showed the meaning and the importance of precise planning and testing of the equipment beforehand. All in all the event as well as the multi-camera production and the live stream went well.		
Subject headings, (keywords) streaming, video broadcast, multi-camera production		
Pages p. 33	Language finnish	URN http://www.urn.fi/URN:NBN:fi:amk-2011111714751
Remarks, notes on appendices Appendice 1		
Tutor Reijo Vuohelainen	Bachelor's thesis assigned by Mikkeli University of Applied Sciences	

SISÄLTÖ

1	JOHDANTO	1
2	STREAMAUS KÄSITTEENÄ.....	2
2.1	Live-streamaus.....	3
2.2	On-demand streamaus	3
3	STREAMAUKSEN TEKNIikka	4
3.1	Kompressointi.....	6
3.2	Sovelluserroksen QoS -hallinta	6
3.3	Streaming-palvelimet.....	13
3.4	Protokollat.....	15
3.4.1	Kuljetusprotokollat	16
3.4.2	Istunnonhallintaprotokolla	18
3.5	Vastaanottajan tekniikka.....	18
3.6	Formaatit.....	19
3.7	Kassun karnevaaleilla käytetyt sovelluserroksen ohjelmat	21
3.7.1	Anystream Agility.....	21
3.7.2	Wowza Media Server.....	22
4	LIVESTREAM KASSUN KARNEVAALEILLA	22
4.1	Käytetty laitteisto.....	23
4.2	Tuotannon suunnittelu	23
4.3	Tuotannon testaus ja rakennus.....	24
4.4	Tuotannon kulku ja purku.....	26
4.5	Johtopäätökset.....	30
5	YHTEENVETO	31
	LÄHTEET	34
	LIITE/LIITTEET	
	1 Tuotantokäsikirja Kassun karnevaalit	

1 JOHDANTO

Stremauksen käyttämisen mahdollisuudesta, videon ja äänen toimittamiseksi katsojalle tai kuuntelijalle, kerrottiin julkisuudessa ensimmäistä kertaa vuonna 1989. Vastaanottajan ei tarvinnut enää ladata koko ohjelmaa katsoakseen vain lyhyen pätkän videon alusta. Vastaanottajalaite eli tietokone tulisi tulevaisuudessa esittämään vastaanotettua dataa nopeammin, kuin katsoja voisi sitä katsoa. Tuolloin tätä tekniikkaa kutsuttiin ”puskuroiduksi mediaksi”. Vasta vuonna 1993 nähtiin ensimmäinen videostreamsovellutus, jonka jälkeen alkoi ilmestyä Internet-radioita sekä ensimmäiset livevideostreamit. (Wikipedia 2011c.)

Nykyään oman videostreamin lähettäminen Internetiin on erittäin helppoa, eikä siihen tarvita kovinkaan suuria tai nopeita laitteita. Videostreamin lähettäminen onnistuu jopa matkapuhelimella. Stream pitää sisällään kuitenkin monimutkaisia ja monisyisiä tekniikoita, joita tämän opinnäytetyön teorian osuuden on tarkoitus avata.

7. – 10.9.2011 järjestettiin Mikkelin ammattikorkeakoulun piha-alueella Kassun karnevaalit. Kassun karnevaalit –tapahtuma oli osa Mikkelin ammattikorkeakoulun 20-vuotisen taipaleen juhlistamista. Tämän opinnäytetyön käytännön osuuden tarkoituksena on selvittää minkälaisia työvaiheita vaati Kassun karnevaaleilla oppilasprojektina toteutettu monikameratuotanto ja siitä lähetty, campustv.fi-sivustolla näkynyt livestream.

Aihetta tarkastellaan nimenomaan streamingtekniikan näkökannalta. Teoriaosuudessa avataan tarkkaan livestreamin takana olevaa tekniikkaa muun muassa streamauksen protokollia sekä yleisimmin käytettyjä streamformaatteja. Mukaan on poimittu perus teknisen tiedon lisäksi uusinta tietoa esim. erilaisista kotikäyttäjän multimediajärjestelmistä.

Opinnäytetyön käytännön osuus koostuu tarkasta raportista Kassun karnevaalien monikameratuotannosta ja sieltä tehdystä livestreamista. Mukana on muun muassa toteutetun streamauksen arkkitehtuurin ja monikameratuotannon videokaaviot. Raportoinnissa on pureuduttu erityisesti siihen, mitkä tuotannon osa-alueet onnistuivat hienosti, ja missä olisi ollut parantamisen ja tarkentamisen varaa.

2 STREAMAUS KÄSITTEENÄ

Streamaus eli suoratoisto on multimediatiedostojen siirtotapa, jossa siirrettävän tiedoston sisältöä voidaan aloittaa katsomaan heti tiedoston latauksen aloittamisen jälkeen. Tekniikalla voidaan toki siirtää mitä tiedostoja tahansa, mutta yleensä sitä hyödynnetään multimediatiedostojen jakamiseen. Tiedostoa ei siis tarvitse ladata kokonaisuudessaan tietokoneelle, ennen kuin sitä voi katsella. Tiedosto latautuu loppuun sillä välin, kun sitä jo katsotaan. Streamausta käytetään yleensä musiikkitiedostojen ja videotiedostojen esittämiseen sekä live-lähetyksiin, kuten rock-konsertteihin tai seminaareihin. (Keränen ym. 2003, 77.)

TV- ja radiolähetykset muistuttavat periaatteessa streaming-tekniikkaa, mutta eivät sitä kirjaimellisesti ole, vaikka lähetyksiä voidaan seurata samalla aikaa, kun ne siirtyvät vastaanottiin. Puhtaasti streaming-tekniikkaan perustumattomia tallenteita ovat esimerkiksi lehdet, cd- ja dvd-levyt. Erona perinteisiin tallenteisiin ja tapoihin katsella mediaa, kuunnella radiota, lukea lehtiä tai katsella videoita on se, että streaming-mallia käyttäen käyttäjä voi vaihtaa vähällä vaivalla streamista toiseen. Tämä on lähes yhtä vaivatonta kuin kanavanvaihto televisiosta. (Topic 2002, 11.)

Analogisten medioiden kopioinnissa laatu kärsii aina vähän toisella tapaa kuin digitaalisessa mediassa. Näiden virheiden korjaaminen ei ole yhtä kätevää, kuin digitaalisen median streamauksessa. Digitaalinen media kulkee eri siirtoteiden kautta pieninä paketteina, ja jos yksikin paketti häviää matkalla bittiavaruuteen, videon toisto pysähtyy ja kyseinen paketti ladataan uudestaan. Streamin laatu riippuu siis aika paljon nettiyhteyden laadusta ja nopeudesta. Puskuroinnilla tätä ongelmaa voidaan lieventää, mutta tämä muistuttaa perinteistä tapaa ladata tiedosto kokonaan kiintolevyille ennen toistoa, jolloin streaming-tekniikan suoma etu valuu hukkaan. (Topic 2002, 12-13.)

Streamausjärjestelmät koostuvat yleensä seitsemästä osasta. Raakamateriaali, esimerkiksi video ja ääni, esikompressoituaan, jonka jälkeen data tallennetaan johonkin tallennuslaitteeseen. Käyttäjän pyynnöstä stream-palvelin hakee kompressoitua datan tallennuslaitteesta. Tämän jälkeen sovelluskerroksen QoS-hallintamoduuli muuntaa audio- ja videobittivirrat verkon tilan ja QoS-vaatimusten mukaiseksi. Tämän jälkeen kuljetusprotokollat ”paketoivat” bittivirrat, ja lähettävät

paketit Internetiin. Internetissä paketit voivat hävitä tai viivästyä. Vastaanottajalle onnistuneesti toimitetut paketit menevät kuljetusprotokollien läpi sovelluskerrokselle prosessoitavaksi, minkä jälkeen audio ja video puretaan erikseen omien dekooderiensa toimesta. Jotta ääni ja video saadaan tahdistetuksi, tarvitaan jonkinlaisia median synkronointimekanismeja. (Wang ym. 2002, 520-521.)

2.1 Live-streamaus

Live-streamaus tarkoittaa sitä, että streamattava lähdemateriaali tapahtuu reaaliajassa. Yleisimmät livestreamin käyttökohteet ovat rock-konserttien streamaus suorana tai vaikkapa nettiradiot. Näissä suorissa lähetyksissä katsojille lähetetään aina samaa tiedostoa, eikä tiedostoa pysty kelaamaan tai keskeyttämään ja jatkamaan samasta kohdasta. (Topic 2002, 18-19.)

Nykyään live-lähetyksen nettiin laittaminen on tehty helpoksi. Netissä on erilaisia ilmaisia livestream-palveluita, joiden avulla kuka vaan webkameran omistava henkilö voi julkaista omaa livelähetystään netissä. Kamera voi olla vaikka läppärin kiinteä webkamera tai internetyhteydellisen puhelimen kamera. Palveluiden kautta lähetettyjä videoita voi tallentaa myöhemmin katsottavaksi ja jakaa erilaisissa sosiaalisissa nettiyhteisöissä, kuten Facebookissa ja Twitterissä. Tällaisia livestream-palveluita ovat esimerkiksi ruotsalainen Bambuser, Justin.tv ja Ustream. Palveluilla on miljoonia käyttäjiä. (Bambuser 2011; Justin.tv 2011; Ustream 2011.)

2.2 On-demand streamaus

Nykyään todella suosittu tapa nauttia videoista streamaustekniikan avulla on on-demand streamaus, josta tunnetuin esimerkki on Youtube.com-sivusto. Ero livestreamiin on siinä, että mitä tahansa saatavilla olevaa videota voidaan katsoa silloin kuin halutaan. Videon voi pysäyttää ja jatkaa hetken kuluttua samasta kohdasta tai sitä voi kelata ja hyppiä kohdasta a kohtaan b. Valmis video ladataan streaming mediaserverille. Kun käyttäjä haluaa katsoa jonkun videon, hänen koneensa ottaa yhteyden mediaserveriin ja lähettää pyynnön halutusta videosta. Mediaserveri

käsittelee yhteydenoton ja pyynnön, ja lähettää halutun videostreamin käyttäjän mediatoistimeen. (Topic 2002, 26.)

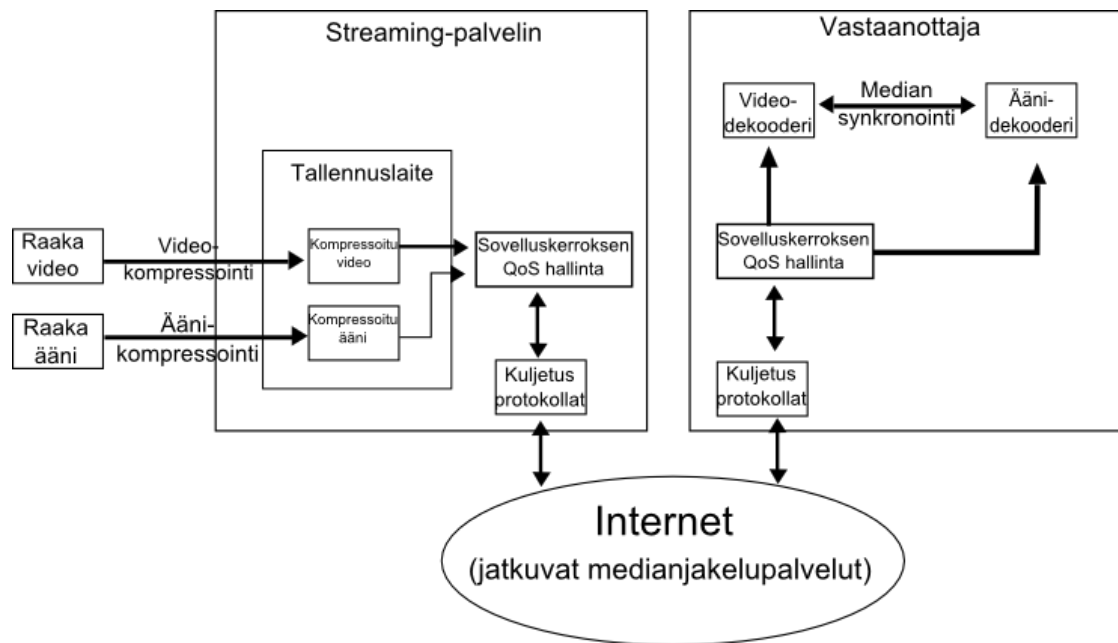
On-demand streamauksen etuna on se, että videon lähetystä ei tarvitse aikatauluttaa mitenkään. Videotiedosto on palvelintietokoneessa odottamassa, kunnes joku sen haluaa katsoa. Streamia ei tarvitse mainostaa eikä sen tarvitse saavuttaa suuria määriä yleisöä tietyllä ajanhetkellä. Tämä mahdollistaa myös sen, että videomateriaalia voidaan tehdä myös hieman erikoisemmista aiheista. (Topic 2002, 26.)

Digitaalisten käyttöoikeuksien hallinta mahdollistaa maksujen keräämisen streamista sekä pääsyn hallinnan. Käyttöoikeuksien haltija voi vaikkapa valtuuttaa videon katsojalle tietyn määrän toistokertoja tai mahdollisuuden videon tallentamiseen. Toisaalta streamin tallennusmahdollisuus muuttuu turhaksi ominaisuudeksi, jos streamin käyttömaksu on alhainen tai sitä ei ole ollenkaan. (Topic 2002, 26-27.)

3 STREAMAUKSEN TEKNIikka

Streamaustekniikan voi toteuttaa monella eri tavalla. Tässä kappaleessa tutkitaan sitä, mistä kaikesta streamaus koostuu. Seuraavassa käsitellään myös sitä, mitä palvelimen tekniikan puolella tapahtuu ja mitä taas vastaanottajan puolella sekä minkälaisen vaiheiden läpi on mentävä, että näyttöpäätellä näkyy video asiakkaan pyynnöstä. Osa-alueet koostuvat stream-palvelimen tekniikasta, asiakkaan tekniikasta sekä käytettävistä protokollista ja formaateista.

Videostreamin arkkitehtuuri on suunnilleen seuraavanlainen: raakamateriaali esim. video ja audio kompressoidaan, jonka jälkeen se tallennetaan stream-palvelimen tallennuslaitteelle. Sieltä se kulkee sovelluskerrkosen QoS-hallinnan kautta kuljetusprotokollien pakkauksen jälkeen Internetiin ja sieltä vastaanottajan koneelle. Vastaanottajan koneella kuljetusprotokollat purkavat paketit. Tämän jälkeen data kulkee taas sovelluserroksen QoS-hallinnan läpi, jonka jälkeen video ja audio dekodataan sekä molemmat synkronoidaan, ennen kuin video näkyy vastaanottajalle. (Wang ym. 2002, 520.) Kuva 1 avaa videostreamauksen arkkitehtuuria.



KUVA 1. Videostreamauksen arkkitehtuuri

Nykyään yhä useampi sivusto käyttää stream-videoissaan adaptive streaming -tekniikkaa. Adaptive streaming on prosessi, joka säätää websivulle lähetetyn videon laatua verkon tilanteesta riippuen, ja näin varmistaa parhaan katselukokemuksen. Internetyhteyksien nopeudet vaihtelevat laajasti verkon tilanteen mukaan. Esimerkiksi käyttäjällä joka yhdistää ISP:hen 56 Kbps, ei välttämättä ole koko ajan vapaana kaistaa 56 Kbps:n verran. Kaistan leveys vaihtelee, mikä tarkoittaa sitä, että yhteyden nopeus kasvaa ja laskee. Tämä aiheuttaa videon laadun heilahtelua. Adaptive streaming säätää videon bittinopeutta verkon nopeuden mukaan. Adaptive streaming yksinkertaistaa sisällön luontia ja hallintaa sekä tekee videostreamin käyttöön valjastamisesta helppoa, eikä vaadi ollenkaan koodausta. (MSDN 2011)

Ennen kuin videota voidaan streamata, se tulee kompressoida, jotta se voidaan lähettää tehokkaasti minnekään. Raakavideomateriaali käyttäisi todella suuren määrän kaistaa. Videon kompressointi voidaan jakaa kahteen eri järjestelmään: skaalautuvaan ja skaalautumattomaan. Kompressoituja bittivirtoja, joiden osat on mahdollista dekodata, saadaan aikaan käyttämällä skaalautuvaa videokoodausta. (Wang ym. 2002, 522.)

3.1 Kompressointi

Kun dekodataan vaan kompressoitua bittivirran osia koko virran sijasta, kuvan laatu tai kuvanopeus kärsii. Skaalautuva video selviää paremmin verkon kaistanleveysvaihteluista. Skaalautumaton video taas kärsii verkon vaihtelusta, koska se ei pysty mukautumaan kaistan vaihteluihin. Skaalautuva video on myös tarpeeksi tehokas ja joustava multicasting-lähetystä varten verkoissa, joissa on erilaajuisia kaistanleveyksiä. Useimmat videostream-palvelut käyttävät skaalautuvaa videokoodausta näiden syiden takia. (Wang ym. 2002, 522.)

Kompressoitu video ja sen audio tallentuvat palvelimen kiintolevyille. Tämän jälkeen sovelluskerroksen QoS-hallinta pitää huolta videon laadusta koskien vaihteluita kaistassa ja pakettien katoamista. Ruuhkahallinta ja virheenkorjaus ovat tekniikoita, jotka kuuluvat sovelluskerroksen QoS-hallintaan. (Wang ym. 2002, 522.)

3.2 Sovelluskerroksen QoS -hallinta

Tässä opinnäytetyössä puhutaan usein kerroksista. Kerroksilla tarkoitetaan tässä työssä yleensä yhtä OSI-mallin seitsemästä kerroksesta. OSI-mallin mukaan datan lähetysprosessi on jaettu seitsemään eri kerrokseen ja jokainen kerros lisää tietoa dataan vuorollaan. Kerrokset ovat ensimmäisestä viimeiseen lueteltuina: fyysinen kerros, siirtoyhteyserros, verkkokerros, kuljetuserros, istuntokerros, esitystapakerros ja sovelluserros. (Sosinsky 2009, 23-24.)

Tässä kappaleessa toistuvat myös usein termit tiedonsiirtonopeus ja väylä. Tiedonsiirtonopeudella tarkoitetaan tiedon siirtymisen nopeutta. Se on tiedon siirtymisen määrä jossain tietyssä ajassa. Väylää voidaan kutsua alijärjestelmäksi, jonka kautta tieto kulkee paikasta toiseen. (Wikipedia 2011a; Wikipedia 2011b.)

Yleensä verkon ruuhkautuminen heikentää streamatun videon esittämisen laatua. Liiallista viivästymistä ja puskemaista pakettien katoamista voidaan vähentää ruuhkanhallinnalla sekä virheiden hallinnalla. Ruuhkanhallinta käsittää kaksi

tekniikkaa, joilla näitä ongelmia voidaan vähentää: tiedonsiirtonopeuden hallinta eli rate control ja tiedonsiirtonopeuden muotoilu eli rate shaping. Tiedonsiirtonopeuden hallinta yrittää vähentää pakettien katoamista ja ruuhkautumisen määrää minimiin soittamalla videostreamin tahdin verkon kaistanleveyteen. Tiedonsiirtonopeuden muotoilu taas pakottaa lähettäjän lähettämään streamin tahdilla, jonka tiedonsiirtonopeuden hallinta-algoritmi on määrännyt. (Wang ym. 2002, 522.)

Tiedonsiirtonopeuden hallintamallit voidaan jakaa kolmeen kategoriaan: lähteeseen perustuva, vastaanottiin perustuva ja hybriditiedonsiirtonopeuden hallinta. Lähteeseen perustuvassa tiedonsiirtonopeuden hallinnassa lähettäjä on vastuussa videon siirtonopeudesta. Lähteeseen perustuva tiedonsiirtonopeuden hallinta toimii verkosta saadun palautetietojen avulla. Lähettäjä säätelee videostreamin tahtia. Lähteeseen perustuvaa tiedonsiirtonopeuden hallintaa voidaan käyttää sekä unicast-että multicast-lähetyksissä. (Wang ym. 2002, 523.)

Lähteeseen perustuva tiedonsiirtonopeuden hallinta lähestyy unicast-videoita kahdella eri tavalla: luotaimiin ja malliin perustuen. Lähde luotaa saatavilla olevaa verkon kaistaa säätämällä lähetysnopeutta niin, ettei pakettien häviösuhde pääse ylittämään tiettyä kynnyksiä. Lähetysnopeuden säätämiseen on kaksi erilaista tapaa. Tiedonsiirtonopeutta voidaan säätää additiivisella lisäyksellä ja kerrannaisvaikutteisella vähennyksellä sekä kerrannaisvaikutteisella lisäyksellä ja kerrannaisvaikutteisella vähennyksellä. (Wang ym. 2002, 523.)

Malliin perustuva tapa pohjaa TCP-yhteyden läpisyöttömalliin. TCP-yhteyden läpisyöttö voidaan selvittää seuraavanlaisella kaavalla:

$$\lambda = \frac{1.22 \times MTU}{RTT \times \sqrt{\rho}} \quad (1)$$

missä λ on läpisyöttö, MTU on yhteyden käyttämä paketin koko, RTT on yhteyden kokonaisaika, ja ρ on pakettien häviösuhde. Yhtälöä käytetään selvittämään videostreamin lähetysnopeus. Malliin pohjautuva tiedonsiirtonopeuden hallinta on

yhteensopiva TCP:n kanssa, koska videoyhteys pystyy välttämään ruuhkautumista samalla tavalla kuin TCP. (Wang ym. 2002, 523.)

Multicast-lähetyksessä lähteeseen perustuvalla tiedonsiirtonopeuden hallinnalla video tuodaan katsojille yhtä väylää pitkin. Tällä single-channel multicast -lähetyksellä voidaan käyttää vain luotaukseen perustuvaa tiedonsiirtonopeuden hallintaa. Tehokkaan yhden väylän multicastista tekee se, että kaikki vastaanottajat jakavat saman väylän. Yhden väylän multicast ei kuitenkaan pysty joustavasti vastaamaan kaikkien vastaanottajien erilaisiin tarpeisiin kaistanopeuksien suhteen. Multicast-videota toimitettaessa erillisinä unicast-virtoina tehokkuus olisi huonoa, mutta toisaalta palvelut olisivat eriytettyjä. Tällöin jokainen vastaanottaja voisi neuvotella palvelujen parametreista lähteen kanssa. Unicast ja yhden väylän multicast ovat toistensa ääripäät, joten saavuttaakseen keskitien kaistanleveyden hyötysuhteen ja palvelun joustavuuden kanssa, on käytettävä vastaanottajapohjaista tai hybridisiirtonopeuden hallintaa. (Wang ym. 2002, 523-524.)

Vastaanottajapohjaisessa tiedonsiirtonopeuden hallinnassa vastaanottaja säätelee vastaanottonopeutta lisäämällä ja pudottamalla väyliä, kun taas lähettäjä ei puutu siirtonopeuden hallintaan. Useimmiten vastaanottajapohjaista tiedonsiirtonopeuden hallintaa käytetään kerroksittaiselle multicastille. Kuten lähdepohjaisessa siirtonopeuden hallinnassa, vastaanottajapohjaistakin siirtonopeuden hallintaa lähestytään luotaukseen perustuvalla ja malleihin perustuvalla tavalla. Luotaukseen perustuva tiedonsiirtonopeudenhallinta toimii yksinkertaistetusti seuraavalla tavalla: kun ruuhkautumista ei ole havaittu, vastaanottaja luotaa saatavissa olevaa kaistaa liittymällä kerrokselle. Tämä kasvattaa vastaanottonopeutta. Jos ruuhkaa ei havaita liittymisen jälkeenkään, liittymiskokeilu on onnistunut, mutta jos ruuhkaa havaitaan vastaanottaja pudottaa uuden lisätyn väylän. Malliin perustuva tapa taas arvioi selvästi saatavilla olevaa verkon kaistaa. (Wang ym. 2002, 524.)

Hybridisiirtonopeuden hallinnassa vastaanottaja säätelee videostreamien vastaanottoa lisäämällä ja pudottamalla väyliä pois käytöstä. Samalla aikaa lähettäjä säätelee jokaisen väylän siirtonopeutta lähettäjiltä saadun palautteen avulla.

Hybridisiirtonopeuden hallinnasta esimerkkinä voidaan sanoa ”destination set grouping”. (Wang ym. 2002, 524.)

Tiedonsiirtonopeuden muotoilu on ruuhkanhallinnan toinen tekniikka, jolla voidaan vähentää pakettien häviämistä ja viiveitä sekä niistä aiheutuvaa haittaa. Tiedonsiirtonopeuden muotoilu on tekniikka, jolla esikompressoitujen videostreamien siirtonopeutta voidaan mukauttaa tavoitenopeuden rajoitukseen. Siirtonopeuden muotoilija on rajapinta tai filteri kompressointikerroksen ja verkkoliikennekerroksen välissä tai kahden verkon segmentin välissä. Tiedonsiirtonopeuden muotoilijan avulla videostream saadaan vastaamaan saatavilla olevan verkon kaistaleveyteen. Siirtonopeuden muotoilijoita on viisi erilaista tyyppiä: koodekkisuodatin eli codec filter, kehyksenpudottajasuodatin eli frame-dropping filter, kerroksen pudotussuodatin eli layer-dropping filter, taajuussuodatin eli frequency filter, ja dekvantisointisuodatin. (Wang ym. 2002, 525.)

Koodekkisuodattimet dekompressoivat ja kompressoivat videostreamia. Niitä käytetään transkoodaukseen eri kompressointimallien välillä. Riippuen käytetystä kompressointimallista, transkoodaus voidaan yksinkertaistaa ilman täyttä dekompressointia ja uudelleen kompressointia. (Wang ym. 2002, 525.)

Kehyksen pudotussuodatin voi erottaa kehystyypit ja pudottaa kehyksiä tärkeyden mukaan. Niitä käytetään vähentämään videostreamin datasuhdetta poistamalla jonkun määrän kehyksiä. Tämä selittyy sillä, että jäljelle jäävät kehykset vaativat matalamman tiedonsiirtonopeuden. Kehyksenpudotussuodattimia voidaan käyttää lähteessä tai verkossa. (Wang ym. 2002, 525.)

Kerroksen pudotussuodatin voi nimensä mukaisesti erottaa ja pudottaa pois kerroksia niiden tärkeyden mukaan. Kerrokset pudotetaan järjestyksessä korkeimmasta parannuskerroksesta peruskerrokseen. (Wang ym. 2002, 525.)

Taajuussuodatin toimii kompressointikerroksessa. Taajuussuodatusmekanismeihin kuuluu alipäästö-, värin vähennys- ja värillisestä yksivärisen -suodatus.

Alipäästösuodatus hylkää ylempien taajuuksien DCT-kertoimet. Väriinvähennyssuodatin tekee saman, mutta toimii vain video-streamin krominanssitiedoilla. Värillisestä yksiväriseen -suodatus poistaa kaiken väritiedon videostreamista. Taajuussuodattimet vähentävät kaistanleveyttä vaikuttamatta kuvannopeuteen. (Wang ym. 2002, 525.)

Dekvantisointisuodattimet toimivat kompressoititasolla. Ensiksi suodatin purkaa DCT-kertoimet kompressoidusta video-streamista dekvantisoimalla. Sen jälkeen suodatin kvantisoii DCT-kertoimet laajemmalla kvantisointiaskelkoolla, mikä vähentää siirtonopeutta. (Wang ym. 2002, 525.)

Ruuhkanhallinnan tehtävä on pakettihäviön vähentäminen. Internetissä pakettien häviämistä on mahdotonta poistaa kokonaan, ja maailmassa keksitään vauhdilla mekanismeja, joilla voidaan parantaa videoiden katselukokemusta pakettien häviämisestä huolimatta. Yksi tällaisista mekanismeista on virheiden hallinta. (Wang ym. 2002, 525.)

Virheiden hallinnalla voidaan käsittää muutama eri tapa jota voidaan käyttää hyväksi videostreamauksen laadun parantamisessa. Nämä ovat FEC eli forward error correction, viive-rajoitettu uudelleensiirto eli delay-constrained retransmission, virheitä kimmottava koodaus eli error-resilient encoding ja virheiden salaaminen eli error concealment. (Wang ym. 2002, 526.)

FEC lisää tarpeetonta tietoa alkuperäiseen viestiin, joten viesti voidaan rakentaa uudestaan pakettien häviämisestä huolimatta. Internetsovelluksissa paketteihin laitetaan lohkokoodit. Video-stream jaetaan segmenteiksi, joista jokainen paketoitetaan k -paketeiksi. Sitten jokaisen segmentin k -paketeille annetaan lohkokoodi n . Tämä johtaa siihen, että jos halutaan palauttaa segmentti täysin, käyttäjän tarvitsee vain saada n -pakettilohkon k -paketit. (Wang ym. 2002, 526.)

Uudelleensiirtoa käytetään harvoin reaaliaikaisen videon lähetyksessä, koska uudelleen lähetetty paketti saattaa myöhästyä toistohetkestään. Kuitenkin streaming-

sovelluksissa, jos liikenne yhteen suuntaan on riittävän nopea eikä viivettä synny yli suurimman sallitun rajan, viiverajoitettu uudelleensiirto on toteuttamiskelpoinen vaihtoehto virheen hallinnalle. Unicast-lähetyksessä, riippuen siitä kuka päättää lähettääkö vai vastaako uudelleensiirto pyyntöön, on kolme viiverajoitettua uudelleensiirtomekanismia. Nämä mekanismit ovat vastaanottajaan pohjautuva hallinta, lähettäjään pohjautuva hallinta sekä hybridihallinta. (Wang ym. 2002, 526.)

Vastaanottajaan perustuvan hallinnan tarkoitus on minimoida uudelleensiirtopyyntöjen, jotka eivät tule saapumaan ajoissa näytettäväksi, määrää. Tässä vastaanottajaan pohjautuvassa hallinnassa vastaanottaja toteuttaa seuraavaa algoritmia:

$$\begin{aligned}
 & \textit{Kun vastaanottaja havaitsee paketin } N \textit{ häviön:} \\
 & \textit{jos } (T_c + RTT + D_s < T_d(N)) \\
 & \textit{lähete pyyntö paketista } N \textit{ lähettäjälle}
 \end{aligned} \tag{2}$$

missä T_c on nykyinen ajanhetki, RTT on arvioitu kiertoaika, D_s on löysä aikaväli ja $T_d(N)$ on aika, milloin paketti N on tarkoitus näyttää. Löysä aikaväli D_s voisi sisältää myös virhemarginaalit RTT -arvioinnista, lähettäjän vasteajasta sekä vastaanottajan dekodausviiveestä. (Wang ym. 2002, 526.)

Lähettäjään perustuvan hallinnan on tarkoitus estää pakettien, jotka tulevat myöhästymään, uudelleensiirto näyttöajankohdastaan vastaanottajan päässä. Lähettäjä toteuttaa seuraavaa algoritmia:

$$\begin{aligned}
 & \textit{Kun lähettäjä saa pyynnön paketista } N : \\
 & \textit{jos } (T_c + RTT/2 + D_s < T'_d(N)) \\
 & \textit{uudelleensiirrä paketti } N \textit{ vastaanottajalle}
 \end{aligned} \tag{3}$$

missä T'_d on arvio $T_d(N)$. (Wang ym. 2002, 527.)

Hybridihallinta on nimensä mukaisesti vain yhdistelmä lähettäjäan ja vastaanottajaan perustuvasta hallinnasta. Multicast-lähetysessä uudelleensiirtoa täytyy rajoittaa lähekkäin sijoitettuihin multicast-jäseniin, koska yhdensuuntaisten matkojen määrä näiden jäsenten välillä jää pieneksi, mikä tekee uudelleensiirrosta tehokasta. Yleensä puulogiikka säädetään rajoittamaan uudelleensiirtopyyntöjen määrää ja skaalaa sekä saavuttamaan elpymistä lähekkäin sijaitsevien multicast-jäsenten kesken. (Wang ym. 2002, 527.)

Virheitä kimmottava koodaus eli error resilient encoding käsittää kaksi eri tekniikkaa Internetvideoista puhuttaessa. Ne ovat optimaalisen tilan valinta ja moninkertaisen kuvauksen koodaus. Pakettien häviämisen vaikutus videon laatuun riippuu koodausmallista lähteessä, verkon ruuhkautuneisuudesta sekä virheiden kätkemismallista vastaanottajan päässä. Ulkoinen koodaus voi saavuttaa korkean kompressointitehokkuuden sillä uhalla, että virheet lähtevät leviämään. Sisäinen koodaus taas pysäyttää virheiden leviämisen tehokkuuden kustannuksella. Ulkoisen ja sisäisen tilan välillä valitsemisen mekanismi tulisi olla hyvä, jotta videon kestävyyttä voitaisiin parantaa. Parhaisiin tuloksiin päästään, kun otetaan huomioon lähteen käyttäytyminen, reitin ominaisuudet sekä vastaanottajan käyttäytyminen. Moninkertaisen kuvauksen koodaus on toinen tapa tasapainoilla kompressointitehokkuuden ja -kestävyyden välillä. Se on myös tehokas käsitellessään pakettien häviämistä. (Wang ym. 2002, 527-528.)

Virheiden kätkemisellä tarkoitetaan sitä, että kun havaitaan, että jokin paketti on hävinnyt, vastaanottaja kätkee menetetyn datan, mikä tekee videon esityksestä miellyttävämmän katsoa. Katsojat voivat sietää vain tietyn verran videosignaalien säröytymistä, joten virheiden kätkentä on kannattava tapa pakettien häviämisen käsittelyyn. Virheiden kätkemistä voidaan lähestyä kahdella eri tapaa: tilaa koskeva ja ajallinen interpolointi. Tilaa koskevassa interpoloinnissa puuttuvat pikseliarvot rakennetaan uudestaan käyttäen tilallisesta viereisiä arvoja. Ajallinen interpolointi rakentaa menetetyn datan edellisistä kehyksistä saaduista tiedoista. (Wang ym. 2002, 528.)

3.3 Streaming-palvelimet

Streaming-palvelimet ovat yleensä optimoitu streamausta varten. Ne käyttävät paremmin hyödykseen verkon kaistanleveyttä ja palvelinkoneen resursseja verrattuna tavallisiin webpalvelimiin. Palvelimissa on omat mallinsa erilaisten ongelmien, kuten ruuhkautumisen estämiseksi. (Topic 2002, 148.)

Streaming-palvelimet ovat avainasemassa streamauspalveluiden toimittamisessa. Jotta pystytään tarjoamaan laadukasta streamia, palvelimien tulee pystyä prosessoimaan multimediatdataa aikarajoitusten sisällä estääkseen nytkähdyksiä videossa ja poksahduksia äänessä. Streaming-palvelimien täytyy myös toimia videonauhurien tapaisesti, eli ohjaustoiminnot kuten pysäytys, keskeytys, jatkaminen ja kelaaminen eteen sekä taakse pitäisivät olla mahdollisia. Streaming-palvelimet koostuvat yleensä kolmesta alajärjestelmästä: tiedottajasta, käyttöjärjestelmästä sekä tallennusjärjestelmästä. (Wang ym. 2002, 533.)

Tiedottaja käsittää sovelluskerroksen ja kuljetusprotokollat, jotka palvelin toteuttaa. Tiedottajan kautta asiakkaat voivat keskustella palvelimen kanssa ja pyytää haluamaansa multimediasisältöä. Käyttöjärjestelmät ovat erilaisia verrattuna perinteisiin käyttöjärjestelmiin, ja niiden on tyydytettävä stream-palveluiden reaaliaikaiset vaatimukset. Käyttöjärjestelmä suojaa tietokoneen laitteistoa ja komponentteja muilta ohjelmistoilta sekä tarjoaa erilaisia palveluita liittyen suorittimeen, muistiin, tallennustilaan sekä syöttö- ja ulostulolaitteisiin. (Wang ym. 2002, 534.)

Käyttöjärjestelmien yksi iso tehtävä on prosessien hallinta. Prosessien hallinta kartoittaa yksittäiset tehtävät suorittimelle määritetyn aikataulumenettelyn mukaan niin, että kaikki prosessit täyttävät vaatimuksensa. Yleensä näitä menettelytapoja on kaksi: aikaisimman määräajan prosessi ensin ja ensisijaisimman tärkeät prosessit ensin. Ensijaisuus määräytyy sen mukaan, kuinka nopeaan tahtiin saapuu pyyntöjä prosessin suorittamisesta. Nämä aikataulumenettelyt ovat kuitenkin vain ennaltaehkäiseviä, joten järjestykseen tulee usein muutoksia, jos jokin prosessi lähettää keskeytyspyynnön. Tämän jälkeen aikataulut on määriteltävä uudestaan. (Wang ym. 2002, 534-535.)

Resurssienhallinta on myös yksi tärkein käyttöjärjestelmän tehtävä. Resurssiksi lasketaan muun muassa suoritin, muisti ja kovalevyt. Resurssit ovat rajalliset joten palvelin voi palvella vain rajallisen määrän asiakkaita. Resurssien hallintaa tarvitaan, että resurssit voidaan sijoittaa sinne, missä niitä tarvitaan eniten. Ennen kuin multimediaspalvelin voi päästää uuden käyttäjän hakemaan sisältöä, täytyy palvelimen suorittaa pääsykoe päättääkseen voiko uuden yhteyden sallia ilman, että jo olemassa olevien yhteyksien suorituskyky kärsisi. (Wang ym. 2002, 535.)

Käyttöjärjestelmän tehtäviin kuuluu myös tiedostojen hallinta. Tiedostojärjestelmä tarjoaa pääsy- ja hallintatoiminnot tiedostojen tallentamiselle ja hakemiselle. Tiedostojärjestelmiä voidaan lähestyä kahdella eri tavalla. Tiedostojen järjestely on levyllä yhdessä paikassa, eli sitä ei ole ripoteltu useammille levyille, ja levyllä käytetään aikatauluttamisalgoritmia tiedostoihin käsiksupääsemiseksi. Toinen lähestymistapa on se, että video ja ääni tiedostot järjestetään usealle eri levyille, ja levynetsintäaikaa vähennetään aikatauluttamisalgoritmeilla. (Wang ym. 2002, 536.)

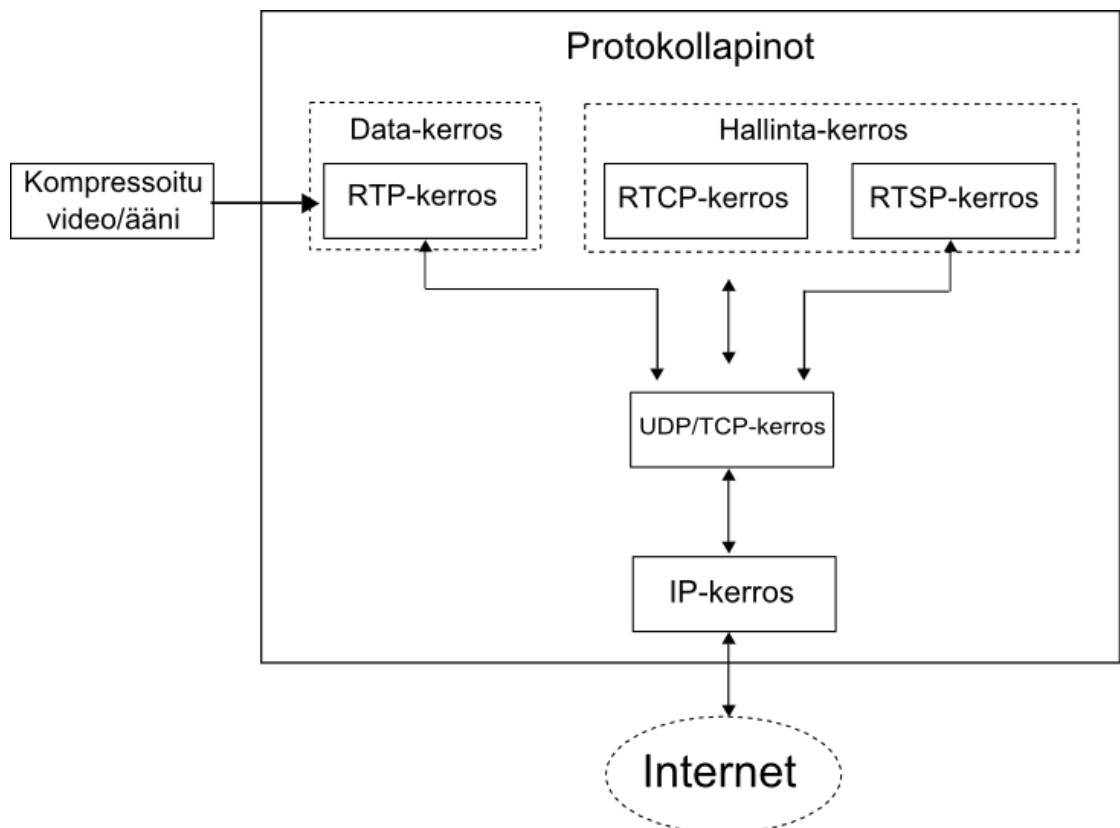
Multimediatallennusjärjestelmien haasteena on korkea läpisyöttö, suuri kapasiteetti ja viansietokyky. Jos koko videotiedosto on tallennettu yhdelle levyille, samanaikaisien pääsyjen määrä tiedostoon rajoittuu levyn läpisyötön mukaan. Tämä määrittää kuinka monta asiakasta voi katsoa samaa videota. Läpisyöttöä voi kasvattaa jakamalla tiedosto osiin ja ripottelemalla osat eri levyille. (Wang ym. 2002, 537.)

Palvelimen on pystyttävä rakentamaan uudelleen hävinnyt tieto, jotta voidaan varmistaa keskeyttämätön palvelu jopa levyjen rikkoutuessa. Tämä saadaan aikaan käyttäen ylimääräistä tietoa. Ylimääräinen tieto voi olla joko virheenkorjauskoodin luomaa pariteettidataa tai kaksoiskappaledataa erillisiltä levyiltä. Viansietotekniikoita on kahta erilaista: virheenkorjauskoodausta ja peilaamista. Pariteettidatan avulla menetetty tieto voidaan rakentaa uudestaan, mutta se vaatii synkronointia ja ylimääräistä prosessointiaikaa. Peilaaminen taas ei vaadi kumpaakaan, mikä yksinkertaistaa videopalvelimien suunnittelua ja käyttöönottoa. (Wang ym. 2002, 539.)

3.4 Protokollat

Protokollat on suunniteltu ja standardoitu asiakkaiden ja streaming-palvelimien välistä kommunikointia varten. Toimintojensa mukaan videostreamaukseen liittyvät protokollat voidaan jakaa kolmeen luokkaan: verkkokerroksen protokollat, kuljetusprotokollat ja istunnon hallintaprotokollat. (Wang ym. 2002, 542.)

Verkkokerroksen protokollat tarjoavat verkon käyttöön liittyviä perustoimintoja kuten osoitteistamista. Internetprotokolla eli IP toimii verkkokerroksen protokollana videon streamauksessa. Kuljetusprotokollat tarjoavat molemminpuoliset siirtotoiminnot streaming-sovelluksille. Nämä protokollat ovat UDP, TCP, RTP ja RTCP, joista UDP ja TCP ovat alemman kerroksen kuljetusprotokollat, ja RTP ja RTCP ylemmän kerroksen protokollia. Istunnon hallintaprotokolla esim. RTSP määrittää viestit sekä toimenpiteet, joilla hallitaan dataa istunnon aikana. Kuva 2 auttaa selventämään protokollien eri kerroksia streamauksessa. (Wang ym. 2002, 542-543.)



KUVA 2. Protokollapinot streamauksessa

UDP eli User Datagram Protocol on Internetprotokolla, joka luo ”tilattomia” yhteyksiä kahden isännän välille. Tilattomuus eli stateless tarkoittaa, ettei datan oikeellisuutta varmisteta. TCP eli Transmission Control Protocol on nykyään ehkä laajimmin käytetty kuljetusprotokolla. TCP sisältää mekanismit, jotka käsittelevät viestin sisältämän tiedon. Samalla varmistetaan, että data lähetetään helposti käsiteltävissä osissa, saapuu perille eheänä ja että data sekvensoidaan ja että uudelleenkoottu data vastaa alunperin lähetettyä kopiota. Real-time Transport Protocol eli RTP on yksi tapa lähettää paketteja, joissa on rikas sisältö Internetin välityksellä. RTSP on lyhenne sanoista Real-Time Streaming Protocol. RTSP on sovelluskerroksen protokolla, jolla hallitaan sitä, miten mediasoitin käsittelee mediapalvelimen lähettämää streamia. (Sosinsky 2009, 467, 456, 646 – 647.)

RTP-, RTCP- ja RTSP-kerrokset noutavat ja paketoivat kompressoidun videon ja äänen. Tässä kerroksessa paketit saavat myös ajastus- ja synkronointitiedot sekä järjestysnumerot. Seuraavaksi paketit siirtyvät UDP- ja TCP-kerrokseen ja lopulta IP-kerrokseen. IP-paketit lähetään Internetin välityksellä vastaanottajalle. Vastaanottajan päässä mediastreamit käyvät saman prosessin läpi käänteisessä järjestyksessä ennen toistamistaan. (Wang ym. 2002, 543.)

3.4.1 Kuljetusprotokollat

Median streamaukseen käytettävät kuljetusprotokollat ovat UDP, TCP, RTP ja RTCP. UDP- ja TCP-protokollat multipleksaavat eri sovelluksista tulevia, mutta samalla koneella pyöriviä datavirtoja. Virnehallinnassa UDP ja TCP käyttävät tarkastussummia havaitakseen bittivirheitä. Jos virhe havaitaan, TCP/UDP-kerros hylkää paketin, jottei ylempi kerros saa korruptoitunutta pakettia. (Wang ym. 2002, 544.)

TCP-protokolla käyttää myös uudelleensiirtoa kadonneiden pakettien löytämiseen. Tämä tekee siitä TCP:stä siirron kannalta luotettavamman. Tämän lisäksi TCP sopeutuu siirtotahtiin verkon ruuhkatilan mukaan. TCP:n uudelleensiirto aiheuttaa kuitenkin viiveitä, jotka eivät ole hyväksyttäviä streamingsovelluksissa, minkä vuoksi UDP:tä käytetään kuljetusprotokollana videostreameissa. (Wang ym. 2002, 544.)

RTP on datan kuljetusprotokolla, jonka kumppanina toimii RTCP, jota voidaan kutsua hallintaprotokollaksi. RTP tukee mediastreamausta usealla eri tavalla. Näistä yksi on aikaleimaus, jonka avulla muut sovellukset voivat synkronoida eri mediastreamejä. RTP antaa saapuville paketeille myös järjestysnumerot, koska paketit eivät aina saavu järjestyksessä vastaanottajalle. Järjestysnumeroita käytetään myös pakettien häviämisen havaitsemiseen. (Wang ym. 2002, 544.)

Tietosisältötyypin tunnistaminen kuuluu myös RTP-protokollaan. Näiden tunnisteiden avulla vastaanottaja tulkitsee saapuvien pakettien sisällön. RTP-protokollaan kuuluu myös lähteen tunnistus. Näin vastaanottaja pystyy erottamaan eri lähteet. (Wang ym. 2002, 544.)

RTCP-hallintaprotokolla työskentelee RTP:n kanssa. RTCP toimittaa QoS-palautetta RTP-istunnon osanottajille. Tarkemmin sanottuna RTCP:n tarjoamat palvelut ovat: QoS-palaute, osallistujan tunnistus, hallintapaketin skaalaus, intermedian synkronointi ja minimaalinen istunnon hallintatieto. QoS-palaute on RTCP:n päätoiminto ja se antaa sovellukselle palautetta datan jakelun laadusta. Palautteen avulla voidaan esimerkiksi säätää lähetysnopeutta tai päätellä ruuhkautumisen laajuus. (Wang ym. 2002, 544-545.)

Osallistujan tunnistaminen on mekanismi, jonka avulla paketteihin voidaan lisätä muun muassa käyttäjänimiä ja sähköpostiosoitteita. Hallintapakettien skaalausmekanismi pitää hallintapakettien kokonaismäärän viidessä prosentissa koko istunnon kaistanleveydestä. Hallintapaketeista 25% on kohdennettu lähettäjän raportteihin ja 75% vastaanottajan raportteihin. RTCP-lähettäjän raportit sisältävät tietoa reaaliajasta ja vastaavista RTP-aikaleimoista. Näitä tietoja voidaan käyttää intermediasykronoinnissa kuten videon huulisynkronoinnissa. Minimaalista istunnonhallintatietoa voidaan käyttää lähettämään siirtoistunnon tietoa, kuten osallistujien nimiä. (Wang ym. 2002, 545.)

3.4.2 Istunnonhallintaprotokolla

RTSP on istuntoprotokolla, joka soveltuu parhaiten median streamaukseen. Sen tärkeimpiin toimintoihin kuuluu videonauhurimaiset toiminnot kuten kelaaminen, stop, pause ja play. Sen lisäksi jakelukanavat ja -mekanismit valitaan RTSP:n avulla. RTSP toimii niin multicastissa, kuin unicastissa. (Wang ym. 2002, 545.)

Toinen RTSP:n tärkeimmistä toiminnoista on luoda ja hallita jatkuvia ääni- ja videostreamoja mediapalvelimien ja asiakkaiden välillä. Asiakas voi pyytää esityksen kuvausta sekä pyytää palvelinta järjestämään istunnon lähettääkseen pyydetyn tiedoston. Palvelin voidaan pyytää neuvotteluun toistamaan mediaa tai nauhoittamaan esitystä. Palvelin ja asiakas voivat ilmoittaa toisilleen uudesta istunnon saataville tulleesta lisämediasta. RTSP on suunniteltu streamin toimituksen aloittamiseen ja streamin toimituksen ohjaamiseen palvelimelta asiakkaille. (Wang ym. 2002, 545-546.)

3.5 Vastaanottajan tekniikka

Tänä päivänä PC on suosittu alusta streamingmedian renderöintiin kulutusta varten, koska PC:t ovat hyvin yleisiä ja verrattaen kustannustehokkaita. PC ei kuitenkaan ole ideaali laite streamivideon katselulle, koska kuluttaja tahtoisu usein katsoa videoita maaten sohvalla, laitteen sijaitessa kauempana huoneessa. (Topic 2002, 75.) PC:n rinnalle streamin katseluun ovatkin tulleet televisiot, esim. Samsungin Smart-TV, ja muut kodin multimedialaitteet, joissa kaikissa on enenevässä määrin suora nettiyhteys sekä tarpeeksi muistia ja kaistaa on-demand –streamin ja livestreamin katselulle. Yleistyneitä ovat videonvuokraus kotisohvalta käsin ja erilaiset medialaitteiden sovellukset, joista osa löytyy myös Internetistä pc:ltä katsottavaksi. (Samsung 2011.)

Streamivideon tietokoneeltaan katselemiseen käyttäjä tarvitsee soittimen. Streamivideosoitin on oikeastaan sovellusohjelma, joka kiinnittyy verkon liitäntään, joka toimittaa streamivideodatapaketit sekä näyttö- että audiolaitteisiin, jotka näyttävät lopullisen ohjelman. Kaikki, mitä soittimen tarvitsee tehdä on puskuroida datapaketit ja varmistaa, että ne ovat oikeassa järjestyksessä. Tämän jälkeen soitin purkaa

datapaketit dekompressoiden digitaalisen hyötykuorman. Sitten soitin maalaan raa'an audio- ja videodatan näyttöpöydälle ja lähettää audiodatan äänikortin muuntimelle. Muunnin muuttaa audion digitaalisesta analogiseksi. Soitin myös pitää huolen siitä, että video streamautuu lähteestä renderöitäväksi katkeamatta ja häiriöittä. (Topic 2002, 76.)

Mediasynkronointi mahdollistaa asiakkaalle erilaisten mediastreamien, kuten videon, audion ja kuvan, esittämisen synkronisella tavalla. Serverit lisäävät aikaleimoja jokaisen lähettämäänsä pakettiin. Vastaanotin käyttää näitä aikaleimoja arvioidessaan verkon värinää ja sopivia bufferointikokoja. Näin vastaanotin varmistaa synkronoidun ja jatkuvan median esityksen. (Wang ym. 2002, 554.)

3.6 Formaattit

Videon streamauksessa käytetään erilaisia formaatteja, joilla jokaisella on hyvät ja huonot puolensa, on enemmänkin kyse streamin laatijan käyttötottumuksista. Yleisimpiä käytettyjä streamformaatteja ovat: Windows Media, RealMedia, Quicktime, MPEG-4 ja Adobe Flash. (Mediacollege 2011a.)

Windows Medialla on ilmeinen etu siitä, että sitä tukee suurin toimija henkilökohtaisessa tietojenkäsittelyssä, Microsoft. On melko varmaa, ettei Microsoft mene konkurssiin tai muuta radikaalisti liiketoimintamalliaan lähiaikoina ja se tuo vakautta. Päivitysten saaminen Windows Media Playeriin on helppoa ja useimmilla ihmisillä, jotka käyttävät Windowsia, on valmiina oikea soitin. On todettu, että Windows Media Player toimii hyvin, tiedostot ovat suhteellisen korkealaatuisia ja pienikokoisia. Haittapuolena on se, että Microsoft tunnetaan usein muuttuvista formateistaan ja standardeistaan. Videon tuottajana voi olla vaikeaa pysyä mukana uusimmassa versiossa. Microsoft on myös erittäin tiukkien tekijänoikeuksien suojaama, ja kaikilla muilla alustoilla kuin Windowsin Internet Explorerilla saattaa videon katselussa tulla ongelmia. (Mediacollege 2011f.)

RealMedialla on maineikas historia ja se oli yksi ensimmäisistä vakavasti otetuista videon streamaukseen käytetyistä ratkaisuista. Alkuaikoina se oli monen mielestä paras streamausformaatti. RealMedia on edelleenkin hyvä formaatti, mutta ei niin laajalti käytetty ratkaisu kuin ennen. RealMedian videokuva tapaa olla vähemmän terävä kuin muiden formaattien - tämä voi olla hyvä tai huono asia riippuen käyttäjän tottumuksista. Liikkuvaa kuvaa hoidetaan kohtuullisen hyvin, mutta pienempiä yksityiskohtia voidaan menettää helposti. Kaiken kaikkiaan kuvanlaatu on täysin riittävä useimpiin sovelluksiin. (Mediacollege 2001e.)

Yksi suuri ongelma on ilmaisen Real Media Player:in lataamisen vaikeus. RealMedia on kamppailut pitääkseen yllä elinvoimaista, kaupallista liiketoimintamallia ja he suosittelivat käyttäjää vahvasti ostamaan täyden version ilmaisen version sijasta. Tämä on tuottanut pettymyksen monelle ihmiselle ja RealMedian maine on kärsinyt siitä viime vuosina. Yleisimmistä streamingformaateista, RealMedia näyttäisi joutuvan kamppailemaan kaikkein eniten. (Mediacollege 2001e.)

QuickTime on formaattina arvostettu, ja ollut läsnä digitaalisten videoiden synnystä lähtien. Monien mielestä Quick Time on monimutkaisempi kuin esimerkiksi Windows Media tai Real Media, kun taas toisille Quick Timen useat ominaisuudet ovat tarpeen. Nettisivujen suunnittelijan täytyy lähinnä tietää kuinka käsitellä QuickTimen .mov-tiedostoja. Quicktimen lisäominaisuuksilla voit luoda esimerkiksi interaktiivisia videoita tai panoraamoja. Joidenkin mielestä, QuickTime on paras formaatti video streamaukseen, kun taas toiset vain eivät pysty tekemään sillä mitään. Siinä on monia uniikkeja ominaisuuksia. Vaaditaan paljon kokeilua, että QuickTime:llä saadaan aikaan hyvänlaatuista jälkeä. QuickTime:n etu on sen hyvä integroitavuus muiden tuotteiden kanssa ja laaja tuki muilta sovelluksilta. (Mediacollege 2001d.)

MPEG-4 formaatti kehitettiin 1990-luvun lopulla. Aluksi kehitystyötä tuki monta yhtiötä kuten Apple ja Microsoft, mutta Microsoft on jo lopettanut aktiivisen tuen. MPEG-4:n etu on se, että sillä voidaan saavuttaa hyvä videonlaatu pitäen tiedoston koon silti alhaisena. Formaattina MPEG-4 voi olla hämmentävä, sen monimutkaisuuden vuoksi. Formaattista löytyy useita eri variaatioita, joista osa on ISO-yhteensopiva ja osa ei. Jotkut toistimet esim. Quicktime kuitenkin toistaa sekä

ISO-yhteensopivat .mp4 tiedostot ja ei yhteensopivat .mov tiedostot. (Mediacollege 2001b)

Flash on erittäin vahva tekijä videon streamauksen maailmassa. Tätä formaattia Mediacollege (2001c) suosittelee voimakkaimmin. Flash käyttää kahta formaattia: .SWF:ää standardeille Flash-tiedostoille, joita käytetään verkkosivuilla, ja .FLV:tä, joka on erityinen Flash Video -formaatti. FLV-tiedostoja voidaan soittaa SWF-tiedostojen sisällä. Flash tukee myös H.264-tiedostoja, mikä on merkittävä harppaus eteenpäin. (Mediacollege 2011c.)

Flash:in haittapuolena on sen kalleus. Saadakseen kaiken irti tästä formaatista, käyttäjän pitää omistaa Adobe Flash Authoring -ohjelma. Sen lisäksi, että se on tyyris, sen käytössä on paljon opittavaa, varsinkin, jos videon editointiohjelmat eivät ole käyttäjälle tuttuja. Hyvänä puolena on se, jos Flashiin on varaa ja on valmis tekemään töitä oppiakseen, että Flash antaa käyttäjälle todella paljon käyttövoimaa ja joustavuutta. Flash tarjoaa esimerkiksi mukautetut kontrollit ja valikot, interaktiivisia videoita ja animaatioita sekä kehittyneen integroinnin web-sivujen kanssa. Flashillä on myös se suuri etu, että se on laajimmin tuettu, erilaisten alustojen soitin. Se on niin yleinen liitännäinen, että lähes kaikilla tietokoneen käyttäjillä on se. (Mediacollege 2011c.)

3.7 Kassun karnevaaleilla käytetyt sovelluskerroksen ohjelmat

Kassun karnevaaleilla streamauksessa käytettiin Anystreamin tai nykyään Telestreamin Agility ohjelmaa sekä Wowza media server:iä. Raakamateriaali, joka meni streaminä ulos, kuvamiksattiin studiossa.

3.7.1 Anystream Agility

Agilityllä pystytään valitsemaan median lähteeksi valmis tiedosto, nauha tai elävää kuvaa ja ääntä, kuten monikameratuotannossamme käytettiin. Agilityllä pystyy tuottamaan korkealaatuisia streamauttuja videoita lukuisissa suosituissa formaateissa kuten Real, Flash 8, Windows Media ja QuickTime. Agility muuntaa analogisen ja

digitaalisen median mahdollistaen sisällön jakamisen ja toistamisen formaatista riippumatta. Agility automatisoi eri formaattien välisen koodauksen sekä sopeutuu käytössä olevaan lähetysoinfrastruktuuriin. (Anystream 2007, 2-6.)

3.7.2 Wowza Media Server

Wowza media server on palvelinohjelmisto, jolla voidaan streamata liveinä sekä on-demandinä videota, ääntä ja muita sovelluksia julkisissa ja yksityisissä verkoissa. Wowza Media Server:in streamia voisi halutessaan toistaa niin tietokoneella, mobiililaitteilla kuin myös Internetiin kytketyillä televisioilla ja muilla verkkolaitteilla. (Wowza 2009.)

4 LIVESTREAM KASSUN KARNEVAALEILLA

Kassun karnevaalit tapahtuma järjestettiin 7. - 10.9.2011 Mikkelin ammattikorkeakoulun Kasarmin kampuksella. Tapahtumassa juhlittiin Mikkelin ammattikorkeakoulun 20. lukuvuoden alkamista. Tapahtumapaikkana toimi Kasarmin kampuksen paraatitentille pystytetty 1000 hengen juhlateltiltä. Tapahtuma oli kaikille avoin, ja ohjelmaa olikin monelle eri kohderyhmälle. Esimerkiksi lapsille oli järjestetty oma Hippo-tapahtumansa lauantai-aamupäivällä karnevaalien viimeisenä päivänä. Ohjelmaa oli järjestetty pitkin päivää ja juhlapäivät huipentuivat iltakonsertteihin. Joiden esiintyjäkaarti koostui tämän hetken kuumimmista nimistä, kuten Anna Puu ja Paleface.

Toteutimme opiskelijatyönä monikameratuotannot Kassun karnevaalien MAMK 20 vuotta -juhlasta, Osmo's Comos -yhtyeen konsertista sekä Paukkumaissi-yhtyeen konsertista. Tuotannon tarkoituksena oli saada 20-vuotisjuhla näkymään myös ihmisille kotona livestreamin kautta. Tuotannossa käytettiin Mikkelin ammattikorkeakoulun CampusTV-oppimisympäristön laitteita. Tuotantotiimi koostui opiskelijoista, jotka tekivät opinnäytetyötä aiheeseen liittyen ja opiskelijoista, jotka tekivät projektiointoja. Tuotannon vastuhenkilönä toimi lehtori, Tomi Numento.

MAMK 20 vuotta -juhla, joka pidettiin karnevaalien ensimmäisenä iltana, kuvattiin kahdella kameralla ja ohjattiin ja leikattiin Mikpolin studiosta käsin sekä streamattiin CampusTV:n sivujen kautta. Perjantain juhlapäivän huipennus eli Osmo's Cosmos -yhtyeen konsertti kuvattiin kolmella kameralla, kuten myös Paukkumaissi-yhtyeen esitys. Ulosmennyt kuva tallennettiin myös palvelimelle ja jälkitöinä yhtyeille tehtiin tallenteet konserteistaan.

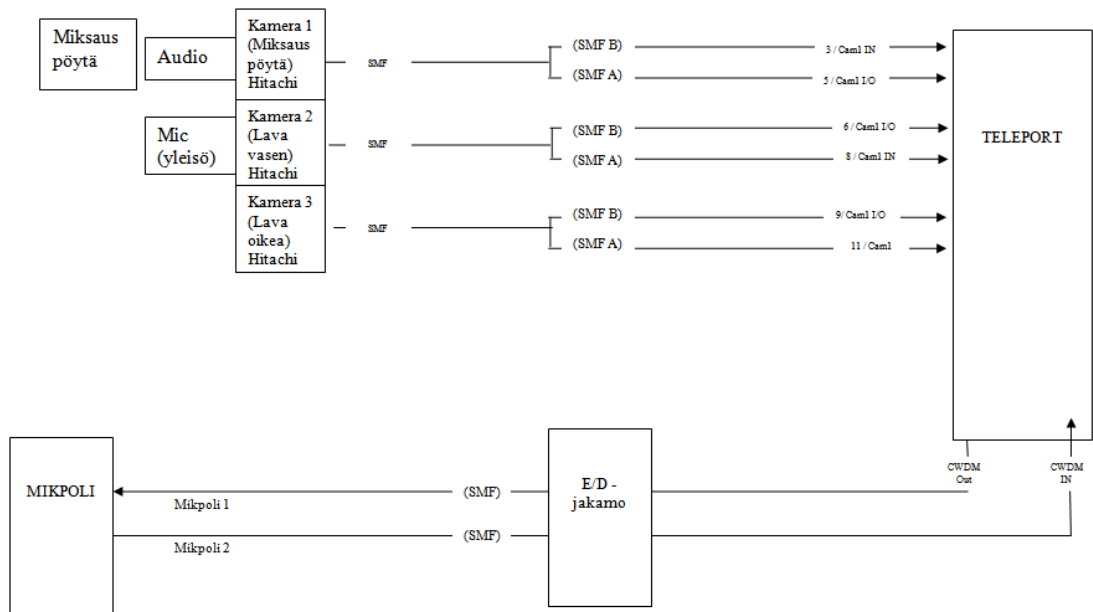
4.1 Käytetty laitteisto

Tuotannossa käytettiin kolmea Hitachi V21-W-kameraa, joita yleensä käytetään tv-studioympäristössä ja monikameratuotannoissa. Kuvasuhteena käytettiin laajakuvaa 16:9. Kameroiden kolmijalkoina käytettiin Sachtlerin DV-12-jalkoja. Kameroihin liitettiin myös Copperhead-kuituperät sekä Pag-tehoakut. Kuituperistä kameroiden kuva- ja äänisignaali sekä komentoäänet kulkevat kuitujen kautta Telecast Teleport -mediamuuntimeen. Teleportista signaalit kulkevat kuitujen kautta Mikpolin studioon.

Ulosmenevä kuva leikkattiin ja kameratyöskentely ohjattiin studiosta käsin. Kuvatarkkailu ja äänitasojen tarkkailu tehtiin myös studiosta käsin. Ääneen ei tosin voinut tehdä mitään hienosäätöjä, sillä ne otettiin teltasta suoraan miksauspöydästä kamera ykkösen kautta ja yleisöäänet erillisellä mikillä kamera kakkosen kautta. Streamauspalvelimen käyttöjärjestelmä oli Windows Server 2008, ja sovellustason streamausohjelmat olivat Anystreamin Agility ja Wowza Media Server.

4.2 Tuotannon suunnittelu

Toteuttamamme monikameratuotanto alkoi tuotannon suunnittelulla ja tuotantotiimin kasaamisella, jonka aloitimme toisen opinnäytetyön tekijän Jani Töllikön kanssa noin kuukautta ennen Kassun karnevaaleja. Tuotantotiimiin jäseniä lähdimme etsimään opiskelijakavereista, joilla oli kokemusta monikameratuotantojen tekemisestä ennestään. Tuotantotiimi saatiin ajoissa kasaan, vaikkakin minimimiehityksellä. Kolme miestä kameroissa ja kolme studiossa minä mukaan lukien. Teimme myös Janin kanssa tuotantokäsikirjan (Liite 1) ja videokaavion (Kuva 3) tuotantoa varten.



KUVA 3. Kassun karnevaalien monikameratuotannon videokaavio

4.3 Tuotannon testaus ja rakennus

Maanantaina 5.9. minä ja kaksi muuta tiimin jäsentä aloitimme Mikpolin studiossa tuotantomme tekniikan testauksen. Rakensimme tuotannon studioon ja testasimme, että kaikki tarvitsemamme laitteet toimivat. Tässä testausvaiheessa meille sattui virhe, sillä unohdimme testata toimivatko äänet. Muuten kaikki laitteet olivat kunnossa. Meidän olisi pitänyt suorittaa tekniikan testaus studiossa huolellisemmin, koska äänien testaamisen unohtaminen kostautui meille lopulta.

Tiistaina 6.9. keskipäivällä aloitimme Janin kanssa kuitujen vedon. Alkuperäisen suunnitelman mukaan meidän piti vetää kuitu suoraan studion kuitukaapilta ikkunoiden kautta juhlateltalle. Pitkän harkinnan jälkeen päätimme kuitenkin vetää kuidun kasarmin E/D-rakennuksen jakamosta teltalle. Näin kuitua ei tarvinnut vetää niin pitkää matkaa, ja saimme kuidun suojakourujen alle muiden johtojen mukana. E/D-rakennuksen jakamosta meni suora kuituyhteys Mikpolin studioon. Näin kuituyhteydet olivat MAMK:in omat eikä ulkopuolisia palveluntarjoajia tarvittu. Ensimmäinen vetämämme kuitu ei toiminut, joten vaihdoimme sen.

Keskiviikkona 7.9. kokoontuimme iltapäivällä rakentamaan tuotantoa illan MAMK 20 vuotta -juhlaa varten. Kameran oli tarkoitus sijoittaa niin, että yksi kamera on

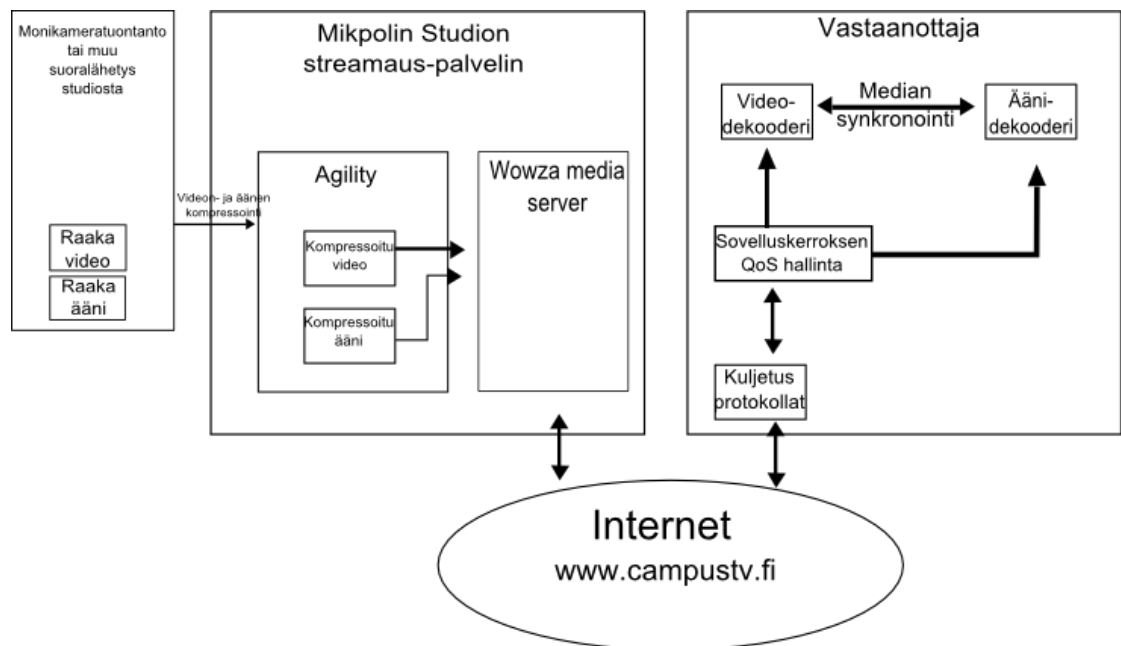
miksauskopin vieressä ja kaksi esiintymislavan edessä oikealla ja vasemmalla puolella. Kävi kuitenkin ilmi, että juhlan ohjelma tapahtuu pääasiallisesti lavan edessä alueella, jossa yleisö yleensä seisoo. Päätimme poiketa hieman suunnitelmasta, ja miksauspöydän vieressä olevan kameran lisäksi käytimme vain yhtä kameraa, joka oli sijoitettu noin teltan puoleen väliin pituussuunnassa, ja oikeassa reunassa lavalle päin katsottuna.

Kolmannen kameran sijoittaminen oikeaoppisesti olisi ollut vaikeaa sekä sähkö-, että kuituyhteyksien vetämisen kannalta. Tulimme siihen tulokseen, että vaiva ei olisi ollut kolmannen kameran tuoman lisäarvon veroista tässä tapauksessa. Tiimimme oli pienentynyt kuudesta jäsenestä viiteen, sairastapauksen vuoksi, eikä kuudetta jäsentä näkynyt enää koko tapahtuman aikana riveissämme, mutta koska käytimme keskiviikkoillan juhlassa vain kahta kameraa, ei miehistöpula vaikuttanut juurikaan tuotannon kulkuun.

Rakensimme kameroille korokkeet samoista levyistä ja jaloista joita oli käytetty lavan rakentamisessa. Korokkeet olivat noin puolen metrin korkuisia. Pystytimme kamerat korokkeille ja vedimme kuituyhteydet teleportilta, joka oli sijoitettu korokkeelleen teltan backstagella. Sähköt laitteistolle saimme tapahtuman järjestäjien puolesta jatkojohdoilla. Yleisöääniä varten mikitimme kakkoskameran Shuren mikrofonilla, jonka suuntasimme mikkitelineen avulla yleisöä kohti. Miksauspöydästä vedimme äänet kamera ykköseen xlr-piuhalla. Tässä vaiheessa tiimi halusi tauolle ja syömään. Sovimme, että kaikki tulevat takaisin noin tuntia ennen tapahtuman alkua.

Kello oli lähelle kuusi illalla eikä paikalla näkynyt tiimin jäsenistä muita kuin minä ja Jani Töllikkö. Odottelimme hetken, että lisää väkeä alkaisi saapua, ja hetken kuluttua porukkaa alkoi yksitellen ilmaantua paikalle. Vasta tässä vaiheessa meille tuli mieleen, minkä virheen tekniikan testauksessa teimme, jätimme äänet testaamatta. Kun rupesimme tarkastamaan äänien toimivuutta, miksauspöytää hoitava äänimies ei ollut vielä saapunut paikalleen, joten emme saaneet ääntä ennen kuin hieman tapahtuman alun jälkeen. Tässä vaiheessa laitoin kuitenkin ulosmenävän kuvan tallennuksen ja streamauksen päälle, ääniä katsojat saivat odotella vain noin kymmenen minuuttia.

Streamauksessa käytettiin Anystreamin Agility-ohjelmaa sekä Wowza Media Server:iä. Studioissa leikattu kuva ja ääni menee Agilityn kautta Wowza Media Serveriin ja Wowza Media Serverin kautta kuvaamme tapahtuma meni CampusTV:n nettisivuille, josta sen pystyi katsomaan kuka tahansa Flash Player -laajennoksen selaimen asentanut ja nettiyhteyden omaava henkilö. Kuvassa 4 on esitelty miltä streamauksen arkkitehtuuri koulumme laitteistolla näyttää yksinkertaistettuna.



KUVA 4. Toteutetun streamauksen arkkitehtuuri

4.4 Tuotannon kulku ja purku

Tapahtuma alkoi illalla seitsemältä, ja harmiksemme meillä ei ollut ääniä miksauspöydästä eikä yleisöääniä, mutta tätä ei kestänyt onneksi kovin kauaa. Olin kiinni ohjauksessa ja kuvamiksauksessa, jotta saisimme ulos edes säädyllistä kuvaa. Kaksi tiimin jäsenistämme oli kameroissa, ja yksi teki kuvatarkkailua ja värikorjausta kuvalle.

Juhlapaikalle sattunut MAMK:in av-asiantuntija auttoi puhelimen välityksellä yhtä tuotantotiimin jäsentä ääniongelmassa. Yleisöäänissä oli vikana äänisignaalin puuttumisen lisäksi se, että kameran pystyttänyt henkilö oli laittanut mikrofonin xlr-

piuhan väärään sisäänmenoon. Ääniongelma oli kuitenkin studiomme päässä, sillä ääniä ei oltu ohjattu tulemaan studion alakerrassa sijaitsevasta äänireititystelineestä ulosmenevään kuvaamme. Pienen etsiskelyn ja tiimimme tehokkuuden ansiosta äänet saatiin pian takaisin, ja loput juhlan kuvauksista menikin hyvin.

Juhlan päättyessä yhdeksältä purimme kamerat ja jalustat pois sekä toimme ne sisälle studioon säilytykseen. Keräsimme turhat kuidut pois, ja suojasimme muovipusseilla sekä teipillä kuidunpäät, joita tulimme vielä käyttämään myöhemmin. Virtalähteet ja jatkojohdot suojasimme myös samalla tavalla. Tämän päivän suoritus ei vastannut aivan odotuksia, mutta kaikki toimi hyvin loppujen lopuksi, ja olimme valmiit jatkamaa viisampina ja innokkaina perjantain Osmo's Cosmos -yhtyeen keikan kuvauksiin.

Perjantaina 9.10 kokoonnuimme tapahtumapaikalle iltapäivällä klo 16 ja aloitimme tuotannon kasausta illan konserttia varten. Yhdelle tuotantotiimin jäsenelle oli tullut este eikä hän voinut osallistua, joten meillä oli pieni alimiehitys. Neljän hengen voimin aloitimme kasaamaan tuotantoa. Rakensimme kakkos- ja kolmoskameralle korokkeet esiintymislavan oikealle ja vasemmalle puolelle.

Keskiviikon jäljiltä kuitu oli vedetty teleportilta ykköskameralle jo valmiiksi. Vedimme lyhyemmät kamerakuidut lavan edessä oleville kameroille. Pystytimme kamerat jalustoilleen rakentamiemme korokkeiden päälle. Lava oli rakennettu niin, että kamerat joutuivat kuvaamaan esiintyjä hieman alaviistosta. Äänentoistolaitteet olivat asetettu niin, että ne olisivat peittäneet suurimman osan kuvasta, jos kamerat olisivat olleet ihanteellisemmalla paikallaan. Suuntasimme backstagelta mikrofonin yleisöä kohti ottamaan talteen keikkatunnelmia. Mikrofonin kiinnitimme kamera kakkoseen, joka oli yleisöstä päin katsottuna lavan vasemmalla puolella. Tekniikka oli kasattu ja testattu illan konserttia varten, joten laitoimme kameroiden kuituperien akut latautumaan ja päätimme saapua takaisin paikalle tuntia ennen tapahtuman alkamista.

Kello oli yhdeksän illalla, ja kaikki jäljellä olevat tuotantotiimin jäsenet olivat paikalla. Vaihdoimme kameroiden kuitupäihin akut ja laitoimme kamerat päälle. Kaikki näytti toimivan, äänet mukaan lukien. Kamera ykkösen, joka oli sijoitettuna

miksauspöydän viereen, kuvassa huomasimme satunnaisen häiriön, missä kuva heilui reunalta toiselle. Päätimme puhdistaa uudestaan kuitujen päät, joilla saattoi olla vaikutusta kameran kuvaan. Puhdistimme kuidut eikä vähäistä häiriötä näkynyt enää kuvassa.

Seuraava ongelma oli päättää, missä kukin olisi illan konsertin aikaan. Meitä oli vain neljä, joten jos kaikki kolme kameraa olisivat miehitetty, studioon jäisi vain yksi, eli minä, joka keskittyisi vain ohjaukseen ja kuvamiksaukseen. Aiempi kokemuksemme rock-konserttien kuvauksesta oli se, että valoshow oli keikoilla usein todella ongelmallinen kuvauksen suhteen, etenkin pimeässä. Kuvatarkkailu ja värien korjaus tulisi olemaan ehdottoman tärkeää, että käytettävä kuva olisi esityskelpoista. Teimme päätöksen, että ykköskamera olisi kylmäkamera eli miehittämätön, ja studiossa olisi lisäksi toinen työntekijä. Rajasimme ykköskameran kuvan niin, että koko orkesteri sekä vähän yleisöä näkyi kuvassa. Kamerat kaksi ja kolme tarjoaisivat tiukempaa ja vaihtelevampaa kuvaa. Tämä osottautui hyväksi ja toimivaksi järjestelyksi.

Kello lähenei vartin yli kymmentä illalla, ja keikan oli määrä alkaa. Laitoimme studiossa kaiken valmiiksi. Käynnistin ulosmenevän kuvan tallennuksen serverille. Ohjauksen ja kuvamiksauksen lisäksi tarkkailin äänentasoja, siltä varalta ettei ääni menisi rikki tai rupeaisi kiertämään yleisömikin takia. Toinen tiimin jäsen hoiti kuvatarkkailun, mikä tuli todella tarpeeseen heti alusta asti. Teltta täyttyi yleisöstä ja keikka alkoi. Valot toimivat meitä vastaan, niin kuin olimme ennustaneetkin, joten sen suhteen teimme mielestäni oikean ratkaisun kun irrotimme yhden henkilön kamera ykkösestä kuvatarkkailuun.

Kaikki meni hyvin pitkän aikaa, jollei oteta huomioon sitä, että kamera ykkösen kuva oli vähän tylsää, ja kuvatarkkailijallakin oli kädet täynnä hommia lavan valojen takia. Kamera ykkösen kuvaan palasi sama häiriö, joka vaivasi meitä testausvaiheessa. Oletimme häiriön johtuvan siitä, että kamera ykkösen kuitua suojaava paksu kumimatto ei riittänyt vaimentamaan kuituun kohdistuvia iskuja, jotka tulivat siitä kun valtava joukko väkeä käveli tai tanssi sen yli. Kuva heittelehti vähän väliä arvaamattomasti, mikä aiheutti haasteita sen suhteen milloin kameran kuvaa uskalsi

käyttää. Sopeuduimme tilanteeseen ja käytimme kamera ykkösen kuvaa vain harkitusti.

Ongelmat eivät kuitenkaan jääneet tähän, vaan kuituun kohdistuvat tärähdykset aiheuttivat ilmeisesti sen, että noin keskivälissä konserttia ykköskameran komentoyhteydet menivät päälle, ja sieltä kuuluvan melun lisäksi kuului todella kovaa särinää ja kohinaa. Kolmoskameran kuvaaja lähti irrottamaan headset-kuulokkeita kamerasta, jotta komentoyhteydet saataisiin taas häiriöttömäksi. Valitettavasti hän irrotti kiireessä väärän piuhan kamerasta, ja menetimme miksatus äänen muutamaksi minuutiksi. Tällä välin oli vaikeaa löytää vaihtelevaa kuvaa, kun kaksi kolmesta kamerasta oli ilman kameramiestä, ja äänenä meillä toimi pelkästään yleisömikrofonista saatava karkea ääni. Kameran taakse palattua ilmoitimme hänelle, että irrotettu piuha oli väärä, joten hän lähti korjaamaan virheensä ja saimme miksatus äänen takaisin pian ja komentoyhteydet toimimaan.

Konsertin loppu sujui moitteettomasti ja kaikki hoitivat osansa ja enemmänkin tällä alimiehityksellä toimittaessa. Ykköskameran kuvan satunnainen heiluminen jatkui loppuun asti, mutta sen lisäksi ei ilmennyt enää ongelmia. Keikan loputtua purimme kamerat sisälle säilytykseen yön yli ja suojasimme kuidut ja muut laitteet. Illan ongelmat johtuivat osittain siitä, että teleportilta kamera ykköselle menevän kamerakuidun suojaus ei ollut riittävä. Osittain myös ongelmatilanteista olisi selvitty hieman paremmin, jos tiimissämme olisi ollut tällöin yksi henkilö enemmän. Lähdimme tästä uudella puhdilla seuraavan aamun Paukkumaissi -yhtyeen konserttiin, joka olisi viimeinen kuvauksistamme.

Lauantaina 10.9 kokoonnuimme kymmeneltä aamulla kasaaman tuotannon. Laitoimme kamerat pystyyn paikoilleen ja putsasimme kuitujen päät. Vaihdoimme täydet akut kameroiden kuituperiin, ja aloitimme tekniikan testauksen. Kaikki näytti toimivan hyvin, niin kuva kuin äänikin. Pidimme ruokatauon, jonka jälkeen Paukkumaissin keikka oli alkamassa. Tänä päivänä tiimimme oli taas viisilukuinen, mikä helpotti kovasti työtämme.

Yhtye aloitti esiintymisensä, ja kuvaukset sujuivat hyvin. Ykköskameran kuvassa olevaa häiriötä näkyi vain muutaman kerran koko konsertin aikana, mikä vahvisti epäilyämme häiriön syystä, sillä yleisöä ei ollut niin sankoin joukoin kuin edellisenä päivänä, ja suurin osa heistä oli lapsia.

Keikan päättyttyä aloimme purkaa koko tuotantoa. Purimme kamerat ja teleportin koulumme studioon, jonka jälkeen aloitimme vetämään kuitujamme pois. Tämän jälkeen purimme myös rakentamamme kamerakorokkeet, ja päivän työt olivat tehty. Kiitimme toisiamme tehdystä työstä, ja lähdimme kukin kotiin pohtimaan kovan työmme tuloksia.

4.5 Johtopäätökset

Tuotannon suunnittelu ja tekniikan testaus olisi pitänyt suorittaa hieman huolellisemmin. Suunnittelun määrässä ei ollut sinänsä mitään vikaa, mutta työvaiheet olisi pitänyt kirjata tarkemmin. Jos olisimme kirjoittaneet paperille kaikki testausta tarvitsevat asiat, niin mahdollisesti olisimme muistaneet testata myös äänet.

Keskiviikon juhlatuotanto lähti käyntiin vähän haparoiden, mutta loppujen lopuksi kaikki toimi niin kuin piti. Niin kuin aiemmin jo mainitsin, tuotannon suunnittelu ja tekniikan testaus olisi pitänyt hoitaa huolellisemmin, etenkin äänten testaus ei olisi saanut unohtua. Myös osasyynä haparointiin voitaisiin mainita lievä epäammattimaisuus opiskelijoista koostuvassa tuotantotiimissämme. On mahdollista, että jos työstä olisi saanut kokemuksen ja opintopisteiden lisäksi rahallista korvausta, monet olisivat olleet halukkaampia antamaan parhaan työpanoksensa.

Perjantain Osmo's Cosmos tuotantokaan ei ollut ongelmaton, mutta sujui huomattavasti paremmin kuin keskiviikon tuotanto. Suurin ongelma eli vakava alimiehitys oikeastaan aiheuttikin perjantain muut ongelmat tai ainakin hidasti niiden korjaamista. Jos tiimissä olisi ollut yksi mies enemmän, niin ykköskameran ei olisi tarvinnut olla miehittämätön eikä kuva ei olisi ollut niin yksipuolista. Lisäksi ykköskameran komentoyhteyksien häiriön korjaaminen olisi tapahtunut nopeammin.

Lisäksi ohjaus –ja kuvamiksaustehtävääni olisi helpottanut hieman jos minun ei olisi tarvinnut tarkkailla myös äänien tasoja, mutta minulla oli jo kokemusta siitä ennestään eikä se aiheuttanut suurempia ongelmia. Ykköskameran kylmäkameraksi laittaminen oli mielestäni oikea ratkaisu, sillä lavan valot eivät olleet ihanteellisimmat kuvaukselle. Kuvatarkkailijalla oli kokoajan kädet täynnä työtä värejä korjaillessa. Ykköskameran kuvassa välillä näkyvä häiriö oli harmittava, mutta sitä ei pystynyt korjaamaan kesken tapahtuman. Oletimme paksun kumimaton suojaavan kuitua riittävästi vaimentaen siihen kohdistuvat iskut. Kuidun olisi voinut vetää teltan kattotelineiden päältä jolloin kuituun ei olisi kohdistunut iskuja, mutta meillä ei ollut tähän mahdollisuutta eikä kalustoa sillä telineet olivat varsin korkealla.

Lauantain Paukkumaissi-yhtyeen kuvaus sujui lähes täydellisesti, ja sanoisin että se oli parhaiten onnistunut kuvaus kaikista kolmesta. Tiimimme oli jälleen viisihenkinen ja kaikki kamerat miehitettyinä. Ykköskameran kuvassa näkynyt häiriö teki paluun, mutta vaivasi paljon harvemmin. Tämä vahvisti epäilyämme siitä, että häiriö johtui suojamaton alla olevaan kamerakuituun kohdistuvista iskuista.

Kaiken kaikkiaan voitaisiin sanoa, että todella huolellinen suunnittelu ja testaus olisivat avaintekijät tällaisen tuotannon onnistumiseen. Lisäksi olisi todella hyödyllistä jos tuotantotiimi olisi tutustunut ennalta laitteestoon ja tuntisi niiden toiminnan etukäteen. Tämä osio olikin aika pitkälti kunnossa tuotannossamme, sillä kukaan ei ollut ensikertalainen. Tällaisen tuotannon onnistumisen kannalta olisi hyvä jos tiimissä olisi myös useampi varamies mahdollisten peruuntumisten varalta.

5 YHTEENVETO

Tämän opinnäytetyön tavoitteena oli selvittää minkälaisia työvaiheita vaati Kassun karnevaaleilla opiskelijatyönä toteutettu monikameratuotanto ja siitä lähetty, campustv.fi-sivustolla näkynyt livestream sekä miten kaikki käytännössä onnistui. Projektin toteuttaneessa ryhmässä toimin itse ohjaajana ja kuvamiksaajana. Monikameratuotanto sujui loppujen lopuksi hyvin ja siitä lähetetty streamvideo näkyi CampusTV.fi-sivustolla, niinkuin pitikin. Tästä työstä tuli loppujen lopuksi aika käytännönläheinen.

Tämä opinnäytetyö opetti minulle, että kaikki asiat pitäisi suunnitella tarkkaan ja hyvissä ajoin. Toisaalta työ opetti myös sen, että jos jokin menee pieleen, on pystyttävä sopeutumaan nopeasti uuteen tilanteeseen ja tehtävä parhaansa niillä eväillä, mitkä on jaettu. Opin myös sen, että nykypäivänä streamin tuottaminen on tehty todella helpoksi, mutta pinnan alla tapahtuu monimutkaisia prosesseja ja niiden tekniikan ymmärtäminen on tärkeää. Livestream on helppo ja edullinen tapa tuoda lisäarvoa erilaisille tapahtumille. Sen avulla saavutetaan helposti lisää yleisöä.

Vaikka onnistuimmekin kaiken kaikkiaan hyvin, parannettavaa tuotannossame olisi ollut. Suunnittelu tehtiin niin hyvin, kuin osattiin, mutta aikatauluja eri asioiden tekemiselle olisi pitänyt miettiä tarkemmin ja ennen kaikkea pysyä niissä. Omalta kohdaltani olisin voinut itse ottaa johtajan vastuun ohjaajana hieman vakavemmin, ja organisoida muiden toimintaa tiukemmin, koska kaverillinen puuhastelu kostaantui tällä kertaa hieman. Lisäksi tekniikan testauksen suorittaminen huolimattomasti voidaan laskea ainakin puolittain minun syykseni.

Jatkossakin voisi järjestää tämän tyyllisiä opiskelijaprojekteja. Projekteista saisi hyviä opinnäytetyön aiheita, projektiopintopisteitä sekä ennen kaikkea opiskelijat oppisivat paljon alasta näin käytännön kokemuksen kautta. Näitä töitä ei voi oppia kunnolla pelkästään teoriaa lukemalla. Olenkin todella kiitollinen tästä mahdollisuudesta.

Jos aloittaisin tämän opinnäytetyön tekemisen nyt, aloittaisin tutkimustyön ja kirjoittamisen paljon aikaisemmin ennen itse tuotannon järjestemistä, niin, että minulla olisi aiheesta vankka teoriaosaamisen tausta. Lisäksi suunnittelisin ja testaisin kaiken mahdollisen paljon aikaisemmin ja huolellisemmin enkä jättäisi mitään arvailujen varaan vahingossakaan.

Kaiken kaikkiaan olen tyytyväinen tekemääni työhön, sillä vaikei täydelliseen tulokseen nyt päästykään, opin kyllä sen, miten seuraavalla kerralla se voi olla jo paljon lähempänä. Opinnäytetyön tekemiseen tarvittiin paljon sellaisia avuja, mitä tulen varmasti jatkossa tarvitsemaan tietotekniikan insinöörin urallani, niin median parissa työskennellessä kuin muutenkin. Tästä työstä on myös varmasti hyötyä muille

monikameratuotannosta ja videon streamauksesta kiinnostuneille ja median parissa työskenteleville.

LÄHTEET

Anystream 2007. Agility 5.7 User Guide. Sterling: Anystream.

Bambuser 2011. WWW-dokumentti. <http://bambuser.com>. Päivitetty 2011. Luettu 7.11.2011.

Justin.tv 2011. About Us. WWW-dokumentti. http://www.justin.tv/p/about_us. Päivitetty 2011. Luettu 7.11.2011.

Keränen, Vesa, Lamberg, Niko & Penttinen, Jukka 2003. Verkkojulkaisun hallinta. Jyväskylä: Docendo.

Mediacollege 2011a. Streaming Video File Formats. WWW-dokumentti. <http://www.mediacollege.com/video/streaming/formats>. Päivitetty 2011. Luettu 14.11.2011.

Mediacollege 2011b. Streaming Video Files with MPEG-4. WWW-dokumentti. <http://www.mediacollege.com/video/streaming/formats/mpeg-4.html>. Päivitetty 2011. Luettu 15.11.2011.

Mediacollege 2011c. Streaming Video using Flash. WWW-dokumentti. <http://www.mediacollege.com/video/streaming/formats/flash.html>. Päivitetty 2011. Luettu 15.11.2011.

Mediacollege 2011d. Streaming Video with Quicktime. WWW-dokumentti. <http://www.mediacollege.com/video/streaming/formats/quicktime.html>. Päivitetty 2011. Luettu 15.11.2011.

Mediacollege 2011e. Streaming Video with RealMedia. WWW-dokumentti. <http://www.mediacollege.com/video/streaming/formats/real-media.html>. Päivitetty 2011. Luettu 15.11.2011.

Mediacollege 2011f. Streaming Video with Windows Media. WWW-dokumentti. <http://www.mediacollege.com/video/streaming/formats/windows-media.html>. Päivitetty 2011. Luettu 15.11.2011.

MSDN. Adaptive streaming. Microsoft. WWW-dokumentti. <http://msdn.microsoft.com/en-us/library/dd159743.aspx>. Päivitetty 2011. Luettu 7.11.2011.

Samsung. Koe Tv televisiossa. WWW-dokumentti. <http://www.samsung.com/fi/current/campaign/smarttv/site/index.html#/introduction>. Päivitetty 2011. Luettu 14.11.2011.

Sosinsky, Barrie 2009. Networking Bible. Indianapolis: Wiley Publishing.

Topic, Michael 2002. Streaming media demystified. New York: McGraw-Hill.

Ustream 2011. About Us. WWW-dokumentti. <http://www.ustream.tv/about>. Päivitetty 2011. Luettu 7.11.2011.

Wang, Yao, Osterman, Jörn & Zhang, Ya-Qin 2002. Video processing and communications. New Jersey: Prentice Hall.

Wikipedia 2011a. Tiedonsiirtonopeus. WWW-dokumentti. <http://fi.wikipedia.org/wiki/Tiedonsiirtonopeus>. Päivitetty 28.09.2011. Luettu 23.11.2011.

Wikipedia 2011b. Väylä. WWW-dokumentti. <http://fi.wikipedia.org/wiki/Väylä>. Päivitetty 10.11.2011. Luettu 23.11.2011.

Wikipedia 2011c. Webcast. WWW-dokumentti. <http://en.wikipedia.org/wiki/Webcast>. Päivitetty 18.11.2011. Luettu 22.11.2011.

Wowza 2009. Press Release. WWW-dokumentti. <http://www.wowza.com/2009-12-17.html>. Päivitetty 17.12.2009. Luettu 8.11.2011.

KASSUN KARNEVAALIT 7.9. -10.9.2011

TUOTANTOKÄSIKIRJA

Tuotannon sisältö	2
YHTEYSTIEDOT	2
Tuotannon kulku	3
Kamerat ja kuvakoot	3
Äänet	4
Varustelista	4

Tuotantokäsikirja Kassun karnevaalit**Tuotannon sisältö**

Mikkelin ammattikorkeakoulu tuottaa Informaatio- ja mediatekniikan laitoksen opiskelijatyönä osana opiskelijoiden opinnäytetyötä 7.-10.9.2011 monikameratuotannon Kassun karnevaaleista. Kassun karnevaalit järjestetään Kasarmin kampuksen paraatikentälle pystytettävässä n. 1000 hengen juhlateltassa. Tarkoituksena on kuvata keskiviikkona 7.9. MAMK:n 20-vuotisjuhla, joka alkaa klo 19:00 ja päättyy 21:00. Tämä tapahtuma streamataan CampusTV:n verkkosivujen kautta. Perjantaina 9.9 kuvataan Osmos Cosmoksen keikka, josta tehdään tallenne. Keikka on merkitty aika välille klo 21 – 24. Lauantaina kuvataan Paukkumaissin esiintyminen klo 11 – 14, josta myös tehdään tallenne. Kaikki tapahtumat kuvataan kolmella kameralla.

YHTEYSTIEDOT

Tuotannon vastuuhenkilöt / MAMK

Tomi Numento yhteystiedot

Tuotannon yhteyshenkilöt / asiakas

Olli Järvenkylä yhteystiedot

Pete Kiisseli /Osmos Cosmos yhteystiedot

Kari Hulkkonen /Paukkumaissi yhteystiedot

Tuotantotiimi ja vastualueet

Mikpoli:

Tomi Numento yhteystiedot

Tuotantokäsikirja Kassun karnevaalit

Janne Saarela	ohjaaja/kuvamiksaus	yhteystiedot
Jani Töllikkö	tekniikka/kamera x	yhteystiedot
Timo Vainikka	kamera x	yhteystiedot
Jarkko Kerminen	kamera x	yhteystiedot
Juha Kärkkäinen		yhteystiedot
Henri Romo		yhteystiedot

Tuotannon kulku

Tiistai 6.9.2011

- Tavaroiden tarkistus, testaus ja pakkaus
- Akut lataukseen
- Serverin tyhjennys?

Keskiviikko 7.9.2011

- Tekniikan rakentaminen ja testaus
- 20-vuotisjuhlan kuvaus klo 19-21
- Tekniikan purku

Perjantai 9.9.2011

- Tekniikan rakentaminen ja testaus
- Osmos Cosmoksen keikan kuvaus klo 21-24
- Tekniikan purku

Lauantain 10.9.2011

- Tekniikan rakentaminen ja testaus

Tuotantokäsikirja Kassun karnevaalit

- Paukkumaissin keikan kuvaus klo 11-14
- Tekniikan purku

Kamerat ja kuvakoot

CAM1

Ykköskamera (Hitachi) sijaitsee n. 20 metrin päässä lavasta miksauskopissa (miksauskopin viereen rakennetulla korokkeella) mahdollisimman lähellä äänipöytää, jotta äänet voidaan vetää kameraan mahdollisimman pienellä vaivalla. Kuvissa on olennaista, että koko lava tulee aluksi kuviin, muutoin kuvakoko rajataan mahdollisimman tiukaksi (n. LKK). Ykköskameralla yritetään saada myös näyttävät valot mukaan.

CAM2

Kakkoskamera (Hitachi) sijoitetaan yleisöstä katsottuna lavan vasempaan reunaan, sille rakennetulle korokkeelle. Sen pääasiallinen tehtävä on poimia mielenkiintoisia yksityiskohtia kuviin. Kuvakoot ovat puolilähikuvasta aina erikoislähikuvaan. (PLK - > LK -> ELK) Erityisesti soolot ym. tulisi saada poimittua kakkoskameralle. Kameramiehen tulee suojata kuulonsa korvatulpilla ja/tai melusuojilla. Komentoyhteys siihen korvaan, joka on pois päin lavasta.

CAM3

Kolmoskamera (Hitachi) sijoitetaan yleisöstä katsottuna lavan oikeaan reunaan (screeniä lähinnä), sille rakennetulle korokkeelle. Tällä kameralla kuvataan melko laajaa kuvaa, yleisön reaktioita sekä kokokuvasta aina puolikuvaan ja puolilähikuvaan (KK -> LPK -> PK -> PLK). Kolmoskamera keskittyy kuvaamaan laulajia. Kuvaajalle korvatulpat ja komentoyhteys kuten kamerassa 2.

Kamerat 2 ja 3 mukautuvat tehtävissä ohjaajan komentoihin ja valoisuusolosuhteisiin. Esimerkiksi jos lavan valot osoittaa suoraan kameraan tietyssä kuvakulmassa, pyritään välttämään sitä.

Tuotantokäsikirja Kassun karnevaalit

Äänet

Äänet otetaan suoraan päälavan miksauspöydästä master out -linjasta xlr-kaapelilla kamera ykkösen kautta. Yleisöäänet saadaan mikittämällä 2 tai 3 kamera Shuren mikrofonilla. Äänimiehen tulee tarkistaa, että programmiin ajettava ääni tulee oikeasta linjasta.

Varustelista

Kamera #1:

- Kamera
- 19 x zoom
- 4 johtoa (remote, signal, xlr (aux video), gen lock)
- Näyttö ja sen kiinnike
- Virtalähde ja virtajohto (15m)
- Jalustalevy
- Käyttöohjaimet 2kpl (Zoomi ja Tarkennus)
- Linssin suoja
- Copperhead ja akku
- Headset ja siihen ruskea välikappale
- Kuitujohtoa (SMF) kela
- Audio (XLR-XLR) 10m
- Jalusta ja kahvat x 2

Kamera #2:

- Kamera
- 12 x zoom
- 4 johtoa (remote, signal, xlr (aux video), gen lock)
- Näyttö ja sen kiinnike
- Virtalähde ja virtajohto (10m)
- Jalustalevy
- Käyttöohjaimet 2kpl (Zoomi ja Tarkennus)
- Linssin suoja
- Copperhead ja akku

Tuotantokäsikirja Kassun karnevaalit

- Headset ja siihen ruskea välikappale
- Kuitujohtoa (SMF) kela
- Audiojohto (XLR) 10m
- Jalusta ja kahvat x 2

Kamera #3:

- Kamera
- 19 x zoom
- 4 johtoa (remote, signal, xlr (aux video), gen lock)
- Näyttö ja sen kiinnike
- virtalähde ja virtajohto 5m
- käyttöohjaimet (zoomi)
- Jalustalevy
- Linssin suoja
- Copperhead ja akku
- Headset ja siihen ruskea välikappale
- Kuitujohtoa (SMF) n.10m (irtonainen)
- Jalusta kahvoineen

Muut:

- ”Inttikuitu”
- Kuitujen tarvikesalkku ja kuituputsain
- Telecast Teleport –mediamuunnin
- PagLoc akkuja 6kpl ja laturi
- Mikrofoni ja teline
- Virtajohtoja
- Nippusiteitä
- Roudarinteippi
- Muovipusseja

Varalle:

- Koaksaalikaapeli (BNC)
- Kuitujohtoa (SMF) pitkä, sininen kela
- Virtajohtoja

Tuotantokäsikirja Kassun karnevaalit

-Audiojohtaja (XLR)

Kuvausohje, jos et kuule komentoja

Aloituskuva kamera 1:stä. Laaja kuva, koko lava näkyy. Zoomaus sisään kokokuvaksi bändistä/juontajasta.

Bändin lavalle tulo kuvataan kameralla 2 tai 3.

Kitara- ym. soolot kameralla 2. Lähikuvaa, erikoislähikuvaa, voi leikkiä fokuksella, jos saa varmasti skarpin lopuksi kohdalleen.

Yleisön huudatukset/laulatukset ym. kameralla 2 ja 3. Koko- ja puolilähikuvaa, pannauksia.

Kameralle esiintymiset / kitaristit kepittää vastakkain yms. näkyviin. Poimi mielenkiintoisia ja vaihtelevia kuvia.

Setin lopuksi kamera 1 zoomaa hitaasti auki, koko lava näkyviin, jonka jälkeen siirrytään grafiikoihin.

Tarjoile vaihtelevaa kuvaa. Myös liikettä ja fokuksen kanssa leikittelyä. Jos olet kuvannut samaa kuvaa jo 15 sekuntia, on korkea aika vaihtaa sitä!