



Mediator

Sammendrag

Tittel:	«Mediator»
No/Date:	1/19. mai 2005
Authors:	Andreas Hveem Bakken Lars Jonas Pedersen
Veileder:	Claus J.S. Knudsen
Oppdragsgiver:	Høgskolen i Gjøvik
Kontaktperson:	Claus J.S. Knudsen
Nøkkelord:	Øyekontakt, videokonferanse, prescense production
Pages:	89
Tilgjengelighet:	Åpen
Vedlegg:	1

Abstrakt

I dette prosjektet er det bygget og videreutviklet en «Mediator». Dette er en applikasjon som har til hensikt å senke brukerterskelen og ved ulike parametere , og gjøre det mer attraktivt å bruke videomediert kommunikasjon, videokonferanser, som en alternativ møteform.

Mediatoren bygger på blandt annet av I2i-teknologien, som er utviklet ved KTH, Stockholm. Denne gjør at man kan oppnå øyekontakt til de andre partene i møtet.

Prosjektet har også utviklet en programvare som har til hensikt å hjelpe brukeren til enkelt å kunne styre mediatores parametre. Applikasjonen har mulighet for både å sende og motta flere typer video og lydskilder.

Mediatoren er utviklet med tanke på å benytte seg av aktuell teknologi for overføring av video og lyd. Dette kan være TCP/IP, ISDN men også fiberoptisk overføring.

Summary

Title: «Mediator»
No/Date: 1/19. May 2005
Authors: Andreas Hveem Bakken
Lars Jonas Pedersen
Supervisors: Claus J.S. Knudsen
Employer: Gjøvik University College
Contact person: Claus J.S. Knudsen
Keywords: Eye to eye, videoconferencing, presense production
Pages: 89
Aviability: Open
Attachments: 1

Abstract

This project has further developed an «Mediator»-application. The application's goal is to enhance the useability, and there by make the use of video mediated meetings, videoconferencing, more attractive.

The «Mediator»-application makes use of the I2i-technology developed at The Royal Institute of Technology (KTH), Stockholm. The technology uses the principle from teleprompts known from TV-broadcasting, and makes the user able to produce eyecontact.

The application is able to handle multi-channel inputs and outputs, witch makes it possible to handle multiple in- and output units. It's able to transfer video and audio streams using technologies as the Internet, ISDN and fiberoptics.

The project has also developed a software wich in an easy way makes the user able to control the parameters of the application.

Forord

«Mediator», eller «Media Tor», som den også ble kalt av medstudenter, er arbeidstittelen på videokonferanse-applikasjonen som er utviklet i dette prosjektet. Den består både av en hardware- og software del. På hardwarensiden har vi benyttet oss av I2i-teknologi, som gjør at applikasjonen kan produsere øyekontakt. I tillegg er det en en mutlimediamatrise, som gjør at man kan distribuere og rute signaler fra ulike inn-enheter til ut-enheter. Enhetene kan f.eks. være kameraer, prosjektører, lyd/videobehandlings-løsninger, datamaskiner, CD-spillere, mikrofoner og ulike videokonferanse-codecer – med andre ord alle enheter som har mulighet for avspilling og opptak. Oppdragsgiver til prosjektet har vært medielaboratoriet på HiG, ved Claus J.S. Knudsen, som er laboratorie ansvarlig.

Rapporten inneholder oversikt over arbeidet prosjektgruppen gjennomførte i løpet av prosjektperioden vårmesteret 2005. I tillegg til å dokumentere arbeidet har det for oss vært minst like viktig å få dokumentert og samle de fleste små og større oppdagelser vi gjorde. På denne måten håper vi en eventuelt videreutvikling ikke skal måtte lete opp igjen stien vår og ikke minst gå i de samme fellene vi gjorde. I stedet håper vi rapporten kan gjøre at en ny marsj kan starte der sporene våre ender, eller at stien kan også benyttes for å nå helt andre mål.

Vi vil rette en spesiell takk til de tre sentrale samarbeidspersonene i prosjektet: Claus for hjelpen underveis, og for at du lot oss få et lærerikt innblikk i akademien. Vi vil også takke Mats Erixon og Charlie Gullström for at de så åpent tok oss i mot, og delte sine kunnskaper til oss.

Gjøvik 19.05.2005

Andreas Hveem Bakken

Lars Jonas R. Pedersen

Innhold

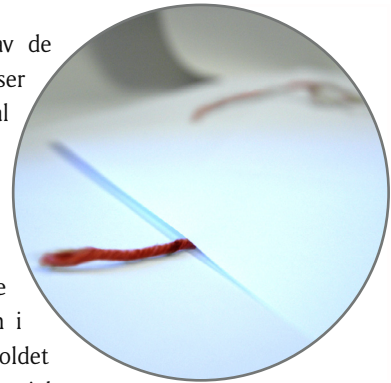
Sammendrag	3
Summary	5
Forord	7
1 Innledning	13
1.1 Organisering av rapporten	13
1.2 Målgruppe for rapporten	13
1.3 Gruppemedlemmenes bakgrunn	14
1.4 Prosjektoppgaven	14
1.5 Rammer og forutsetninger	16
1.6 Arbeidsmetode	20
1.7 Kvalitetssikring	20
2 Kildesøk og forstudier	23
2.1 Ulike skjermtyper	23
2.2 Øyekontakt	25
2.3 I2i teknologien	26
2.4 Speilvending	28
2.5 Fotograferer transmisjon i speiltypene	30
2.6 Fotografering av refleksjon	32
2.7 Dobbelspeiling	33
2.8 Måling med transmisjonsdensitometer	35
2.9 Kamera	36
2.10 Kommunikasjonsenheter	37
2.11 Analog overføring av video og audio	39
2.12 Matrise	41

2.13 Planlegging av konstruksjon	43
2.14 Miljøet rundt mediatoren	46
2.15 Kommunikasjon	47
2.16 Videokonferanser	48
2.17 Programmering	49
2.18 GUI	50
3 Utførelse	53
3.1 Software	53
3.2 Hardware	60
3.3 Implementering	65
3.4 Økonomi	67
3.5 Mål som ikke ble nådd	68
4 Evaluering/diskusjon	71
4.1 Prosjektgjennomføringen	71
5 Konklusjon	75
5.1 Gruppen	75
5.2 Virtuelle rom	75
6 Fremtid/visjon	79
8 Ordliste	85
9 Vedlegg	89

1 Innledning

1.1 Organisering av rapporten

Hovedinnholdet av rapporten består av de tre kapitlene innledning, forundersøkelser og utførelse. Innledningskapittelet skal gi leseren innsikt i bakgrunnen og rammene prosjektet opererte innenfor. Det neste tar for seg undersøkelser vi gjorde før vi kunne starte å utvikle selve applikasjonen – før en til slutt kan følge selve sammenføyingen av applikasjonen i kapittelet «utførelse». Etter hovedinnholdet i rapporten kommer en evaluering av prosjektet, samtidig som vi forsøker å se for oss den videre veien for denne type applikasjoner. Etter kildehenvisningen avsluttes rapporten med ordliste og figurliste, som skal hjelpe leseren til å forstå og finne frem i rapporten.



1.2 Målgruppe for rapporten

Rapporten har en tredelt målgruppe. I og med at prosjektet er et hovedprosjekt som skal sensureres, er det naturlig å skrive rapporten slik at sensor får innsikt i arbeidet, valg og utfordringer vi møtte i prosjektet. Videre håper og tror vi at rapporten kan være til hjelp for de som skal videreutvikle og skape lignende applikasjoner. Til sist håper vi andre som jobber med videokonferanser og andre former for møter i ikke fysiske rom, kan finne relevant informasjon i rapporten.



1.3 Gruppemedlemmenes bakgrunn

Gruppens to medlemmer studerer til grafisk- og dataingeniør ved Høgskolen i Gjøvik (HiG). Lars Jonas går på studieretning for digital medieproduksjon og Andreas går på studieretning for programutvikling.

Begge har erfaring fra musikk, som både musikere og med aktuelt teknisk utstyr. I tillegg har Lars Jonas jobbet en del med video. Disse erfaringene var svært viktige for at vi kunne ta på oss et prosjekt, som involverer så mange forskjellige fagområder.



1.4 Prosjektoppgaven

1.4.1 Bakgrunn

Fravær av øyekontakt er et stort problem ved dagens videokonferanse-løsninger, noe som skyldes at kameraet som regel er plassert på toppen eller ved siden av skjermen. For å få en naturlig kommunikasjon er øyekontakten essensiell. Ved Kungliga Tekniska Högskolan (KTH), Stockholm har det blitt utviklet teknologi der de oppnår øyekontakt ved å bruke teleprompt-prinsippet fra nyhetsendinger.

Der får nyhetsoppleseren teksten opp foran kamera, slik at han kan lese teksten samtidig som en ser i kamera. Gjennom å se direkte i linsen på kameraet oppfatter seeren det som om man ser brukeren i øynene, og får følelsen av av man prater direkte til seeren. I kjølevannet av dette er det også blitt etablert et foretak, som er kommet langt i kommersialiseringen av en applikasjon basert på denne teknologien. Gjennom Mediator-prosjektet vil vi bringe denne teknologien til HiG, og Norge, og få en operativ applikasjon ved multimedielaboratoriene ved HiG. I tillegg til å benytte den allerede eksisterende teknologien, ville vi i samarbeid med KTH videreutvikle applikasjonen, spesielt med tanke på å utvikle et softwarebasert brukergrensesnitt



1.4.2 Problemstilling

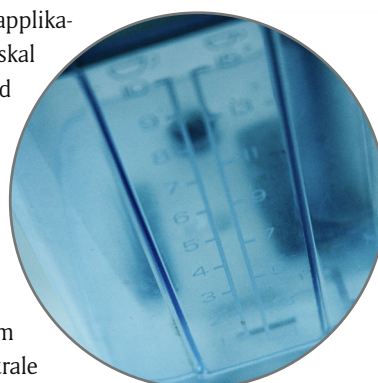
Hvordan kan vi videreutvikle en applikasjon som gjør at personer med minimal teknisk kompetanse kan kommunisere og føle nærhet i et virtuelt rom? Hvordan gjøre den tekniske «veggen» mellom de fysiske rommene i dette møtet så transparent som mulig? Hvordan kan brukeren på en enkel måte kontrollere applikasjonen?



1.4.3 Resultatmål

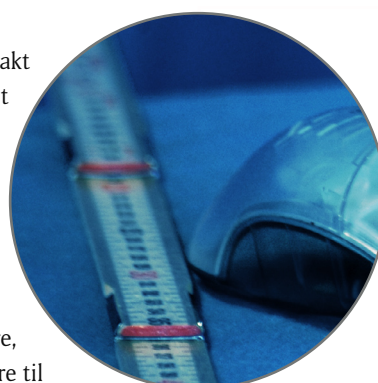
Gjennom prosjektet ønsker vi å utvikle en applikasjon for virtuelle møter. Applikasjonen skal være en naturlig del av læringsmiljøet ved HiG, og skal være operativ tjuetimer i døgnet. Hovedmålet i prosjektet ligger i å skape en operativ applikasjon som er enkel i bruk og faller naturlig inn i det allerede eksisterende fysiske miljøet. Det skal også utvikles en programvare, der brukeren så enkelt og intuitivt som mulig skal kunne kontrollere de sentrale parametrene i applikasjonen.

Ved hjelp av I2i-teknologien som er utviklet ved KTH, ønsker vi også å la brukerne oppnå øyekontakt med partene i videokonferansen.



1.4.4 Effektmål

Dersom brukerne oppnår øyekontakt vil dette gjøre det medierte møtet til et sterkere møte, og man vil løfte denne moteformen nærmere det fysiske møtet – man kan si at prosjektet ønsker å gjøre den teknologiske veggen mellom partene mer transparent. Dersom man samtidig med dette gjør interaksjonen mellom brukeren og applikasjonen enklere, i et mer tilpasset miljø, tror vi dette vil føre til at nye brukergrupper vil ta i bruk virtuelle møter, i dette tilfellet videokonferanser.



1.4.5 Målgruppe for applikasjonen

Vi ønsker at applikasjonen skal kunne brukes av personer som tidligere ikke har kjent seg kvalifisert til å bruke videokonferanser. Dette innbefatter for eksempel personer med liten erfaring i bruk av datamaskiner. Samtidig er det viktig at de mer erfarne brukerne kan operere applikasjonen effektivt uten å føle for store begrensinger.



1.5 Rammer og forutsetninger

1.5.1 Økonomi

Da vi i desember 2005 takket ja til prosjektet, var de økonomiske rammene ganske usikre. Både på kostnads- og finansieringssiden var det fremlagt diffuse tall. Mange prosjekter er usikre på kostnadsiden, men i dette prosjektet var det også stor usikkerhet på finansieringssiden. Vi prøvde så raskt som mulig å skaffe en oversikt over situasjonen.



Vi satte opp to kostnadsnivåer, som hver for seg representerte ytterpunkter. Med tanke på utstyr og materialer til prosjektet, havnet vi da på en minimumskostnad på ca. ni tusen kroner, og et øvre kostnadsnivå langt over hva noen av våre budsjetter kunne romme. Etter samtaler med veileder, var ni tusen en kostnad som kunne dekkles. Dette utgjorde nivå 1. Vi satte i tillegg opp et budsjett for nivå 2, som var det høyeste realistiske nivået. Nå var det opp til oss å finne de beste og billigste løsningene, i tillegg til å jobbe for finansiell støtte til prosjektet. Etterhvert som det ble klarhet i økonomien, ville vi sette opp budsjett for et nivå mellom 1 og 2.

I starten var strategien å utvikle mediatoren i samarbeid med næringslivet, slik at de i tillegg til å bidra i med innspill i utarbeidelsen av applikasjonen, kunne bidra i finansieringen av prosjektet. Etter brev til kontakter som veileder hadde med næringslivet, viste det seg at interessen for prosjektet var lunken. Vi valgte derfor ikke å involvere næringslivet i den grad, men heller fokusere på å få en operativ mediator som var tilpasset behovet ved HiG. At vi i starten prioriterte å involvere næringslivet, gjorde at det gikk relativt lang tid før vi fikk avklart videre gang i prosjektet, og at vi havnet et stykke etter tidsplanen.

1.5.2 Utstyr

Applikasjon

Ved prosjektets oppstart hadde skolen en del utstyr vi kunne benytte i applikasjonen. Noe av utstyret var ganske gammelt, men det var vår oppgave å kartlegge hvorvidt utstyret kunne brukes. Vi hadde tilgang til:

- Sony SB-V3000, multimediamatrise for å rute audio og videosignaler til og fra de ulike enhetene. Applikasjonen vår skulle ha en slik matrise, som kontrolleres fra det grafiske brukergrensesnittet på PC.
- Dataskjermer. Skolen har en del dataskjermer som ikke er i bruk. Vi måtte finne ut om hvorvidt disse kunne brukes i prosjektet.
- Multimedielaboratoriene ved HiG har to Tandberg 800 videokonferanse-enheter, der den ene kunne benyttes i applikasjonen.
- Vi fikk også mer eller mindre arve en «PolySpan Streamstation» fra den gamle «Avdeling for Helsefag». Dette er en datamaskin som er bygd for arkivering og sanntid streaming av videokonferanser.
- Til de ulike forsøkene hadde vi tilgang til alt utstyret på medielaboratoriene ved HiG.
- Når applikasjonen skulle implementeres i miljøet ved multimedielaboratoriene, er det i laboratorientes U-lab allerede lagt opp kabler for lyd-, bilde- og digitale signaler.



Kontor

Vi var heldige og fikk tildelt et innredet grupperom, som vi delte med gruppen «X-Ray NOW». Vi vil takke Maria, Amund og Tone for at de holdt ut med oss – Selv i harde tider, som når vi stekte pizza-skjæderen, og røykla hele kjelleren.

Av skolen fikk vi låne to Apple G3 400 MHz med 256 MB. minne, som etterhvert ble til én med 512 MB minne.

Vi brukte også eget utstyr i arbeidet. Dette er utstyr vi kjente og var komfortable med fra før. Dette var bl.a. Andreas sin egen PC – en Whitebox CL 50, 1,6 GHz, 1024 mb. minne, med to skjermer, som ble brukt til programutviklingen. Av annet utstyr brukte vi to kamera, Canon IXUS 40, og Nikon D70.



1.5.3 Kommunikasjonsmetoder

I prosjektets gang har vi brukt mange forskjellige måter å kommunisere på. I tillegg til de tradisjonelle metodene som telefon, brev og epost, har vi også brukt tjenester som MSN messenger, Tandberg videokonferanseenheter og vi fikk vårt første møte med Marratech. Marratech er et program for å gjennomføre videokonferanser på Internett, både som unicast og multicast. Det er utviklet og blir stadig videreutviklet i samarbeid med universitetet i Luleå.



1.5.4 Prosjektets nettsted

(http://hovedprosjekter.hig.no/v2005/data/gr15_mediator/)

Nettstedet vårt har vært delt inn i to deler. En offentlig tilgjengelig del, der en i grove trekk kan lese hva prosjektet går ut på, følge progresjonen i prosjektet, og informasjon om personene som er involvert i prosjektet. Den andre delen av nettstedet har vært passordbeskyttet.

Denne delen brukte vi som et verktøy til å lagre status i arbeidet slik at dette, i tillegg til å være en kontinuerlig oppdatert statusrapport, ble en logg vi kunne bruke når vi skulle skrive rapport. Grunnen til at vi passordbeskyttet denne delen, var at vi her skrev løse interne, og ikke alltid like godt formulerte notater.

Siden vi jobbet litt her og der, dvs hjemme, pc-laboratoriene, publiseringslaboratoriet, og på kontoret brukte vi området til å laste opp filer vi jobbet med. På denne måten kunne vi hente de ned fra hvor det skulle være, samtidig som vi fikk samlet de fleste filene på en og samme plass, som en backup.

Vi valgte ikke å legge sidene på webtjeneren vi fikk tildelt fra HiG. Dette fordi denne hadde for store begrensinger ifh. til bruken vi la opp til. Vi hadde f.eks. erfart at det ikke var enkelt å få tilgang til skolens servere utenfor skolens nettverk, og at man under studietiden vår har hatt problemer med å få en stabil serverpark. Skolen kunne heller ikke tilby en egen e-post konto, bare videresending til våre personlige epostkontoer. Løsningen ble at vi fikk gratis webhotell og epostkonto på nett-tjeneren artsys.no, som tilhører gang Interaktiv. Fra leverandør er det på denne en garantert oppetid på 99,3%, og de siste årene har oppetiden vært på 99,9%. Til å administrere nettsiden brukte vi publiseringsløsningen ArtSys, som Lars Jonas har vært med å utvikle for gang Interaktiv. Til bruk i løsningen brukte vi en mysql-database vi fikk opprettet hos IT-tjenesten ved HiG.



1.5.5 Oppdragsgiver

Hovedoppdragsgiver er medielaboratoriene ved HiG. Kontaktperson har vært identisk med vår veileder Claus J.S. Knudsen. Prosjektet har også vært sterkt knyttet opp mot samarbeidet mellom KTH ved Mats Erixson og Gullström Architects AB som sammen har stått for utviklingen av I2i-teknologien vi benytter. Endel av forsøkene er gjort på oppfordring fra disse, og de vil videre også få tilgang til å benytte alt hva prosjektet skaper.



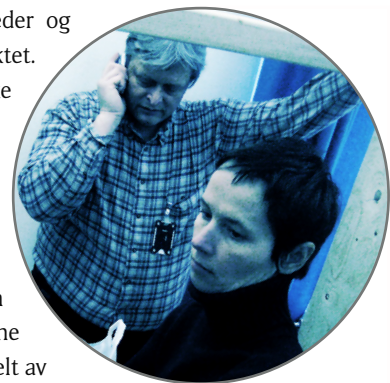
1.5.6 Tidsressurser

Prosjektet startet i januar 2005, og prosjektrapporten ble levert 19. mai, med påfølgende presentasjon i perioden 25. - 27. mai. Se vedlagte GANNT-skjemaer som viser planlagt prosjektgang. Prosjektet er satt til 20 studiepoeng. Dette tilsvarer ca 25-30 uketimer pr. person i et semester. Dvs at vår gruppe, som består av to personer, er antatt å bruke rundt 60 uketimer.



1.5.7 Ressurspersoner

Claus J. S. Knudsen har vært vår veileder og en enorm støtte gjennom hele prosjektet. Når det oppstod problemer han ikke direkte kunne hjelpe oss med har han svært velvillig delt sitt nettverk av ressurspersoner med oss. I tillegg til veileder har de viktigste ressurspersonene i prosjektet vært Mats Erixson og Charlie Gullström, som arbeider med utviklingen av applikasjonene ved KTH. De har tatt oss inn i varmen, delt av sin kunnskap, og virkelig engasjert seg i arbeidet vårt.



1.6 Arbeidsmetode

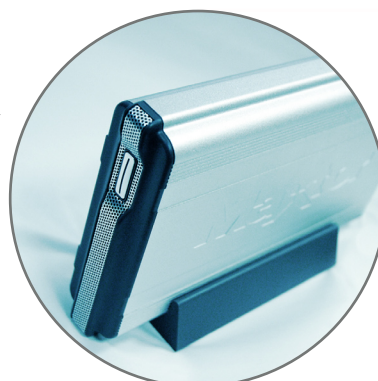
Vi har forsøkt å gjennomføre prosjektet som en iterativ arbeidsprosess. Etterhvert som arbeidet siger frem, ser man ting tidligere i prosessen vi kunne gjort bedre. I en iterativ prosess tar man stegene tilbake for å gjøre disse forbedringene, ofte flere ganger, slik at man stadig får bedre resultat. Motsetningen til den iterative prosessen er en fossefalls-tankegang. Da gjør man helt ferdig en del, for man går videre. I studietiden har vi ikke jobbet så mye i store prosjekter, og heller ikke fokusert så mye på arbeidsprosesser, så vi forventet ikke å klare å gjennomføre dette 100%, men bruke tankegangen som en plattform.



1.7 Kvalitetssikring

1.7.1 Backup

Vi har tatt backup etter behov. Som lagringsmedium for backupdataene har vi brukt hverandres datamaskiner, og eksterne harddisker. Mindre- og filer med stor betydning har vi også tatt backup av på nettstedet vårt på artsys.no



1.7.2 Sosialt/arbeidsmiljø

I starten av prosjektet reiste vi på ekskursjon til KTH i Stockholm for å få en innføring i den aktuelle tekologien vi skulle bruke. Turen ble også svært viktig for å bli godt kjent og etablere et godt sosialt miljø i gruppen. Når man er bare to stykker, er det spesielt viktig at en vet hvor en har hverandre, og får oversikt over hverandres sterke og svake sider. Derfor reiste vi til Stockholm to dager før vi skulle møte på KTH. Vi brukte dagene til mye sosialt





Andreas nyter frokosten på en av Stockhoms mange spisesteder.

og ble også kjent, om enn noe overfladisk, med byen Stockholm. Vi bodde da hos Nils Kristian Raddum, som stilte hybelen sin til rådighet. Andreas sov på gulvet, mens Lars Jonas sov i en stressless under oppholdet. Vi vil oppfordre alle som skal gjennomføre et hovedprosjekt, til tidlig å prioritere en periode der en må bo oppå hverandre, dvs. bli kjent på godt og vondt. Dette kommer uansett til å skje i løpet av prosjektet.

For å bevare det gode sosiale miljøet i gruppen har vi stort sett hver fredag



Vi fikk med oss tre konserter. Her på jazzclub, og Reaggekonsert.



Studentene hadde akekonkurranse på KTH.

spandert på oss fellesmiddag og minimum en pakke twist – og under måltidet har vi diskutert statusen i prosjektet og arbeidet framover. Disse har fungert som små statusmøter, som har gjort at vi har fått oversikten og sett progresjonen siden sist. Ved å se fremgang i prosjektet, får en den lille ekstra motivasjonen, som absolutt kan være nødvendig når man sitter fast i en if-setning, eller datamaskinen sier takk for seg, når det er en time siden forrige lagring.

Samarbeidet med veileder har også utviklet seg til å bli svært åpent. I starten hadde vi faste møter annenhver onsdag, men ettehver som arbeidet kom igang ble det mer uformelle samtaler der døren til Claus stod åpen når vi trengte det. Det var likevel viktig å holde på noe av strukturen der en hadde mer formelle møter der en satte seg ned for å diskuterte prosjektet i sin helhet.



Andreas og Nils-Kristian studerer vårmoten på NK.



Tre Kronor om natten...



En litt forfrost Totning på veg til Vasa-museet

1.7.3 Språkvask

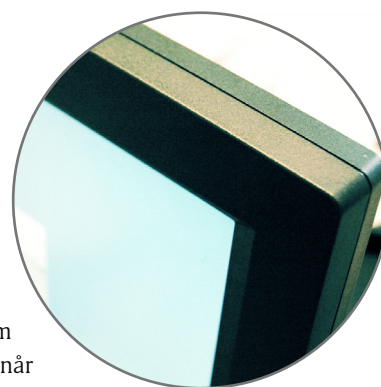
I tillegg til selv å lese korrektur har vi fått god hjelp fra Helge Persersen og Kari Rønnestad – foreldrene til Lars Jonas, som har brukt flere vårkvelder på å rette, ikke alltid like godt formulerte avsnitt. Prosjektet har vært svært hektisk, og dette har ført til at vi ikke har hatt tid til å finpusse språket til den minste detalj. I formuleringen av rapporten har vi forsøkt å formulere oss slik at det teksten blir lett og forståelig å lese. I skriveprosessen har vi erfart at kvaliteten på formuleringene og språket stiger parallelt med tiden en bruker.

2 Kildesøk og forstudier

2.1 Ulike skjermtyper

2.1.1 Bakgrunn

Vi tok utgangspunkt i tre typer skjemer LCD, Plasma, og CRT (billedrørsTV). For at kommunikasjonen i videokonferanser skal være god, er det viktig at det er minst mulig forsinkelse i signalene mellom partene. Det kan en for eksempel merke når nyhetene på TV intervjuer direkte fra steder langt borte. Når det gjelder skjermer er det også en liten forsinkelse fra signalet kommer, til det blir vist som bilde på skjermen. For at den totale forsinkelsen skal bli minst mulig, må en altså velge en skjerm med kortest mulig responstid. En annen faktor er hvor lyssterk skjermen er. Mediatoren skal helst kunne brukes under de fleste lysforhold. Derfor er det viktig at skjermen gir nok lys, slik at bildet ikke forsvinner under lyse omgivelser.



2.1.2 Utførelse

For å finne fakta om temaet gjorde vi forespørsler mot leverandører og søk på Internett. Vi har ikke hatt ressurser til å verifisere denne informasjonen, så den må stå for egen regning. Hovedsakelig brukte vi informasjonen som vi hentet hos uavhengige kilder. Informasjon fra produsenter og leverandører er som regel tenkt brukt til salg og markedsføring –og disse stemte ikke alltid overens med informasjonen vi fant hos uavhengige kilder.

2.1.3 Resultat

LCD

I et diskusjonsforum på Internett fikk vi opplyst at det på LCD-skjermer som regel er sammenheng mellom responstid og lysstyrke – altså går responstiden opp parallelt med lysstyrken.

LCD vs. TFT

Dinside.no tar i en artikkel for seg bl.a. forskjellen mellom LCD og Plasma, der går det fram at responstiden (etterglød i pikslene) på de forskjellige typene er ganske forskjellig. På plasmaskjermer ligger den rundt 1,5 mikrosekunder, mens den på LCD-skjermer ligger på mellom 12 og 30 millisekunder, der det generelt er slik at responstiden blir lengre jo lavere prisen er.

LCD vs. Billedrørs-TV

I den samme artikkelen tar man for seg forskjellen mellom LCD og vanlige billedrørs-TVer. Som en kan se av bildene til venstre, er det relativ stor forskjell i kontrastforhold. I tall sier artikkelen at det på LCD er et kontrastforhold på 500:1, og billedrør har 3000:1. Hva fabrikantene oppgir skal man ta med en klype salt – for LCD skjermer kan bli oppgitt med kontrastforhold opptil 3000:1, men de beste er ifølge dinside.no målt til 800:1.



Øverst et bilde av et bilde på en LCD-skjerm. En kan se en markant forskjell i kontrastforholdet mellom denne og CRT-skjermen under.

2.1.4 Konklusjon

Da applikasjonen kommer til å være plassert i kritiske lysforhold var valget ganske enkelt; Bruk av billedrørs-TV. Billedrør er også betraktelig billigere enn andre alternativer. Nå får man de også med flat skjermflate, så de negative sidene er kun at TVen er større og tyngre – og dette var ikke noe stort problem i vår konstruksjon.

2.2 Øyekontakt

2.2.1 Bakgrunn

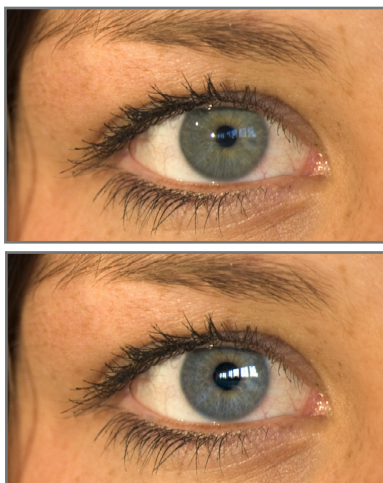
En av de store forskjellene mellom mediatoren og andre videokonferanse-løsninger er at deltakerene faktisk har øyekontakt med hverandre under samtalen. Vi ville derfor finne ut hvor stor kraft øyekontakt har i forhold til andre deler av kroppspråket, og hva øyekontakt betyr for kommunikasjonen mellom mennesker.



2.2.2 Utførelse

Vi gikk på biblioteket ved HiG. Her ble vi tatt imot av en vennlig bibliotekar som viste oss seksjonen for bøker om kroppsspråk. Vi valgte å ta for oss boken "Vis meg ditt kroppsspråk – og jeg skal si deg hva du oppnår" [REF], og fordypet oss i kroppspråkets betydning – da med spesiell interesse for øyekontakt.

2.2.3 Resultat



En stor og liten pupille

Språket øynene snakker består hovedsaklig av to deler; blick og pupillene.

Pupillene endrer seg uavhengig av kroppen. De fleste vet at de utvider seg i mørke, og blir mindre i sterkt lys, men pupillene blir også påvirket av følelsene. Dersom vi blir opphisset, glade eller engstelige, vil pupillene utvide seg. I motsatt fall, når vi føler noe er ekkelt, vil pupillene trekke seg sammen. Endringene av pupillenes størrelse skjer helt uten at vi bevisst kan styre de, men ved hjelp av kunstig stimuli, som f.eks rusmidler, kan man motvirke eller fremkalle disse endringene. At pupillene utvider seg, blir like ubevisst oppfattet. Vi oppfatter disse signalene om vi vil eller ikke. Disse endringene skjer også i møter mellom mennesker. Møter vi et smilende menneske, og pupillene er store, er det ganske sikkert at personen er oppriktig glad for å se oss. Er de derimot små, kan det være at smilet er påtatt. Blick kan fortelle mye på kort tid over store avstander, men hvordan vi oppfatter blikket varierer mellom kulturer. I vår kultur ser vi hverandre i øynene et par sekunder av gangen, og når vi prater går blikket stort sett mellom øynene og munnen. I løpet av et minutt, er vi programmert til å ha mellom 30 og 40 sekunder med blick-kontakt. Når noen har lite blick-kontakt med den man prater med, er det fordi en på en eller annen måte føler seg ukomfortabel; vi sitter da enten for nær hverandre, eller vi prater om noe som er vanskelig. Når den andre parten oppfatter denne mangelen på blick-kontakt, begynner man med en gang å analysere personen; er personen sjenert, engstelig, uninteressert, feig eller ikke troverdig? Men ser vi for lenge, kan dette bli oppfattet som ubehagelig stirring.

Stirring kan tolkes som et forsøk på å vise makt, eller mangel på respekt.



2.2.4 Konklusjon

Øyekontakt er en svært viktig faktor i kommunikasjon, og det er et stort fremskritt å oppnå dette i videokonferanser. Øyekontakt kan gjøre budskapet en legger frem mye klarere, sterkere, og samtidig avsløre graden av troverdighet. Øyekontakt vil også bringe videokonferanser nærmere det fysiske møtet.

2.3 I2i teknologien

2.3.1 Bakgrunn og utførelse

På vår ekskursjon til Stockholm fikk vi en innføring i I2i-teknologien, som skulle benyttes, slik at mediatoren skulle kunne produsere øyekontakt i det ikke-fysiske rom. Det viste seg at det eksisterte to forskjellige varianter av teknologien. En hvor kameraet er plassert bak speilet, og en der displayet er plassert bak speilet. De ulike variantene speilet da henholdsvis bildet fra displayet til brukeren og motivet til kameraet.



Vi fikk være med å bygge og prøve begge variantene. På denne måten fikk vi en god og grundig oversikt over styrker og svakheter ved hver av de.



Prinsippet med liggende display og kameraet bak speilet

2.3.2 Resultat

Løsning 1: Kamera bak speilet (display projekseres via speilet)

Svakhet · Bildet fra displayet blir speilvendt til brukeren.

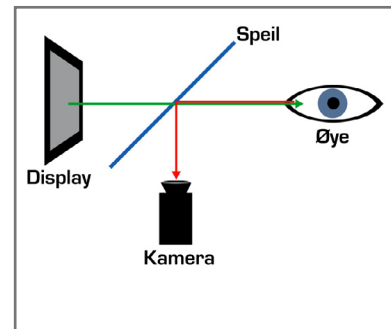
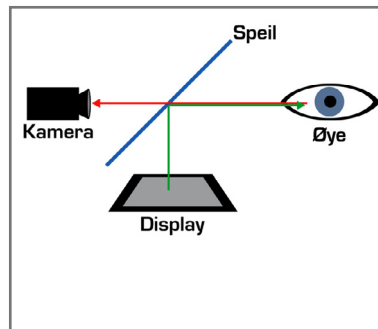
Styrke · Kameraet sender ut et riktig bilde med tanke på speilvendthet.



Prinsippet med display bak speilet, og kameraet henter motivet via speilet.

Løsning 2: Display bak speilet (motivet speiles til kamera)

- Svakhet
- Kameraet sender ut et speilvendt bilde. Om man da ikke speiler to ganger.
 - Bildet til kameraet taper seg noe i speilingen.
 - Man får økt avstand fra bruker til display, i og med at kameraet står imellom.
- Styrke
- Man får et riktig bilde med tanke på speilvendthet på displayet.



Prinsipptegning av de to variantene.

2.3.3 Konklusjon

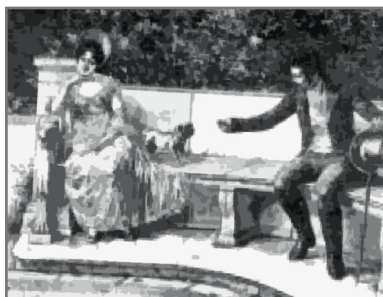
Ut i fra disse punktene bestemte vi oss for å gå for løsning nummer 1. Denne hadde færrest svakheter, og svakheten som var, hadde vi muligens en løsning på. Noen LCD og plasmaskjermer kan i menyen programmeres til å vise bildet speilvendt. Vi hadde også en teoretisk løsning ved bruk av CRT-skjerm. Det er derimot værre å speilvende signalet fra kameraet. I så fall må en ha en egen modul, enten elektronisk eller software-basert for å snu bildet før det sendes.



Et litt komplisert bilde. Bildet er tatt av Lars Jonas, som en kan se i bakgrunnen. En kan også merke øyekontakten Andreas oppnår. Prinsippet vi valgte å bruke i vår applikasjon, er av typen Lars Jonas står foran.

2.4 Speilvending

2.4.1 Bakgrunn



På det øverste bildet er det kvinnen som først blir «oppdaget», og får vår oppmerksomhet. I den underste bildet er det mannen som havner i fokus. Dette er pga. at i vår kultur er det primære bildeområdet oppe til venstre.

Et speilvendt bilde kan oppfattes ganske forskjellig fra originalen. I vår kultur «leser» vi bildene fra øverste venstre hjørne nedover mot høyre hjørne. For at bildet skal oppfattes riktig, er det derfor viktig at det beholder sin originale form.

Når man bruker den varianten av I2i som vi valgte å basere vår mediator på, blir bildet til brukeren speilvendt ifh. til det som viser på displayet. For

at bildet da skal bli korrekt, må det speiles en gang til. Vi hadde flere alternativer for speilvendingsproblematikken:

1) «Mekanisk» speiling via et speil til. Dette vil gi tap i kvalitet, og man vil få en mer komplisert og større konstruksjon. Et speil av denne størrelsen koster også litt, selv om dette ikke trenger å være et beamsplitter speil.

2) Man kan vha. software speilvende signalet. Dette vil gi en økt responstid, fra signalet kommer til det vises på skjermen. Uten at vi har sjekket det ut, tipper vi dette også vil koste en ikke ubetydelig sum.

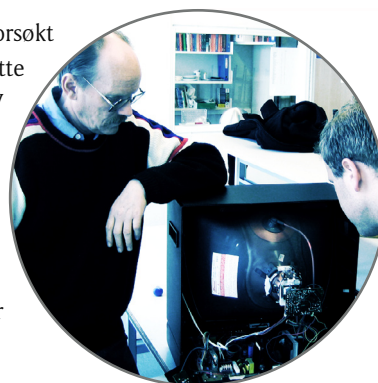
3) En mottar et signal som er speilvendt. Om en kobler sammen to I2i-enheter, en av hver variant – vil den ene levere et speilvendt signal, og den andre vise et speilvendt signal av det den mottar. På denne måten utfyller de hverandre og det vil ikke være behov for ekstra speilvending.

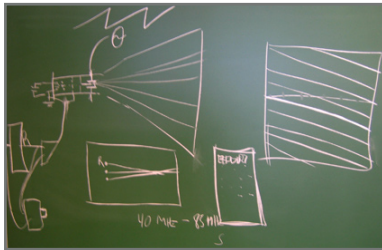
4) Dersom man brukte en god gammel billedrørs-TV til display, er teorien at man har to synk-signaler: Et horisontalt og en vertikalt. Teoretisk skal man da ved å få bytte om retningen på signalet for horisontal synk, kunne speilvende bildet. Vi ville forsøke dette. Dette var den ideelle løsningen, da dette ikke gir noen forsinkelse, er kostnadseffektivt og i teorien skulle en ikke tape bildekvalitet. Det var derfor denne løsningen vi ville gjøre et forsøk med i første omgang.



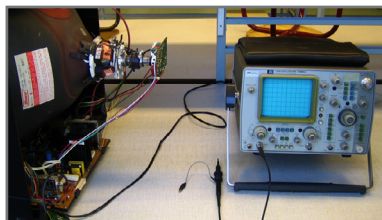
2.4.2 Utførelse

Vi kunne ikke finne noen som hadde forsøkt dette før oss, så vi var usikre på om dette kunne gå. Derfor ville vi få tak i en TV som hadde sett sine beste dager, og egentlig hadde sendt ut sine siste bilder. Vi sendte en fellesmail til alle ansatte og studenter ved HiG, og etterlyste en TV i denne kategorien. Vi fikk bra respons, og allerede dagen etter hadde vi fem TVer som var klar for en siste tjeneste.





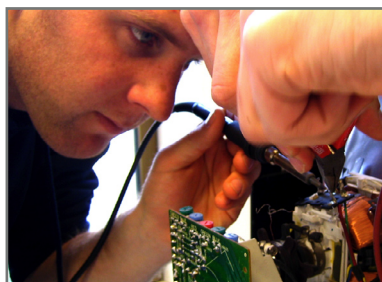
En enkel innføring i elektronkanon-prinsippet, og den teoretiske bakgrunnen for forsøket.



For å prøve å finne hvilke ledninger som førte hvilke signaler, ble det brukt oscilloskop. Dette kan brukes for å kartlegge frekvens, spenning, periodetider og puls/pauseforhold i elektriske signaler.

Etter litt research fant vi ut at TVer hadde spenninger opp mot 16000 volt. Dette er spenninger som verken data- eller grafiske ingeniører bør stikke fingrene i, så vi trengte fagfolk til å hjelpe oss. Vi tok kontakt med Ola Røyrvik, som jobber på elektrolaboratoriet ved HiG. Han stilte velvillig opp for å hjelpe, slik at vi kunne gjennomføre forsøket på en sikker og god måte. For vi fikk gå i gang med eksperimentet fikk vi en generell innføring i hvordan billedrørs-TVer er oppbygd. Vi manglet koblingskjema, og vår fagmann var litt skeptisk til å fortsette eksperimentet. Gikk dette galt, kunne det få konsekvenser. Med dyktig resonering, og kartlegging av signalet vha. et oscilloskop, kom vår mann fram til hvilke fire ledninger som var synk-signalene. Hvilke som var horisontale og hvilke som var vertikale var foreløpig en liten gåte. Vi valgte den vitenskapelige «prøve og feile»-metoden. Vi visste ikke helt hva som kunne skje om vi koblet feil, men i og med at den gule og blå ledningen lå så nært hverandre, forsøkte vi først med en svensk omkobling. Først kom en lyd som vi aldri har hørt fra en TV før – så gikk sikringen. Første forsøk gikk ikke helt etter planen. Etter å ha koblet tilbake til originalt oppsett og skiftet stikk-kontakt, kunne vi konkludere med at TVen ennå var like hel. Vi koblet så sammen de to tykkeste ledningene, og teorien var at de horisontale synk-signalene er raskere enn de vertikale, og trenger derfor tykkere ledninger. Vi satte stikk-kontakten i for andre forsøk...

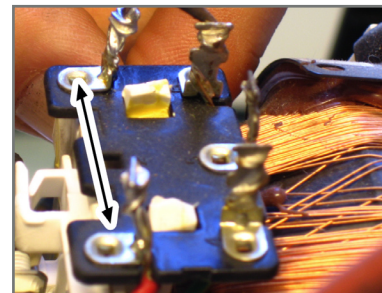
2.4.3 Resultat



Loddingsarbeider



Speilvendt mandagsmatinè. I speilbildet kan en skimte to førnøyde prosjektarbeidere.



Pilen til venstre i bildet viser hvilke ledninger som måtte bytte loddepunkt, for at bildet på TVen skulle bli speilvendt

2.4.4 Konklusjon

På TVer med elektron-kanon, kan man ved å bytte om ledningene for de horisontale synk-signalene, få speilvendt bildet som blir vist på skjermen. Forsøket ble gjennomført med en relativt gammel TV. Det gjenstod å finne ut hvorvidt dette også fungerer på dagens TVer, da disse trolig inneholder mer ømfintlig teknologi.

2.5 Fotografere transmisjon i speiltyperne

2.5.1 Bakgrunn

Pga. økonomien i prosjektet måtte vi spare der det spares kunne. En av de største kostnadene var det spesielle beamsplitterspeilet, der den billigste modellen kostet ca 13000 kroner. Etter hva vi fikk opplyst fungerte dette på samme måte som et tradisjonelt enveisspeil, men skulle visstnok ha et betydelig mindre lystap.

Ved å bruke et vanlig argus-speil kunne vi spare ca 10 000 kroner. Vi gikk derfor grundig til verks for å kartlegge hvor stor og hvilke forskjeller det egentlig var mellom de to typene enveisspeil.



2.5.2 Utførelse



Slik var oppstillingen av utsyrene under forssøkene

Forsøket ble utført på medielaboratoriet ved HiG, både med speilreflekskamera og videokamera. Under forsøket kom det noe falskt lys inn gjennom vinduene i laboratoriet, så resultatene kan ikke sees på som helt eksakte. Vi hadde heller ikke mulighet til å måle styrken på lyset under forsøket.

Kamera og kamerainstillinger under forsøket

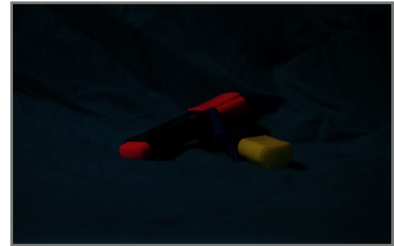
Foto	
Kamera	Nikon D-70
ISO	200
Lukkertid	1/100 sek
Blender	4,5
Fokus:	70 mm

Video	
Kamera	Panasonic dvx-100
Bildeforsterker [gain]	0
Blender	16
Fokus:	44
Zoom	Z61
ND-filter	Av

2.5.3 Resultat



Fotografering uten at lyset må igjennom noe før det treffer linsen.



Fotografering gjennom argusspeilet.



Fotografering gjennom beamsplitterspeil.



Fotografering gjennom argusspeilet med lukkertid på 1/6 sekund.

I tillegg til stillbilder følger en video med opptakene vi gjorde med videokamera [Vedlegg]

2.5.4 Konklusjon



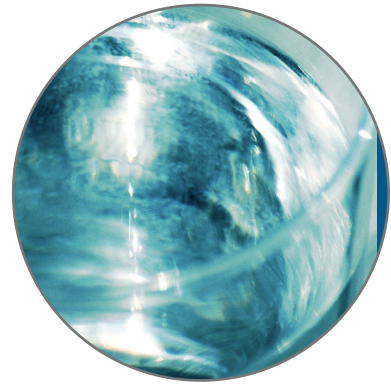
Man kan se sebrastripene på de lyse feltene.

Vi registrerte en ganske stor forskjell på de to speiltypene. For å oppnå tilnærmet lik lysstyrke i bildene måtte vi gå opp minimum 4 hele blendersteg med speilreflekskamera. Når vi gjennomførte forsøket med video klarte vi ikke å oppnå samme lysstyrke i bildet, selv når vi: Stilte inn blenderåpningen slik at vi fikk 80% sebrastriper med blender 16 gjennom beamsplitter-speilet, og gikk til åpen blender og skiftet ut beamsplitter-speilet med argusspeilet. Selv da var det ikke tendenser til sebrastriper. Med disse resultatene måtte vi gjøre et forsøk der vi kunne kartlegge hvilke lysforhold det ville være der mediatoren skulle være plassert. Dersom vi uansett måtte bruke ND-filer ved bruk av beamsplitter-speilet, kunne argusspeilet brukes som en erstatning.

2.6 Fotografering av refleksjon

2.6.1 Bakgrunn

Vi ville se hvor stor forskjell det var mpt. refleksjon eller speiling i de to speiltypene.



2.6.2 Utførelse

Ved å legge speilene på et svart teppe, og sikre at det ikke kom noe lys under speilene, ville vi se hvor god refleksjon hver av speiltypene hadde. Motivet ville bli fotografen selv. Fotografiene er tatt med like instillinger og utstyr som ble brukt ved forsøket «fotografering av transmisjon».

2.6.3 Resultat



Refleksjon fra beamsplitterspeil.



Refleksjon fra argusspeil

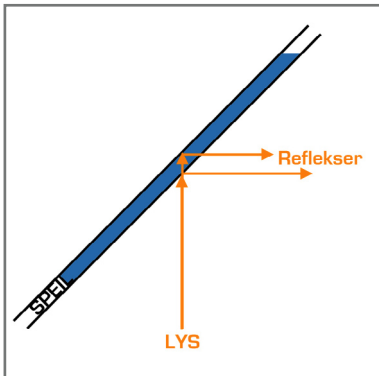
2.6.4 Konklusjon

Fra ekskursjonen til Stockholm visste vi at beamsplitter-speilet hadde gode nok refleksjonsegenskaper. Forsøket ble derfor gjennomført for å fastslå hvorvidt argusspeilet hadde refleksjonsegenskaper som minimum tilsvarte beamsplitterspeilets. Resultatet av forsøket viste at argusspeilet reflekterte bedre enn beamsplitterspeilet, så refleksjonsegenskapene var ingen hindring for å benytte det i vår applikasjon.

2.7 Dobbelspeiling

2.7.1 Bakgrunn

Fra Mats Erixon fikk vi tilsendt to speil til å bruke i applikasjonen. Et stort og et lite. Det store speilet var mye tykkere enn det lille –for etter hva vi kunne skjønne for å kunne bære sin egenvekt. Vi fikk i oppgave å kartlegge forskjellen på disse. Mats mente at det tykke speilet hadde sterkere og mer merkbar dobbeltspeiling, ifh. til det tynne. Teorien bak påstanden er vist i figuren til venstre. Jo tykkere speilet er, desto lengre blir det mellom de to refleksjonene. Hva som ellers er forskjellen mellom de to speilene fikk vi ikke opplysninger om.



2.7.2 Utførelse

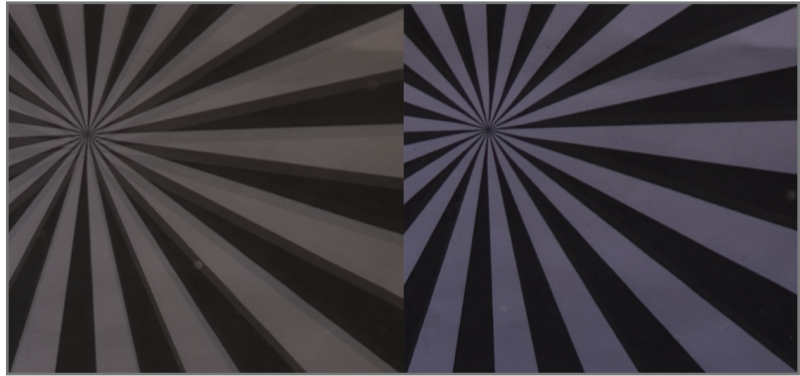
Kamerainnstillinger: Kamera: Nikon D-70, (Raw-format), 0,5 sek ISO: 200

Under forsøkene ble speilene fotografert i 45 graders vinkel ifh. til kamera. På baksiden ble de dekket med et lag med tøy og et lag med presenning. Dette for å sikre at det ikke kom noe lys inn bakfra. Testplansen var et A4 ark med svart-hvitt felter plassert i sirkel. Dette for å vise en eventuelt dobbeltspeiling i ulike vinkler.



Oppstilling av forsøket

2.7.3 Resultat



Det tynne speilet. Et bilde av refleksjonen på hver side



Det tynne speilet. Et bilde av refleksjonen på hver side

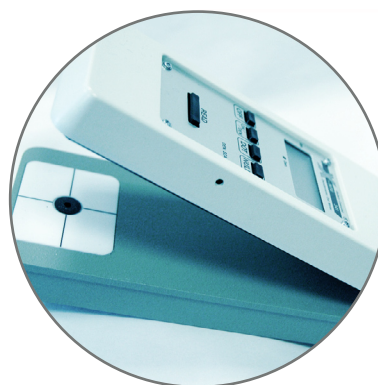
2.7.4 Konklusjon

Første gangen vi gjennomførte forsøket var resultatene så ulike at vi satte spørsmålsteget ved det. Når vi da for andre gang gjennomførte forsøket med pinlig nøyaktighet –kom vi til den samme konklusjonen: Det tynne speilet er, dersom man bruker riktig side, merkbart bedre enn det tykke speilet.

2.8 Måling med transmisjonsdensitometer

2.8.1 Bakgrunn

For å tallfeste forskjellen mellom de to speilene tok vi kontakt med Sven Erik Skarsbo ved Fargelaboratoriet på HiG. Der fikk vi låne et svart hvitt-transmisjonsdensitometer, som kan måle hvor mye lys som slipper igjennom et legeme.



2.8.2 Utførelse

Vi tok 10 målinger på ulike plasser på hver av de to speilene, og fikk ut to sett med verdier for transmisjonen.

2.8.3 Resultat

Argusspeil

Måling	Verdi
1	0,86
2	0,86
3	0,86
4	0,86
5	0,86
6	0,88
7	0,89
8	0,88
9	0,87
10	0,88
Gjennomsnitt	0,87

Beamsplitter

Måling	Verdi
1	0,07
2	0,08
3	0,09
4	0,08
5	0,08
6	0,09
7	0,09
8	0,08
9	0,07
10	0,07
Gjennomsnitt	0,08

For å få resultatene til å fortelle oss noe konkret, tok vi et søk på Internett, og fant en side som omhandlet temaet [ref]. Formelen som ble brukt var:

Prosent transmisjon = $(1/(10^{\text{verdi}})) * 100$. Med utgangspunkt i gjennomsnittsverdiene til hver av speiltypene og nevnte formel, kom vi fram til hvor stor prosentdel av lyset som slipper gjennom i hver av de to typene.

Argusspeil: 13,49 %

Beamsplitterspeil 83,17 %

Differansen blir da 69% mellom de to typene.

2.8.4 Konklusjon

Resultatene bekreftet den store forskjellen vi kunne registrere i forsøket med fotografering gjennom speilene. Det som overrasket oss var at lystapet i argusspeilet var på hele 87%, altså kommer kun 13% av lyset gjennom. Vi ble ved dette forsøket enda mer overbevist om at argusspeilet var en absolutt siste utvei. Resultatene fra fotoforsøkene viste at lystapet i beamsplitterspeilet ikke var mulig å oppfatte med det blotte øye. At målingene i dette forsøket da viste over 10% lystap var tankevekkende. En god forklaring kunne vi ikke finne, men det kan tenkes at lystapet varierer med lysets innfallsvinkel. En forklaring på forskjellen i resultatene, kan man kanskje finne om man går dypere i hvordan beamsplitterspeilet fungerer. Dette valgte vi ikke å gjøre, da vi nå hadde fått gode nok resultater til at vi kunne gjøre et valg.



Sony VC-C4



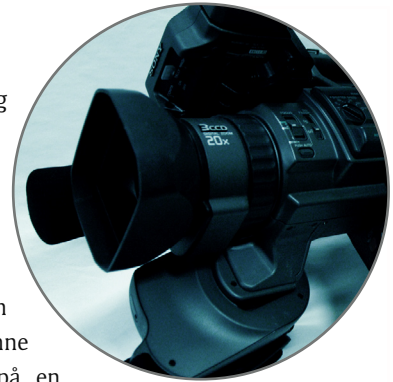
Kameraet som ble dedikert til prosjektet.

2.9 Kamera

Tidlig i prosjektet ville vi ha et bevegelig kamera i konstruksjonen vår. Dvs. at kameraet bak speilet skulle kunne panne, tilte og zoome vha. softwaren. Claus hadde tilgang til en hovedoppgave fra KTH, som i sin helhet tok for seg hvordan man kunne fjernstyre et kamera av typen Sony VC-C4. Kameraet skal man kunne kontrollere vha RS232 (seriellporten) på en server eller lignende, som igjen kan kontrolleres ved hjelp av f.eks. TCP-IP protokoll.

Etterhvert som prosjektet utviklet seg og vi fikk innsikt i hvordan applikasjonen vår ville se ut, ble det klart at et bevegelig kamera bak speilet ville ha et svært begrenset bevegelsesområde. Derfor bestemte vi oss for å bruke et fastmontert kamera, og heller lage en festeordning som kunne justeres slik at man oppnådde øyekontakt. I tillegg til kameraet bak speilet, var vi også inne på tanken å montere et bevegelig kamera i taket, slik at man kunne bruke dette kameraet til å få en oversikt over miljøet rundt mediatoren. For å ikke gape over for mye, slo vi fra oss dette.

Når vi undersøkte hvilke kamera som kunne brukes fastmontert bak speilet, kom Claus med et profesjonelt kamera, av typen Sony DSR 390, som har vært i bruk på HiG, der kassettmekanismen nå var defekt. Dette ble dedikert til prosjektet.



2.10 Kommunikasjonsenheter

2.10.1 Tandberg 800

Selv om Tandberg 800 er en eldre og sammenlignet med de andre videokonferanseenheterne fra Tandberg, en billig utgave – er det en omfattende og kompleks videokonferanseenhet. Den støtter konferanser både over ISDN og IP, med inntil 10 video og 4 lyddeltakere. Dette gjør at partene kan bruke forskjellige kommunikasjonsmidler i samme konferanse – f.eks. mobiltelefoner og web-kameraer i tillegg til videokonferanseenheter.



Den har også et bevegelig kamera som sitter fast på enheten. For å betjene og kontrollere enheten har man flere muligheter; fjernkontroll, telnet, RS232 og et webbasert brukergrensesnitt som er implementert i enheten. Av tilkoblinger har den:



Baksiden med tilkoblingene til Tandberg 800

Kommunikasjon:

3 stk. ISDN
1 stk. Cat 5

Bildesignaler inn og ut:

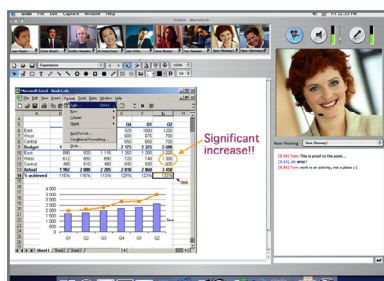
1stk. VGA
1 stk. S-video
2 stk. RCA

Lyd inn (mono):

2 stk. audio XLR
2 stk. RCA

Lyd ut (mono):

3 stk. RCA



Skjermkudd fra Marratech brukt til markedsføring. Legg merke til at de her har brukt bilder der personene søker øyekontakt.

2.10.2 Marratech

Marratech er en softwarebasert videokonferanse-software som er utviklet ved Universitetet i Luleå. Den består av to deler: server og klient. Begge er tilgjengelig til både Windows, PC, Linux, Mac og Solaris. Marratech støtter konferanser både som multicast og som unicast [ORDF], og det er mulig lagre en innspilling av hele konferansen med både lyd, bilde og et eget whiteboard som er implementert i programmet.



Klientsoftwaren er gratis tilgjengelig for nedlasting, slik at man på denne måten kan invitere brukere i konferansen uten at det koster noe ekstra. Med Marratech er det stort sett spesifikasjonene til datamaskinen og hvilke tilkoblinger man har tilgang til, som avgjør kvaliteten og mulighetene i konferansen. Den kommende versjonen av Marratech vil støtte H.323. Dette fikk vi vite da utviklingsmiljøet til Marratech, gjorde en test av dette mot medielaboratoriet på HiG i mai 2005.

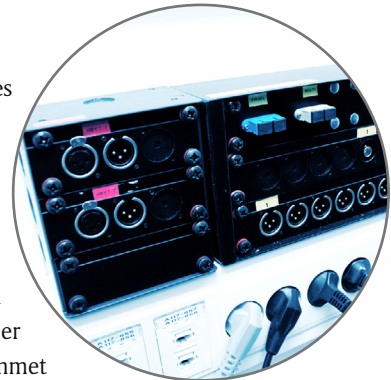
2.10.3 Polyspan streamstation

Prosjektet fikk tildelt en Polyspan Streamstation, som tidligere er brukt ved «Avdeling for Helsefag til fjernundervisning. Dette er en litt eldre enhet hvor man kan sende videostreamer i sanntid og arkivere de slik at de kan lastes ned senere. Enhetene er laget spesielt med tanke på foredrag over Internett. Man kan f.eks. legge ved powerpoint-presentasjoner i motet. Videoavspillingen er basert på Windows Media Player, og enheten blir kontrollert gjennom et enkelt webbasert brukergrensesnitt, og en fjernkontroll til verten for forelesningen. Under uttesting fikk vi ikke denne til å virke sammen med en maskin med Windows XP. Vi valgte derfor å utelate denne fra applikasjonen, men den kan enkelt kobles til som en egen selvstendig enhet ved behov.



2.10.4 U-lab

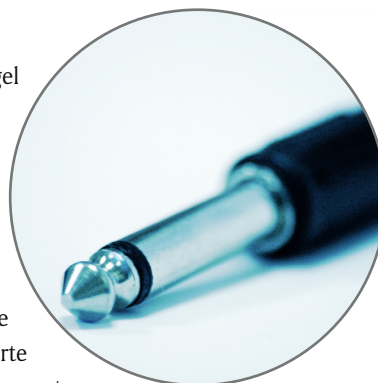
Medielaboratoriene ved HiG har et felles kabelnettverk, for både for lyd, video og datasignaler. I alle rommene i laboratoriet er det tilkoblingspunkter for disse, som går til en sentral patche-enhet. Dette gjør at signalene sendes til det rommet og utgangen man ønsker. Denne muligheten gjør at f.eks. mediatoren kan sende signaler fra en pågående videokonferanse til rommet hvor man redigerer video, slik at man kan lagre hele konferansen. Ved hvert tilkoblingspunkt er det mulig å sende 6 signaler hver vei til den sentrale patche-enheten.



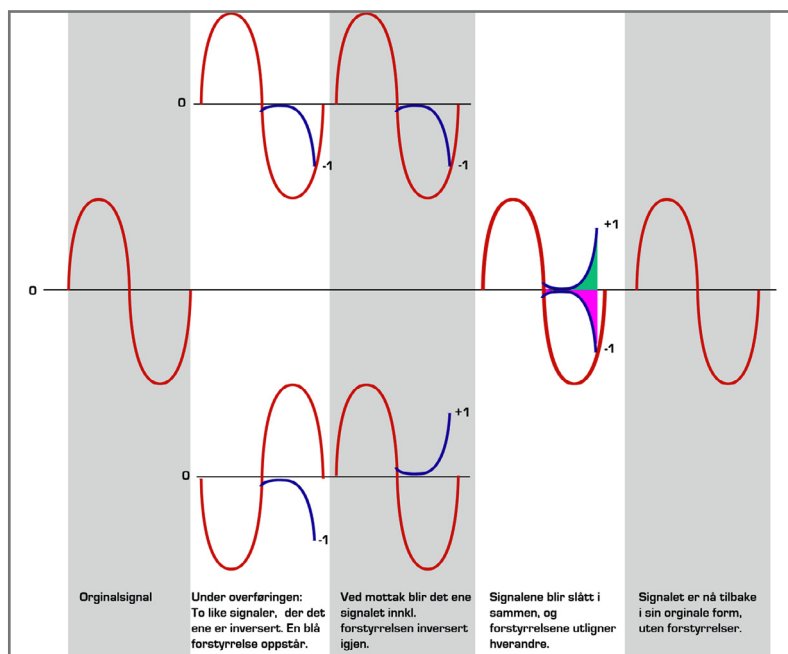
2.11 Analog overføring av video og audio

2.11.1 Audiosignaler

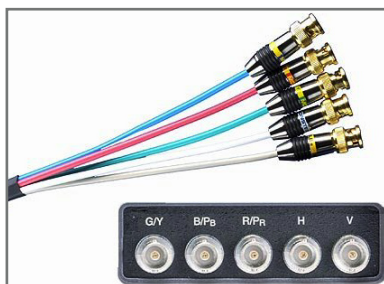
I profesjonelle løsninger blir det som regel brukt balanserte løsninger. Balanserte tilkoblinger eliminerer forstyrrelser fra f.eks. strømledninger. Når man overfører et balansert lydsignal, sender man i praksis to identiske signaler i tillegg til jord. Men de to signalene er inverse. Om man legger disse sammen vil de nulle hverandre ut. I stedet blir det inverserte signalet, for de blir lagt i sammen, inversert på nytt. Dersom det er kommet en forstyrrelse på signalet på veien, vil denne forstyrrelsen være tilnærmet lik på de to signalene. Ved at det ene signalet blir inversert vil da forstyrrelsene nulle ut hverandre.



Ideelt sett skulle vår applikasjon også ha brukt balanserte lydkabler, men balansert utstyr er dyrere og krever flere kabler og tilkoblinger. Ved bruk av ubalanserte lydsignaler kan man f.eks sende stereosignaler gjennom en enkel XLR-kabel, der man ved balanserte løsninger må bruke to.



Et forsøk på en enkel forklaring for hvordan balansert overføring av lydsignaler foregår.



RGBHV



Scart



Komponent



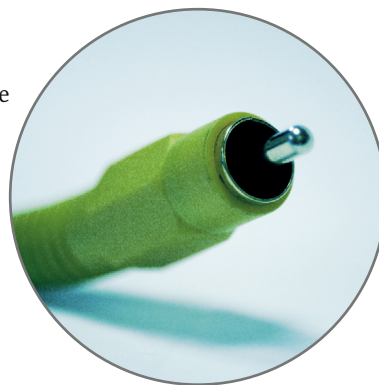
S-video



Composite

2.11.2 Videosignaler

Som man har balanserte og ubalanserte lydsignaler, er det også i videoverden forskjellige måter på å sende signaler på. Som med lydsignaler er hovedparameterene pris, kvalitet og hvor mange kabler som må brukes for å sende videosignalet.



RGBHV

I bunn og grunn består alle videosignaler av de forskjellige komponentene i dette signalet: Rødt, Grønt, Blått, Horisontal-synk og Vertikal-synk. Dette gir et veldig stort signal, noe som gjør det veldig dyrt og komplisert i bruk, men gir den beste kvaliteten. Men kvaliteten blir så mye bedre enn hva som er mulig å vise på. f.eks en TV. Det blir derfor svært sjeldent brukt. Dersom dette signalet benyttes, er tilkoblingen som regel fem BNC-plugger.

RGB-signal

Denne løsningen sender fargene i 3 separate signaler, der begge synkroniserings-signalerne blir baket inn i det grønne signalet. Den mest kjente tilkoblingen er den karakteristiske scart-kontakten, som også brukes til andre typer signaler

Komponent (digitalt: YCbCr /analogt: YPrPb eller Y B-Y R-Y)

En enkel forklaring på dette var litt vanskelig å finne, uten å måtte lime inn tekst vi ikke selv forstod. Men i grove trekk kunne vi forstå at man sender de tre fargene i separate signaler. I tillegg baker man inn luminans(svart/hvitt)-signaler på to av fargene (rod og blå) vha en matematisk formel, slik at mottager kan skille ut de fem signalene. Det er dette signal på denne formen som blir lagret på DVD-plater.

y/c (s-video)

Denne overføringen benytter to signaler. Et sender informasjonen om lys/mørke i bildet, altså et svart-hvitt bilde. Det andre signalet sender fargene i bildet. Når signalet kommer frem til f.eks. en TV blir fargene lagt på svart-hvitt signalet, og man får et video-bilde med bedre kvalitet enn ved senere nevnt composite-signal. Med kvalitet mener vi bildeskarpheit og oppløsning, i tillegg til at man unngår fenomen som f.eks. moiré. I forhold til composite-signalet, vil dette signalet ha en oppløsning på ca 400 linjer.

Å overføre et y/c-signal krever at man sender over to kabler, eller en S-videokabel.

Composite

Dette er den enkleste og derfor billigste måten å sende videosignaler på. Det mest kjente bildet på dette signalet er den gule RCA-pluggen de fleste har et forhold til. Composite-signalet er et signal der bildeinformasjonen er samlet i ett signal. Denne overføringen er mye brukt i til videovisninger i hjemmet, men er også overføringen som gir dårligst bildekvalitet. Med composite-signal anslår man i beste fall en oppløsning på ca 330 vertikale linjer.

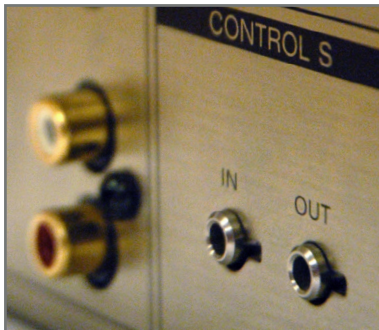
2.12 Matrise

2.12.1 Sony SB-V3000

Om man tenker softwaren som hjernen i applikasjonen vår, er hjertet koblingsmatrisen. Funksjonen til matrisen er å sørge for at riktige signaler kommer fram til de rette enhetene i applikasjonen. F.eks. skal den sørge for at signalet fra mikrofonen blir sendt til Tandberg 800 og/eller til PCen som kjører Marratech.



Sony SB-V3000



Tilkoblingene for Control S er standard 3,5 mm. mono TRS-jackplugger

Applikasjonen vår skulle kontrolleres

vha. software – for at dette skulle gå måtte også

matrisen kunne kontrolleres enten direkte eller indirekte fra PCen.

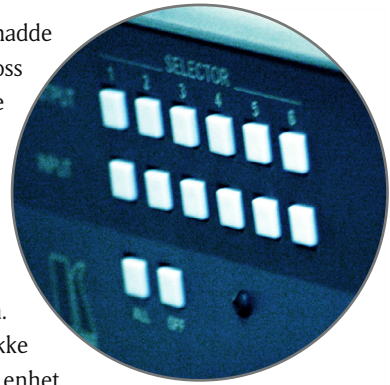
Ved prosjektstart hadde vi fått tildelt en matrise: Sony SB-V3000. Med denne fulgte det verken brukermanual eller andre papirer. Etter informasjonen vi fikk, skulle matrisen ha en RS232-tilkobling, slik at den vha. denne kunne kontrolleres av en PC. Når vi hentet matrisen kunne vi konstatere at dette ikke stemte – den hadde ingen RS232-port.

Pga. de uklare økonomiske rammene i prosjektet ville vi ikke gi opp å bruke Sony-matrisen med en gang. Vi søkte etter informasjon om denne på Internett, og fant etterhvert bruksanvisningen. Her viste det seg at en mulighet for å kontrollere matrisen var vha. control-S. Control-S er en protokoll som er utviklet av Sony - og blir brukt for å kontrollere audio og videoproduktene deres. Slik vi tolket informasjonen vi fant på Internett er dette en protokoll basert på elektropulser eller infrarøde signaler. Mye av informasjonen vi fant på Internett var informasjon som vi ikke hadde kompetanse til å bearbeide, men ut i fra setningen «I know of few successes, in part because it seems S-Link is poorly documented and understood», forstod vi at dette ikke var en altfor vanlig protokoll. Vi måtte innse at vår kompetanse ikke var tilstrekkelig for å avgjøre hvorvidt det var riktig å bruke matrisen videre.

Etter å ha trålet fagmiljøet på HiG for personer som skulle ha kompetanse på dette feltet, fikk vi istand et møte med Ole Johnny Berg. Ut ifra informasjonen vi hadde, forklarte han at det kunne være mulig å få en datamaskin til å kontrollere matrisen, men at dette ville kreve både mye kompetanse, arbeid, tid og endel penger. I og med at det var lite eller ingen dokumentasjon på denne protokollen, måtte man måle og kartlegge elektriske signaler i tillegg til at man muligens måtte bygge enheter for å gjøre datamaskinens bytes om til de riktige elektriske pulsene. Vi fikk heller ingen garanti for at vi ville finne en løsning – og det var heller ikke sikkert en løsning ville være noe særlig god.

2.12.2 Bestemmelsen

På bakgrunn av informasjonen vi nå hadde – bestemte vi oss for ikke å benytte oss av Sony-matrisen. Vi begynte å tråle markedet etter matriser som passet til vårt bruk. Den norske produsenten Network Electronics ble alternativet vi jobbet videre med. De produserer både audio og video-matriser av ulike størrelser. Disse kan kontrolleres både vha. telnet og RS232. Networks produserte ikke multimediamatriser, som i en og samme enhet kunne rute både audio og video-signaler. Prisen pr. matrise var på ca 9000 ekskl. moms – dette ble 18 000, i og med at vi både skulle ha matrise for audio og video.



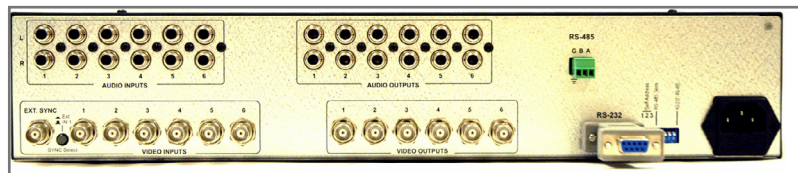
Audiomatrisen fra Network Electronics – her i bruk i applikasjonene vi var med å bygge i Stockholm. Under kan man se deler av de fiberoberoptiske tilkoblingene.

På mediatorene vi bygde i Stockholm, brukte de bl.a. matriser fra Networks - og disse fungerte godt, og tillegg til det vi kunne lese i dokumentasjonen, var de relativt greie å kontrollere fra datamaskiner.

Vi sjekket likevel litt videre og fra loggen vår gikk det fram det at vi tidligere hadde vært interesserte i en matrise fra Kramer Electronics. Vi tok opp igjen denne tråden og det viste seg at ved å bruke denne, som var en multimediamatrise, slapp vi med halve kostnaden ifh. til Networks. Selv om audioinngangene på disse var RCA-tilkoblinger mot balanserte XLR-innganger på Networksmatrisen, var økonomien i prosjektet såpass presset at vi ikke kunne forsvare et kjøp hos våre landsmenn.



Matrisen vi valgte å bruke i applikasjonen. En Kramer VS-606XL multimediamatrise.



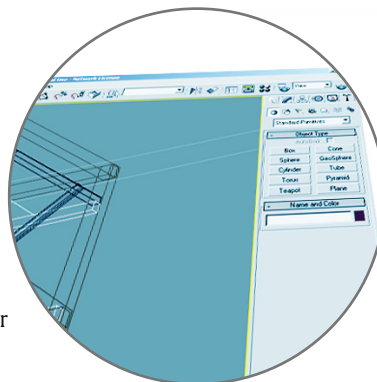
Baksiden av Kramer matrisen.

2.13 Planlegging av konstruksjon

2.13.1 Bakgrunn

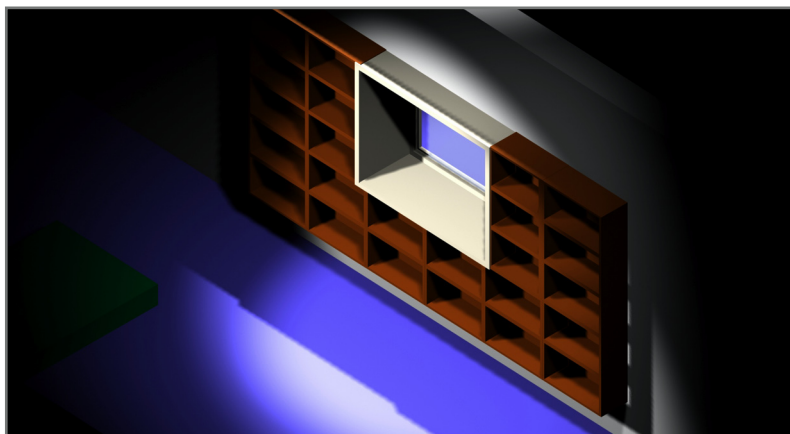
For at applikasjonen skulle dekke flest mulig av behovene ved HiG, måtte vi kartlegge hvilke krav og spesifikasjoner den måtte ha:

- Kunne benytte forskjellige speilstørrelser i konstruksjonen.
- Må kunne flyttes, derfor:
 - Ikke for komplisert
 - Lett vekt
 - Må kunne skrus fra hverandre til transportable moduler
- Enkel å vedlikeholde, og enkelt å skaffe deler til.
- Kamera måtte kunne justeres i høyderetning og sidelengs. Man tenkte seg også muligheten for senere å kunne gjøre dette maskinelt.
- Tåle aktiviteten rundt. Applikasjonen er plassert i medielaboratoriet, der det kan være stor og hektisk aktivitet. Konstruksjonen må derfor være konstruert med tanke på dette..
- Man må kunne flytte den rundt i rommet på f.eks. på hjul.
- Kunne brukes i andre settinger enn i vinduet i multimedielaboratoriet.



2.13.2 Utførelse

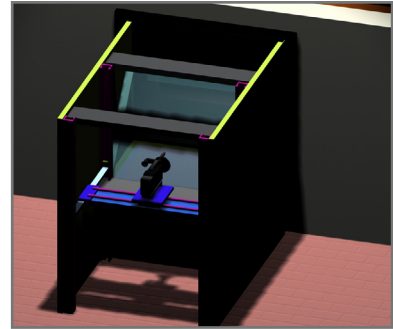
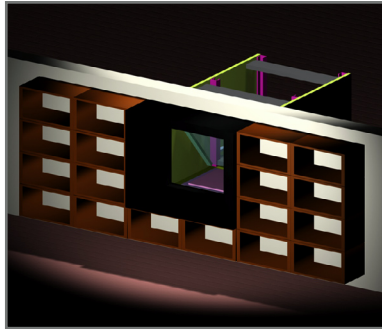
For at konstruksjonen skulle passe inn, måtte vi også gjøre de nødvendige oppmålinger. Oppmålingene ble grunnlaget for modellen vi laget av omgivelsene i 3D-studio MAX. Ved å igjenskape miljøet på denne måten kunne vi prøve forskjellige løsninger for konstruksjonen på en effektiv måte – både med tanke på tid, penger og kompetanse. Faren er derimot at man kan konstruere ting som er svært vanskelig, for ikke å si umulig å gjennomføre i den virkelige verden.



Et bilde fra 3D-modellen vi laget av omgivelsene til mediatoren. De grønne firkantene er for å illudere bordene i gangen

Skisse 1. mars

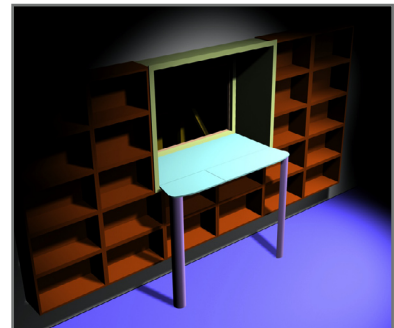
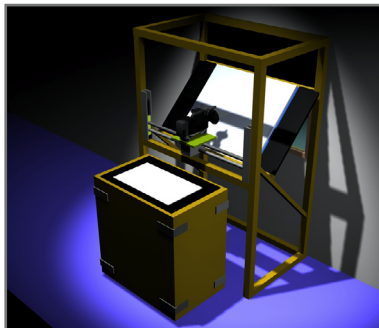
Denne var en rask skisse som ble laget med utgangspunkt i det vi hadde sett i Sverige. Konstruksjonen er ganske stor, og hele miljøet er laget for oppmålingene av omgivelsene ble gjort.



Den første skissen av konstruksjonen. Den ble tegnet for «frihånd», uten oppmålinger av hverken speil rommet eller utstyret.

Førsteutkast 15.mars

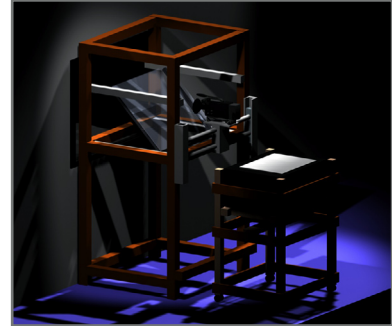
Denne ble laget for vi visste hvor store speilene som skulle benyttes var. På dette tidspunkt var det også den billige argusspeil-løsningen som var foretrukket. Men vi hadde fått til en løsning, der man kunne ta TVen ut som en egen enhet, og renhold og lignende. ble tenkt. på. Vi hadde også tenkt ut en mer plassbesparende løsning for kamerafestet. Men kameraet ble stående ut fra konstruksjonen, og var sårbart for bevegelser rundt. For å få «rommet» mellom kamera og speil mørkt, ville vi kle konstruksjonen med molton.



Et førsteutkast der vi hadde gjort oppmålinger av skapene, vinduet, speilet, TVen vi på dette tidspunkt hadde tenkt å bruke og plasseringen til vinduet på veggen.

Revidert utkast 2.april

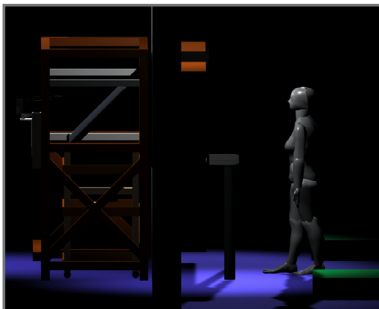
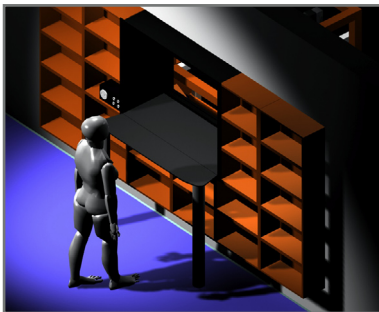
Her hadde vi satt inn et speil i riktig størrelse. Også ergonomiske problemstillinger som hvor høytalere og tilkoblinger for eksterne enheter, som PC, minidisc og lignende, kunne plasseres, ble tatt med i modellen. Generelt begynte vi å nærme oss en endelig løsning på dette utkastet.



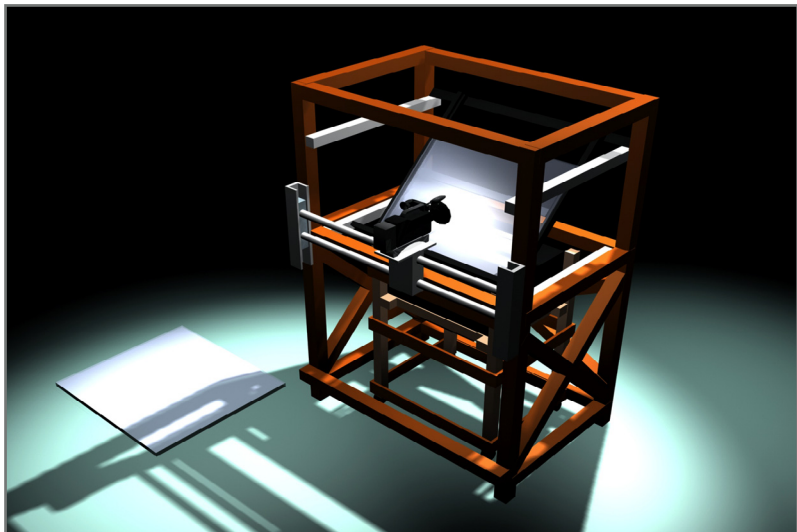
Konsulenthjelp

Vi fikk Odd Magne Nørjordet, som jobber som byggingeniør hos Palm og Bratlie AS, til å se på utkastet av 2. april, og komme med kommentarer. Vi er tross alt ikke byggingeniører, og hadde relativt liten erfaring med konstruksjoner. Vi fikk kommentert at konstruksjonen manglet «kryss», som er avstivere i konstruksjonen – disse forhindrer at konstruksjonen klapper sammen som et korthus. Ved større konstruksjoner må man ha dette. I vårt tilfelle kunne man bruke stålhjørner, men det beste hadde vært å få kryss i konstruksjonen.

Vi hadde til nå ikke tenkt så mye på hvilke materialer vi skulle bruke. De aktuelle materialene var aluminiumsprofiler og tre. Når vi diskuterte saken med vår byggingeniør kom vi til at det enkleste og billigste alternativet var å bygge i tre, selv om det peneste og mest praktiske med tanke på transport og vedlikehold ville være å bygge i aluminiumsprofiler.



2.10.3 Resultat/Konklusjon



2.14 Miljøet rundt mediatoren

2.14.1 Bakgrunn

For at applikasjonen skulle gli best mulig inn i miljøet, måtte vi kartlegge hvilke spesielle hensyn vi måtte ta i utformingen av den.

Applikasjonen skulle plasseres i rom A216, som blir brukt bl.a. til øvinger i lyssetting og flerkameraproduksjon.

Vi måtte derfor konstruere mhp. at konstruksjonen skulle kunne tåle roff behandling, og

helst også kunne fjernes og settes på plass igjen på en enkel måte. Stor aktivitet vil også føre til at det blir mye støv i rommet, så en må derfor lett komme til for å fjerne støv som legger seg på speiler og skjermer. Vi må også designe slik at minst mulig at støvet kommer inn i selve mediatoren, da det ikke er ideelt å vaske speil og skjermer for ofte.

Brukere av mediatoren vil møte den fra gangen. Her var det en utfordring å få til god lyssetting, da spesielt på dagtid. Veggene bak mediatoren har store vinduer, som gir stor kontrast mellom bakgrunn og ansiktet får skygge. Løsningene på dette problemet er enten å skjerme for lyset som slipper inn viduet, eller lyssette personen som bruker mediatoren.

I gangen kan brukerne se mediatoren gjennom et vindu inn til A216. Dette viduet vil også speile omgivelsene i gangen inn på bildet fra mediatoren. Veggene er en brannvegg, så vinduet har vært en stor investering, det er derfor uaktuelt å fjerne vinduet. Denne veggene kunne også by på en utfordring når det skulle hentes signaler fra gangen, via U-lab, til matrisen og til klienten tenkt plassert ved I2i-enheten. Brannveggen er tykk, og man kan ikke ha hull som en eventuell brann vil kunne spre seg igjennom.

I gangen er det også stoler og bord som blir brukt av studentene til arbeidsplasser. Dersom mediatoren blir flittig brukt, kan den være et forstyrrende element for de som jobber her – på en annen side kan disse utgjøre en fin kulisse. De arbeidende kan, om de vil, høre og se kommunikasjonen i mediatoren. Det er da også begrenset hvor privat samtale kan føre.



2.15 Kommunikasjon

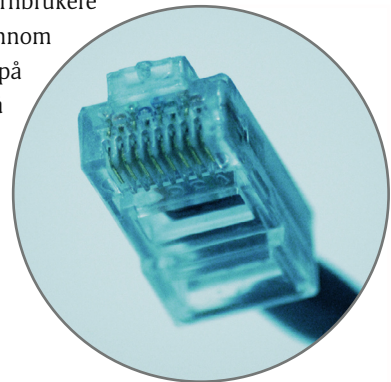
2.15.1 RS232

RS 232 er et grensesnitt for å koble eksterne enheter til en datamaskin, standardisert av EIA(Electronic Industries Association). Maksimumsavstanden er ca 15 meter. Protokollen er kjent for å være robust, og er da brukt når det er mindre, men viktige data som skal overføres.
google: "define:rs232"



2.15.2 Telnet

Telnet er en protokoll, som gjør at fjernbrukere kan få tilgang til en datamaskin gjennom TCP/IP, som er protokollen som brukes på Internett. For å få tilgang via telnet, må man oppgi brukernavn og passord. Når man har fått tilgang til datamaskinen, kan man operere maskinen som om man hadde lokal tilgang.



2.16 Videokonferanser

2.16.1 Programmer og selvstående enheter

For å gjennomføre en videokonferanse, trenger man en enhet som tar seg av det tekniske.

Dvs. digitalisere video og lyd fra deg og sende det til partene i konferansen.

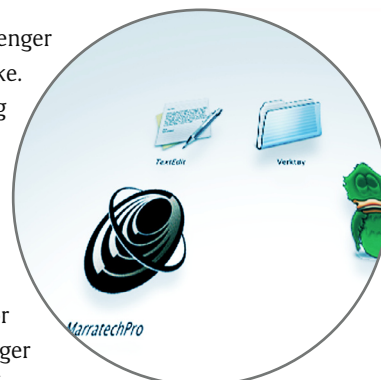
Samtidig som den skal sørge for å decode og vise både innkommende lyd og bilde. Tandberg produserer slike selvstående enheter, men etterhvert har

det kommet andre løsninger på dette. For eksempel har Microsofts MSN-messenger implementert en mulighet til videokonferanse mellom to personer. I tillegg ble vi, av Jonny Widen

introdusert for flere dataprogrammer som kan gjøre dette. Av disse har vi ennå

ikke sett noen som leverer så god bildekvalitet som Tandberg 800-enheten vi benyttet –men dataprogrammene er vesentlig billigere enn Tandberg-løsningene. Et dataprogram for multiparts videokonferanser koster ca. 2000 kr, mens en «rimelig» Tandberg koster over 100 000. En må i tillegg til programmet også ha en datamaskin med gode nok spesifikasjoner.

Et dataprogram for multiparts videokonferanser koster ca. 2000 kr, mens en «rimelig» Tandberg koster over 100 000. En må i tillegg til programmet også ha en datamaskin med gode nok spesifikasjoner.



2.16.2 Protokoller og algoritmer

Når vi skulle forstå forskjellene mellom dataprogrammer og selvstående enheter,

forstod vi at hvilke protokoller og algoritmer som ble brukt, var avgjørende for

kvaliteten på video og lyd. Gjennom søk på Internett fikk vi etterhvert en aldri

så liten oversikt over jungelen. Det ble pratet om en del G'er og H'er, og

etterhvert kom vi frem til konklusjonen:

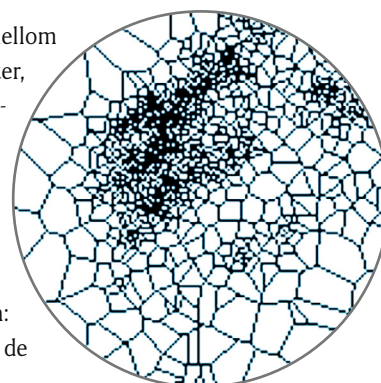
De som begynner med G er for lyd og de som begynner med H er for video.

· H.261 er en eldre komprimerings-algoritme som er blitt gjort overflødig av H.263.

· H.263 er en komprimerings-algoritme som er arvtageren til H.261.

· H.264 er den nyeste komprimerings-algoritmen for videokonferanse-bruk, og for en gitt båndbredde vil H.264 gi ca. dobbelt så god bildekvalitet som H.263. Denne har også støtte for overføring av stereo lyd.

· H.320 (pakke) er en ITU-anbefalt standard med en del individuelle anbefalinger, for koding, komprimering av signaler og opprettelse av samband (H.221, H.230, H.321, H.242 og H.261). Den inkluderer de tre audio-algortimene G.711, G.722 og



G.728 Kan brukes i punkt-til-punkt samband og i multipunkt-samband.

- H.323 er en standard for lyd, bilde og datakommunikasjon over IP-baserte nettverk. Ved å benytte denne standarden trenger ikke sluttbrukeren å tenke på kompatibilitet mellom ulike utstørsleverandører i designet av et videokonferansenett. Den kan brukes i punkt-til-punkt samband og i multipunkt samband.

- H.324 er en ITU-standard som tillater punkt-til-punkt oppkobling av data, video og audio-konferanser over analoge telefonlinjer (POTS). Den kan inkludere H.261, men de fleste implementasjonene vil sannsynligvis bruke H.263 – en skalerbar versjon av H.231.

- H.320 er som H.324, men for digitale telefonlinjer (ISDN).

- G.711 er en komprimerings-algoritme for audio. 100 kb per sekund hver vei og gir en meget god lyd kvalitet. Dette er algoritmen som i størst grad blir brukt innenfor den stadig mer aktuelle IP-telefonien.

- G.723.1 er komprimerings-algoritme for audio, som krever rundt 25 kb, og med samtale-kvalitet som tilsvarer mobiltelefoni

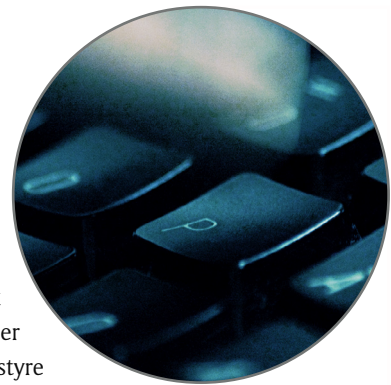
2.17 Programmering

2.17.1 Bakgrunn

Et mål i prosjektet var å gjøre mediatoren så portabel som mulig, og vi måtte finne ut hvilke utviklingsverktøy/kodespråk som kunne støtte best mulig opp under dette. Programvaren skulle bl.a. kunne styre Tandberg-codecen over Telnet (eventuelt RS-232)

og Kramer-matrisen gjennom RS-232. Slik som medielaboratoriene er bygd opp ved HiG, fant vi ut at det var best å plassere både matrise og codec i patch/serverskapet (A212G) og koble enhetene videre til U-lab og Internett. Ut i fra dette var det lett å ta avgjørelsen om at programvaren måtte bygges rundt en klient/server modell, Dette fordi U-lab systemet ikke kan patche RS-232, og matrisen ikke kan styres over TCP/IP.

Vi vurderte flere ulike programmeringsspråk for utvikling av klienten, blant annet var vi inne på tanken å bruke HTML serverside scripting, som PHP eller ASP. Vi hadde tidligere brukt begge disse i andre private prosjekter. Dette ville vært en grei løsning, siden begge disse er svært utbredte og godt dokumenterte. HTML/PHP likevel ikke idéelle teknologier for å drive den type GUI basert software som var vårt formål. For å oppdatere informasjonen i GUIen må man i mange tilfeller laste alle elementene på siden på nytt, noe som resulterer i unødig venting og forstyrrelser i skjermbildet. Fra en ergonomisk synsvinkel er det viktig at brukeren får rask respons på handlinger han gjør mot programmet – i og med at vårt program har en uerfaren målgruppe, ble dette desto viktigere.



2.17.2 Resultat/konklusjon

Java har derimot egenskaper som gjør at brukeren kan få rask respons. Java kan programmeres som flere forskjellige typer applikasjoner; F.eks Java WebStart, applets og gjennom Java J2EE er det også mulig å lage løsninger på server-siden. For at de ikke skal kunne lage for mye «krøll» på klientenes maskiner har Java applets svært begrensede tilgangsrettigheter. F.eks har de kun tillatelse til å koble seg opp mot serveren de selv ble lastet fra. Med bakgrunn i dette vurderte vi lenge å lage klienten i Java WebStart, som opererer med større frihet.

Likevel valgte vi å utvikle programmet som en applet, da denne kan kjøres direkte i nettleseren. Dette forutsetter at en har java VM installert – noe de fleste av dagens maskiner har. At vi valgte å lage en applet på klient-siden gjør at applikasjonen i teorien kan kjøres på alle maskiner koblet til Internett. På denne måten kan mediatoren også kontrolleres fra alle disse.

Kommunikasjonen mellom klienten og resten av mediatoren går via en server. Serveren tar seg av all kommunikasjon mot de ulike enhetene, og sender tilbakemeldinger til Java appleten.

Siden en applet ikke har lov til å skrive/opprette filer på klientmaskinen, må alle innstillinger ligge i en database. Det var ønskelig å benytte en database som støtter SQL standarden, og samtidig en gratis løsning. MySQL er en utbredt, velkjent variant, som samtidig er enkel å sette opp og administrere med løsninger som PhpMyAdmin. Dermed var valget ganske enkelt på dette området.

2.18 GUI

For vi begynte å designe brukergrensesnittet til softwaren som skal kontrollere applikasjonen, kartla vi GUIen hos andre applikasjoner som hadde lignende målgrupper og formål som vår applikasjon. Applikasjonen vår skal være tilpasset brukere med liten teknisk erfaring. I tillegg skal de vanligste funksjonene brukes av de litt mer erfarne brukerne, uten at de føler den alt for tungvint i bruk. Vi trekte relasjoner mellom vår applikasjon og minibanker.

Minibanker skal lett kunne brukes av uerfarne brukere, samtidig som det for erfarne brukere ikke skal være for omstendelig. Brukergrensesnittet må derfor være intuitivt, og formspråket skal være imotekommende og tydelig. Vi fikk se at her hadde det vært en utvikling fra tiden da det kun var grønn skrift på sort bakgrunn. Egenskaper vi merket oss ved minibankene, som kunne være aktuelle å bruke vårt brukergrensesnitt, var:

- Svært begrensede muligheter. Minibankene tilbyr ikke alle tjenester som en bank gjør. F.eks. overføring mellom konti og andre avanserte operasjoner.
- Formspråket er preget av avrundede kanter og begrensede farger.
- Lyd når en trykker på knapper.



- På minibanker med touch-screen har knappene animasjoner som simulerer egenskapene til en fysisk knapp.
- Det er forsiktig fargebruk, med mørk bakgrunn og sterke farger for å understreke viktige meldinger, og kritiske valg.

I tillegg til minibankene sammenlignet vi operativsystemene OSX og Windows XP. Apple som har utviklet OSX er kjent for å være svært gode på brukervennlighet både i utvikling av hardware og software. Når vi studerte litt hva som generelt skilte operativsystemene fra hverandre, satt vi igjen med et inntrykk av at OSX har begrenset eller gjemt mulighetene noe, for å skape et enklere uttrykk. OSX har også et rundere og tydeligere formspråk enn Windows. Som et eksempel på dette har man størrelsen på og formen på ikonene.



Microsoft Word ikon

3 Utførelse

3.1 Software

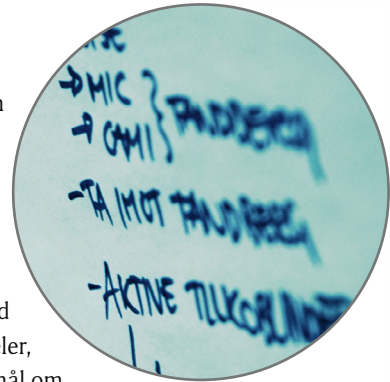
3.1.1 Planlegging

Kravspesifisering

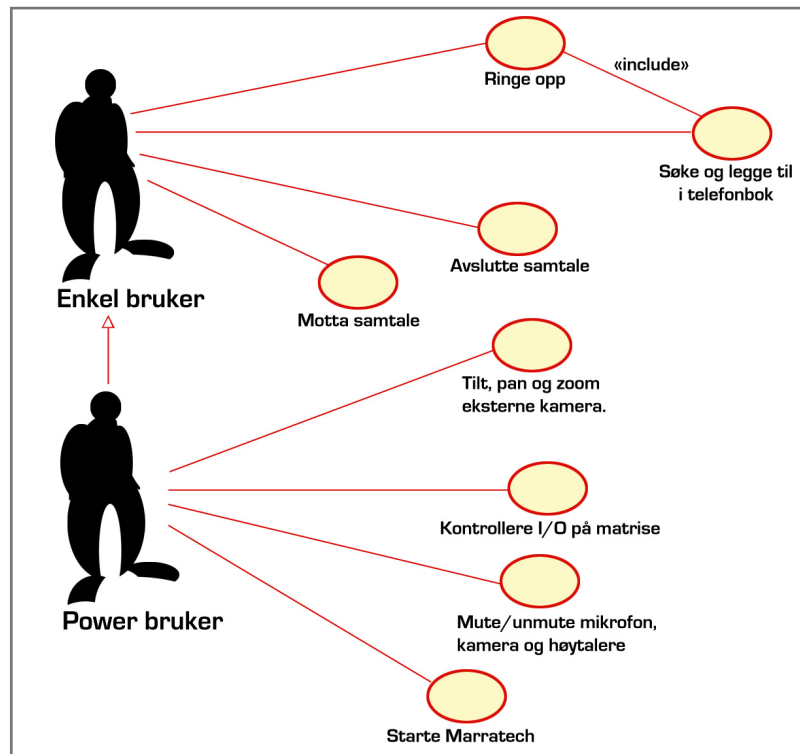
Ved prosjektets start forelå det lite/ingen kravspesifikasjon fra oppdragsgiver på hva mediators brukergrensesnitt og software skulle omfatte. Samtidig var det klart at uten et felles, samlende grensesnitt, ville mediators bli vanskelig å betjene. Det ville bli et kaos med uavhengige fjernkontroller, løse paneler, koblinger osv. Dette ville stride mot vårt mål om forenklede videokonferanser, derfor ble et av målene

at vi skulle utvikle et styrings-program som skulle kunne kjøres på en PC, og inngå som en del av applikasjonen.

Via grensesnittet skal det være mulig å foreta oppringninger gjennom mediators, både gjennom ISDN og TCP/IP, styre hvilke lyd- og bildesignaler som går inn og ut av mediators, samt at det skal kunne være mulig å bruke eksterne programmer som f.eks Marratech i kommunikasjonen. I tillegg skal det være mulig å lagre telefonnumre i en implementert telefonbok. Det vil også være nødvendig med et passordbeskyttet område hvor en kan administrere instillingene til applikasjonen. For å få en oversikt over hvilke funksjoner styringsprogrammet skulle ha, laget vi et usecase-diagram.



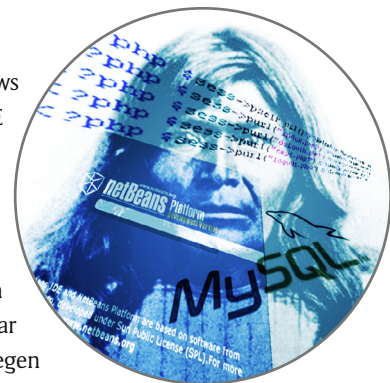
3.1.2 Usecase-diagram



I vårt tilfelle vil det ikke være nødvendig å legge inn support for flere samtidige brukere av applikasjonen, siden grensesnittet i denne omgang bare vil bli brukt for å kontrollere én mediator.

2.1.3 Utviklingsmiljø

Utviklingen har foregått under Windows XP, hvor vi har benyttet NetBeans IDE 4.1 som utviklingsmiljø for Java. Under utviklingen benyttet vi Sun Application Server til testing. All utvikling skjedde på Andreas' bærbare PC, med codec tilkoblet over Internett, og matrisen tilkoblet RS-232 porten. Databasen har under hele utviklingsforløpet kjørt på egen server, den samme server som til slutt skal inneholde hele applikasjonen.



NetBeans IDE

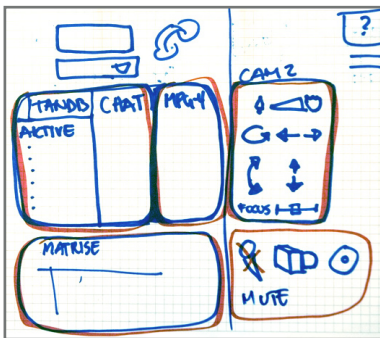
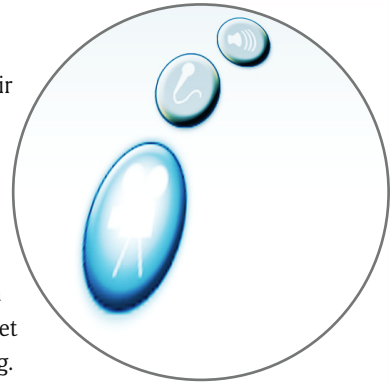
NetBeans 4.1 tilbyr omfattende support for Javas platform for applikasjonsutvikling mot web, J2EE 1.4.2, og gjør det samtidig til en lek å utvikle GUI baserte applikasjoner via J2SE 1.5.

Web Server

Prosjektet fikk tildelt en Windows maskin som skulle brukes til server. Av serverprogramvare hadde vi positive erfaringer med både Apache, MySQL og PHP. PhpDev er en samlepakke som inneholder en ferdig oppsatt programvare med alle disse komponentene innkludert.

3.1.4 GUI til klient

Det grafiske brukergrensesnittet blir programvaren sitt ansikt mot brukeren. Et godt designet GUI hjelper brukeren slik at han til enhver tid gjør de rette tingene. I og med at målgruppen for applikasjonen bl.a. bestod personer med mindre erfaring, og muligens en aldri så liten frykt for tekniske utfordringer, var det avgjørende at GUIen er så intuitiv så mulig.



Mockup'en so ble laget etter vi hver for oss hadde laget et utkast.

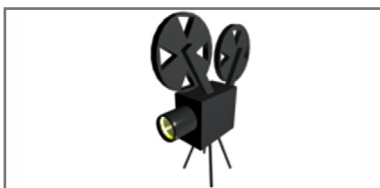
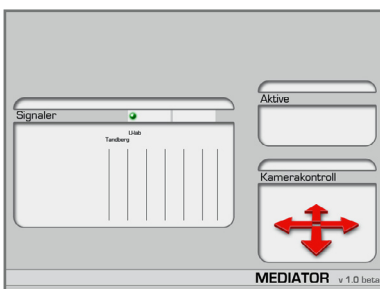
Når vi startet med designet, brukte vi det daværende funksjonskartet som grunnlag. Hver for oss laget vi en blyantskisse på hvordan vi tenkte oss det skulle se ut. Videre samlet vi dette i en svært enkel dummy eller mockup.

Neste steg var å bestemme formspråket til GUIen. Fra forundersøkelsene satt vi igjen med egenskapene:

- Avrundede former
- Oversiktlig
- Tydelig
- Lave kontraster i fargene (f.eks unngå komplimentærkontraster)
- Myk
- Varm

Førsteutkast til GUI

Ut ifra egenskapene over, ble det laget et utkast i Adobe Photoshop over hvordan hovedelementene skulle se ut. Når det gjelder fargevalgene, valgte vi grå som bakgrunnsfarge pga. dens nøytrale egenskaper, og at den gir god men nøytral kontrast til de fleste farger. Rodfargen ble valgt fordi den er en varm farge, som gir god kontrast mot gråfargen. For å få et distansert forhold til det, ventet vi noen dager før vi evaluerte designet på nytt. I evalueringen kom vi frem til at designen var litt stressende. Da begynte vi å se på fargenes psykologiske virkning. Rødt er en farge som øker adrenalinnivået og er forbundet med krig og kamp.



Ikoner

Etter å laget førsteutkastet til hovedelementene, måtte formspråket overføres til ikonene i applikasjonen. Det første settet med ikoner utviklet vi i 3D Studio MAX. Ikonene var tenkt som realistiske fremstillinger av metaforene. Da vi så ikonene for seg selv, måtte vi si oss fornøyde. Da vi derimot satt de inn i layouten så vi at de

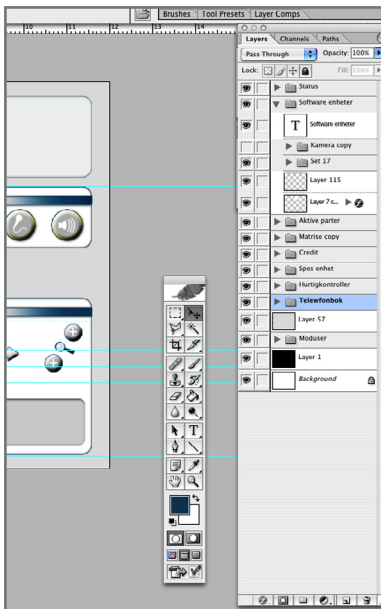
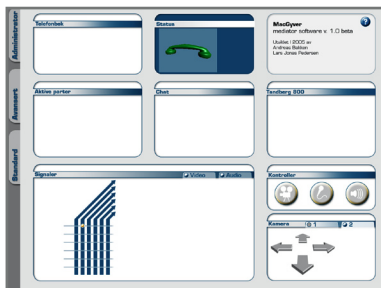
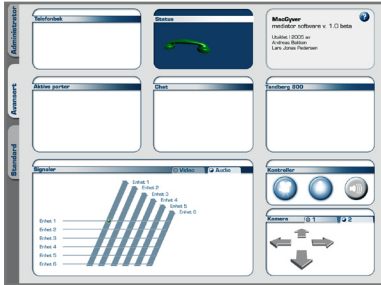
var alt for detaljerte og lite stiliserte. Ikonene gjorde at GUIen kunne minne om en nettside fra tidlig nittital. I det videre arbeidet ble ikonene utviklet sammen med resten av GUIen – for å sikre helhet i designet.

Videre utvikling

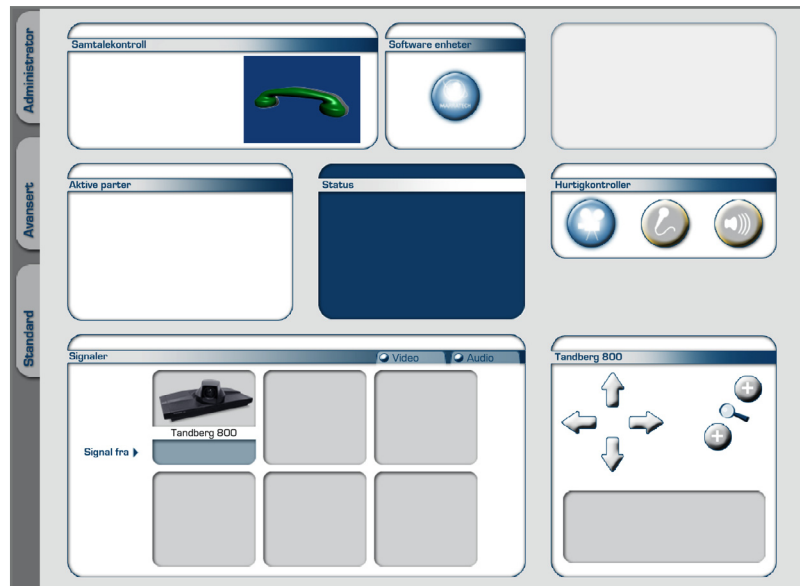
Mellom første og andre ikonsett ble vi også introdusert til programvaren Marratech, som er en softwarebasert videokonferanseløsning. Her hentet vi inspirasjon og ideer til ikoner og løsninger på problemer. Underveis i utviklingen ble det også gjort endringer i funksjonskartet, som igjen førte til endringer i hvilke funksjoner som skulle være tilgjengelige i GUIen. En av disse endringene var at programmet skulle ha forskjellige bruksmodus; standard og avansert. Standard er grensesnittet med svært begrensede muligheter, mens avansert er tiltenkt brukere med litt mer erfaring. I tillegg til disse to modusene, skulle man ha mulighet til å gjøre innstillinger av mediatoren vha et eget passordbeskyttet administrasjonsmodus.

I og med at rødt hadde de motsatte psykologiske virkningene enn de vi ønsket, valgte vi i neste utkast blått som erstatning. Virkningene til blått er at den senker pulsen og får oss til å slappe av. I hele prosjektet hadde vi brukt fonten Eurostile i programmet. Dette er en font som ikke er like mye brukt som fonter som Arial, Helvetica og Times, men som har en karakteristisk stil, samtidig som den er lett å lese. I starten brukte vi den tykkeste utgaven av skriften, men etterhvert fant vi det mer elegant og behagelig med tynnere skrift.

En av de tingene vi brukte mest til på, var å forenkle fremstillingen av matrisen. Vi hengte oss opp i matrisepriippet vi hadde sett i softwaren som fulgte med matrisene, men etterhvert kom vi fram til en løsning som vi mener gir en klarere fremstilling av statusen til matrisen. Løsningen tar litt større plass, men i og med at det er en 6x6 matrise, er det ikke noe problem.



Bilde fra arbeidet med GUIen i Adobe Photoshop.



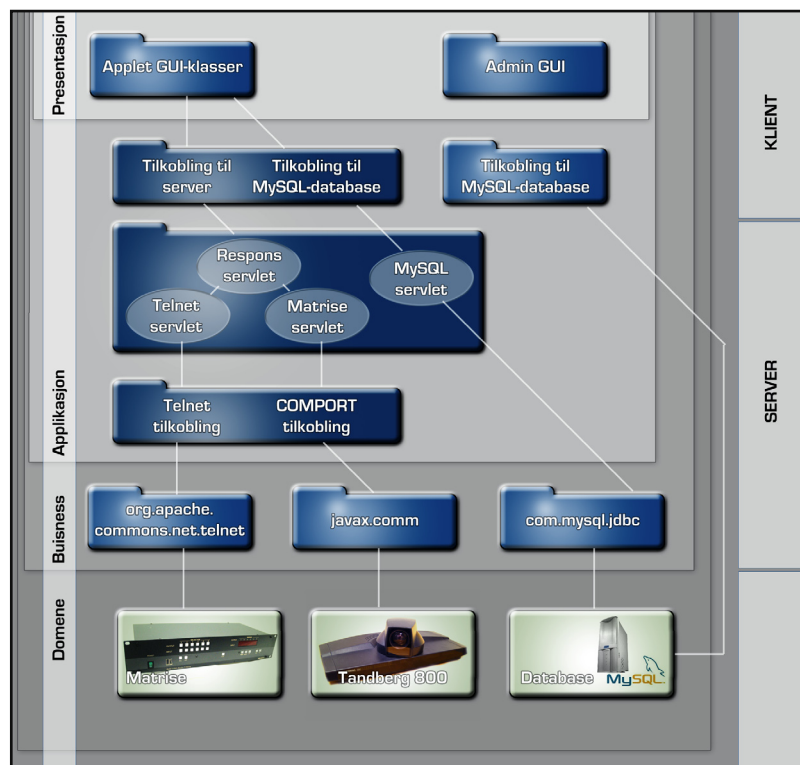
Den endelige GUIen fra Photoshop.

3.1.5 Programmeringsprosessen

Programvaren ble utviklet i flere moduler, bl.a en modul for klient-siden og en for server-siden. De er plassert i hver sine pakker, men er implementert på web-serveren som en enkel web-applikasjon. I tråd med den iterative arbeidsmetoden, ble hver modul utviklet, testet og videreutviklet gjennom hele prosjektets periode. Utover det har vi ikke forholdt oss til noe større rammeverk for prosjekt- og konfigurasjonsstyring. Dette var ikke nødvendig siden gruppen var såpass liten, og at programvareutviklingen inngikk som en del av mange andre oppgaver i prosjektet. Vi har imidlertid hatt god struktur og hyppighet på sikkerhetskopiene for å sikre versjonshåndtering og at verdifull kode ikke gikk tapt.



3.1.6 Designskjema



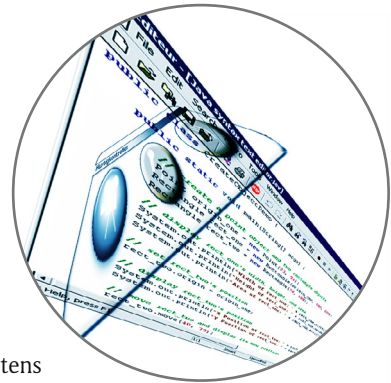
Diagrammet viser gjennom en lagdelt struktur hvordan brukergrensesnittet er bygd opp og implementert. Brukeren blir presentert det øverste laget, hvor vi finner programvarens GUI klasser. Litt lenger ned finner vi applikasjonslaget, som inneholder klasser for kommunikasjon mellom applet og server. Serveren benytter ferdige pakker for kommunikasjon over telnet, seriellport og MySQL. Lengst ned i diagrammet ser vi de fysiske enhetene serveren snakker med.

3.1.7 Klient

Generelt

Klient-appleten kan startes ved å åpne en URL til serverens "front-side" i en nettleser. Klassene til appleten er fordelt over tre JAR arkiv:

- 'mediator_client.jar': Denne inneholder klasser for GUI og kommunikasjon mot serveren.
- 'mediator_images.jar': Inneholder appletens grafikk
- 'mediator_commons.jar': inneholder felles klasser for både server og klient



Klient-appleten vår var avhengig av å kunne kommunisere med serveren for å kunne utføre sine oppgaver mot mediatoren. Appleten benytter flere ulike tråder under eksekveringen.

Ut ifra usecase-diagrammet kan vi se at interaksjonen med brukeren kan skje i to ulike modus som vi valgte å kalle «standard» og «avansert».

Standard modus

Funksjonaliteten er begrenset til kun å inneholde de aller enkleste operasjonene. Dette inkluderer å ringe opp og ta imot samtaler, hente og lagre numre i telefonboken, samt å kunne skru av og på eget kamera, mikrofon og høyttalere. Hvilke innsignaler som matrisen skal benytte kan ikke settes eksplisitt her, men dette kan lett endres i administrator GUI. Hensikten med Standard modus er at selv den mest uerfarne bruker skal kunne klare å sette opp en forbindelse via mediatoren. Når appleten starter, vil den bli startet i dette modus.

Avansert modus

Inneholder en god del mer funksjonalitet, og gjør det attraktivt for selv en erfaren bruker å sette opp mediatoren via brukergrensesnittet. Her kan man i tillegg til Standard modus' funksjonalitet også velge hvilke signaler som skal gå gjennom matrisen og ut på de ulike kommunikasjonsenhetene, styre motpartens kamera (hvis motparten tillater dette), støtte for flerpartskonferanser, samt å kunne benytte Marratech på klientmaskinen.

Administrator

Vi hadde i utgangspunktet tenkt å integre administrator-funksjonaliteten i appleten, men tok etterhvert avgjørelsen om å skille denne fra klienten, både for å begrense kompleksiteten på grensesnittet, og fordi dette er funksjoner som sjelden skal kjøres. Denne delen av programvaren ble utviklet i PHP, da dette gir en enklere løsning for å oppdatere og endre poster i databasen. For å skape en helhet har vi beholdt den samme grafiske profilen som i resten av programmet.

3.1.8 Programmering av administrasjon



Slik ser administrasjonen ut i moduset «innstillinger»



Databasedesign

I og med at applikasjonen vår har en svært begrenset funksjonalitet, krever den heller ikke så mange innstillinger. Administrasjonsdelen er heller ingen stor og omfattende løsning, men vha. denne får en gjort de få nødvendige innstillingene på en enkel og grei måte.

Databasen og innstillinger

Innstillingene man gjør i administrasjonen blir lagret i MySQL-databasen på serveren. Da det var relativt få innstillinger som skulle lagres, bestemte vi oss for å lagre alle innstillingene i en og samme tabell. Merkelig nok fikk denne navnet «innstillinger». Innstillingene ble lagret etter følgende mønster:

- INN_LYD/VIDEO_1-6 I kolonnen «tekst» lagres navnet på enheten som er koblet til den aktuelle inngangen.
- UT_VIDEO/LYD_1-6 I tillegg til navnet på enheten som kan motta signaler bruker klienten her kolonnen «tall» til å lagre hvilke kilde (innsignal for video og lyd 1-6) som til enhver tid er rutet til inngangen.
- PRE_LYD/VIDEO_1-6 I «tall» lagres tallet på inn-enheten som skal rutes til den aktuelle ut-enheten (1-6) ved oppstart av klienten (appleten).
- COMPORT Angir hvilken COMPORT serveren skal bruke i sin kommunikasjon med matrisen. F.eks. «COM2»
- TANDIS_UN Brukernavn til Tandberg-codec'en
- TANDIS_PW Passord til Tandberg-codec'en
- TANDIS_IP IP-adressen til Tandberg-codec'en. Denne blir benyttet i kommunikasjonen serveren gjør med denne over Telnet.

Programmering av administrasjonen

Administrasjonssidene benytter seg av php-sessions for å registrere hvorvidt brukeren er logget inn, og eventuelt hvilke av modusene «innstillinger» eller «innganger», han opererer i. Vi har ikke validert hverken css eller html, men testet den i Safari og MS Internett-explorer.

De sentrale funksjonene i programkoden til administrasjonen er:

OK(), som viser vår «tommel opp»-knapp i stedet for en vanlig html-submit.
input(\$tekst, \$navn, \$verdi), som genererer input elementene i formen. Dersom \$navn, som er det unike navnet på dette elementet, inneholder «pw», blir elementet et passord-element.

prevalg(\$type, \$nr) - henter de forhåndslagrede rutingene. Der \$type er «video» eller «lyd», og \$nr er utgang (1-6)

hentverdi(\$navn, \$tekst) - Returnerer verdien som er lagret i «navn» for \$tekst, som er navnet på kolonnen en vil hente fra. Dvs enten «tekst», eller «tall».

oppdater_innstilling(\$innstilling, \$tekst, \$tall). Skriver endringer til \$innstilling, da enten \$tall eller \$tekst, til kolonnene med samme navn,



Funksjonen «input»

3.1.9 Programmering av server

På serveren kjøres flere ulike servlets, som appleten kan få tilgang til gjennom vanlig URL forespørsler. En servlet håndterer kommunikasjonen mot Tandberg-codec, og sørger for å sende kommandoer og motta respons over Telnet.

En annen servlet kommuniserer gjennom RS-232 porten og sørger for interaksjon med matrisen. I tillegg finnes det en dedikert servlet som sender all respons, fra både codec og matrise, tilbake til klientens applet. Det er derfra opp til klienten å behandle disse videre.

Integrert i server-delen er pakker som hjelper til med protokollene for både seriellport og Telnet. For seriellporten benyttes en extension til Java; *javax.comm*. Telnet blir håndtert av pakken *org.apache.commons.net.telnet*. Kommunikasjon mot databasen skjer gjennom *com.mysql.JDBC*. Alle klasser kjører på Apache Web Server med PHP4.



3.2 Hardware

3.2.1 Bygging av av I2i-enheten

Forberedelser

Med utgangspunkt i tegningene vi hadde laget i 3Dstudio MAX, begynte selve byggingen av I2i-enheten. På tegningene hadde vi ikke tatt for oss detaljene i konstruksjonen, så underveis i byggingen brukte vi mye tid på å tenke ut dimensjonering, og hvordan skulle løse ulike sammenføyninger. For å bygge trenger man verktøy, og dersom vi skulle kjøpt dette til prosjektet hadde det blitt en svært kostbar affære. Vi

tok kontakt med Statsbygg, mer kjent som vaktmesterene ved HiG. De var svært hjelpelige med å låne oss det verktøyet vi ikke hadde selv. Videre måtte vi ha materialer å bygge med. Ingen av oss hadde den store erfaringen med bygging, men etter en generell innføring i begreper som «to-fire» og andre snekkeruttrykk, våget vi oss til Byggmaker Skattum, for å gjøre de nødvendige innkjøp. På den første turen ble det eneste kjøpet en meterstokk – for da vi våget oss inn



i trelasthallen, ble vi overkjørt med begreper og et «snekkerspråk» vi ikke helt forstod. Vi følte at vi ikke fikk roen som skulle til for å gjøre de riktige materialvalgene, og forlot butikken med en alvorlig knekk på snekker-selvtilliten. Etter noen timer med psykisk bearbeidelse av inntrykkene, tok vi turen på ny. Fast bestemt på å virke litt mer snikkervante enn tidligere på dagen. Med meterstokk i sidelomma og ikledd litt mer passende antrekk, ble vi litt mer nøytrale i miljøet. Nå fikk vi roen vi trengte for å finne de riktige innkjøp. Når vi kom tilbake på skolen startet byggingen.

Bygging av skjelettet

Vi startet byggingen av de første modulene – front og baksiden av skjelettet. En av egenskapene konstruksjonen skulle ha; var at den relativt enkelt skulle kunne fjernes fra rommet. Derfor ble den bygget i moduler, som enkeltvis kunne bæres ut av rommet. Konstruksjonen skulle også enkelt kunne flyttes rundt i rommet, og vi bestemte derfor å også sette hjul på hele – ikke bare TVen, slik som på tegningene.

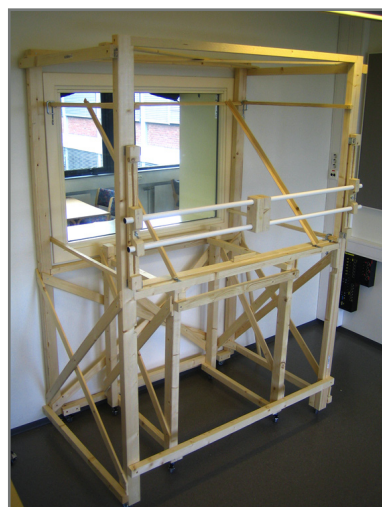
Når vi satte sammen den høye delen av konstruksjonen, vaglet den såpass at vi bestemte oss for å lage en ekstra kryssavstivning i toppen. Resten av byggingen gikk relativt greit, og være unnagjort på ca. fem dager. Utførelsen kunne vært enda mer «forseg-gjort», men om en tar tid, ressurser og ikke minst tidligere erfaring fra trearbeider tatt i betraktning, ble konstruksjonen så absolutt tilfredsstillende.



Underveis



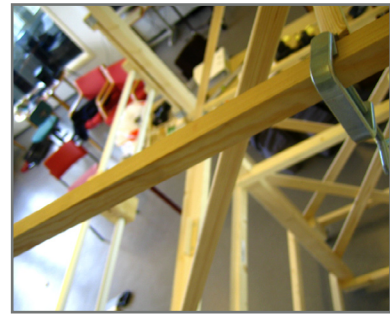
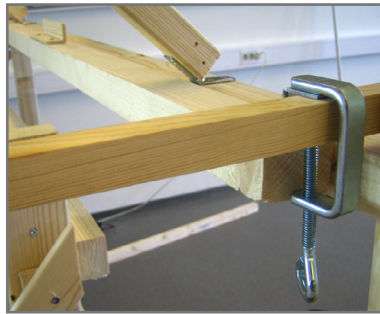
Etter første dag.



Ferdig konstruksjon

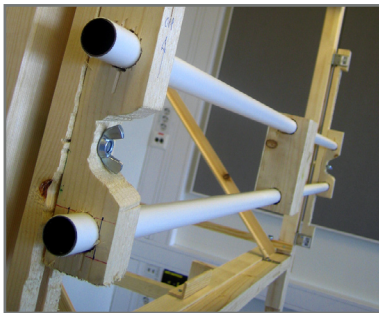
Hvordan holde speilet?

På dette stadiet i prosjektet var det bestemt at vi skulle bruke dyre beamsplitterspeil på utlån fra KTH. Dette gjorde at vi sikret konstruksjonselementene som holder speilene litt ekstra nøye. Slik vi hadde planlagt konstruksjonen var det kun speilenes bredde som kunne være variabel, men fra KTH kom det signaler om at det var ønskelig også å kunne variere vinkelen på speilene. De vinklede speilholderene som hviler på de tverrgående «bjelkene», ble i den ene enden festet med hengsler, og bjelkene festet med skrutvinger, slik at avstanden i mellom disse lett kunne justeres. Dette gjorde at man kunne variere vinklene fra ca 30 grader og oppover. For å hindre at speilholderene skulle ski ut, festet vi den andre enden med stålvinkler.



Hengsler, slik at man kan variere vinkelen.

Skrutvinger slik at man kan variere avstanden mellom bærebjelkene.



Kameraholder

For at det skal oppnåes øyekontakt er det avgjørende at brukeren ser i kameraet. I og med at høyden på brukerne kan variere må kameraets høyde kunne justeres. Det kan også være aktuelt å justere kameraets horisontale posisjon –dette fordi speilets bredde og posisjon i konstruksjonen kan variere. Vi valgte ikke å bruke mye tid på denne delen, da dette er omfattende nok til å være et hovedprosjekt i seg selv. Vi laget en enkel løsning i tre, der kameraet manuelt kan justeres på de nevnte parametere. Presisjon og kvalitet på denne løsningen er så som så, og den må sees på som en provisorisk løsning.



Klar til operasjon

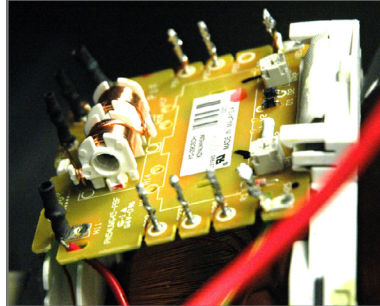
TV

Vi undersøkte hvilke TVer som var tilgjengelige, og endte til slutt opp med å kjøpe en 29" Panasonic hos Elkjøp. Når rekvisisjon var utskrevet fra HiG, gikk kjøpet i boks. Vi var ganske spente på hvorvidt TVen tålte at vi byttet om signalene slik vi hadde gjort med TVen av eldre type. Vi ventet derfor noen dager, før vi en mandagsmorgen startet på arbeidet. Vi var noe mer «husvarme» nå enn forrige gang vi åpnet et fjernsyn. Tross aldersforskjellen var det en overraskende likhet mellom de to TVene. Med de lik farge på kablene så dette ut som dette skulle bli en enkel operasjon.

På forsøket med den gamle TVen brukte vi loddebolten for lenge, med det resultat at festepunktene smeltet løs fra plastikken. I og med at TVen i konstruksjonen skulle ligge og skulle kunne transporteres– kunne vi denne gangen ikke tillate

dette å skje. For å forhindre dette bestemte vi oss for å kutte av ledningene, i stedet for å lodde de av.

Da ledningene var koblet om, var det bare å sette i stikk-kontakten og håpe på det beste. Lyden var fin, og etterhvert som bildet kom fram kunne vi konkludere med at dette gikk som fot i hose.



Nesten helt likt som i TVen i forsøket.



Velykket operasjon



Ferdig TV-holder

TV-holderen

I den delen av konstruksjonen som holder TVen, var det ikke behov for ekstra støtte under TVen, slik det var tegnet inn i 3D-studio. TVen var 45 kg. tung, og når vi hadde bygd konstruksjonen klarte den uten problemer å holde vekten vår –så det var det ikke nødvendig med flere forsterkende elementer.

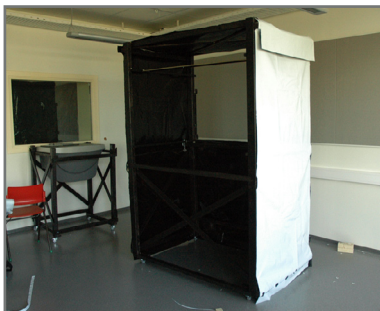
Når vi bygde TV-holderen prøvde vi tenke på hvilke skader det kunne gjøre TVen å ligge, i stedet for å stå. Vår medstudent Rolf A. Storset stilte spørsmål ved hvorvidt luftingen ikke ville bli forhindret når den ligger. Vi studerte TVen vår, og så at det kunne by på problemer. For å være på den sikre siden valgte vi derfor å implementere en vifte i konstruksjonen, for å hjelpe den varme luften bort.



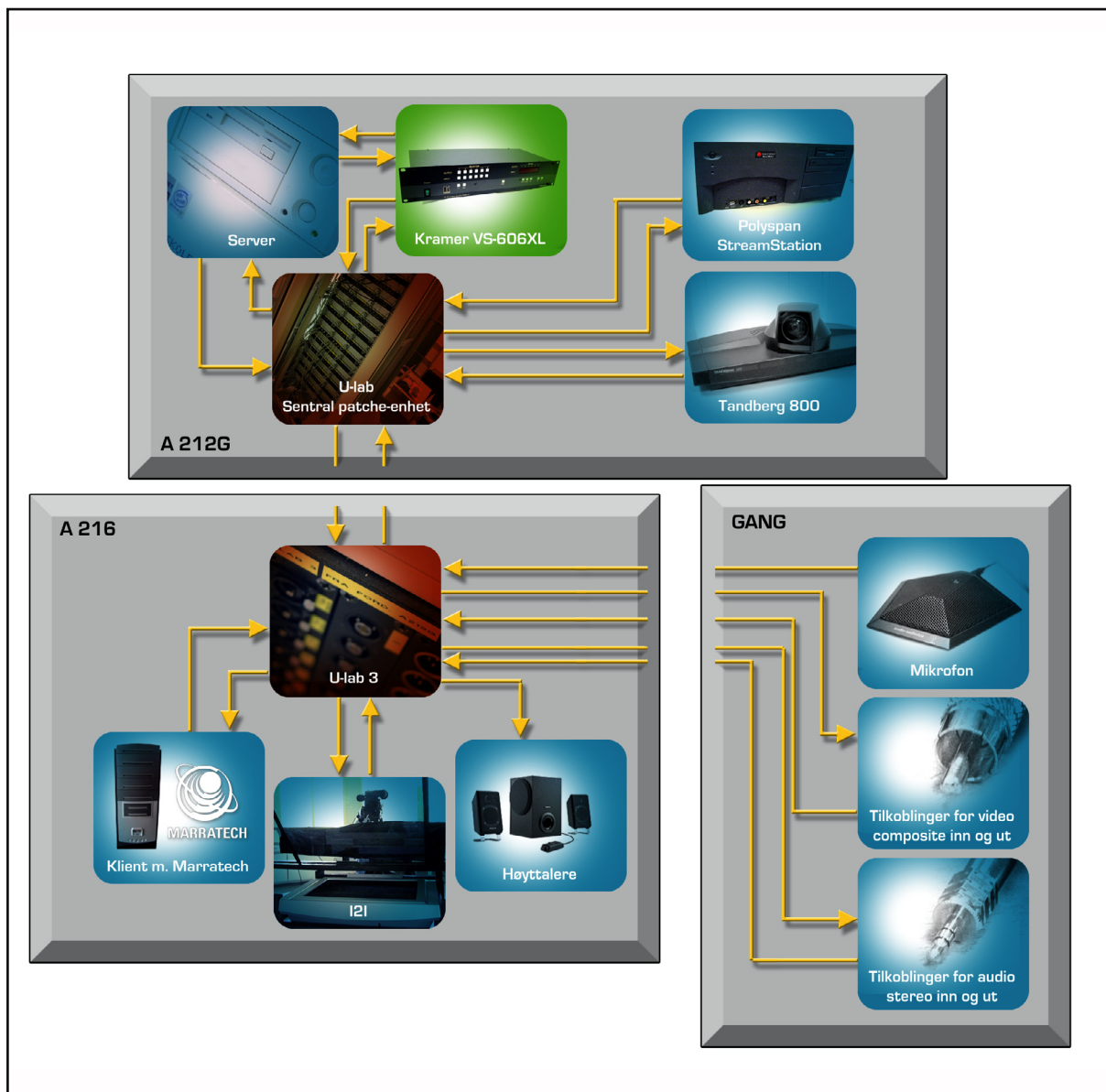
Hjul gjør transporten enkel.

Mørklegge I2i

For at I2i-enheten skulle virke best mulig måtte det være mørkt bak speilet. For å oppnå dette kledde vi hele konstruksjonen med sort presenning. Denne hadde for glatt overflate, slik at en god del lys ble reflektert i steden for å bli absorbert. Dette ble spesielt merkbart i området bak speilet – derfor kledde vi i tillegg dette området med svart tøy, slik at overflaten ble matt.



3.2.2 Deployment-diagram



Fysisk sammenkobling av mediatores komponenter.

3.2.3 Omgivelser



Legg merke til refleksjonen i vinduet, og vinduet bak.

I miljøet rundt mediatoren fant vi i forundersøkelsene flere faktorer som kunne by på utfordringer. Den første utfordringen var vinduene i bakgrunnen som gjør at det kommer sterkt lys inn bakfra. Dette løste vi ved å, vha borrelås, lage «gardiner» av samme type presentasjon som vi kledde I2i-enheten med – dette er ingen estetisk god løsning, men man får det man betaler for, og om ønskelig kan og bør denne bakgrunnen byttes ut med en penere og mer diskret løsning. Borrelåsen gjør at den ved behov lett kan taes ned.

Den andre store utfordringen var vinduet som sitter i vegg mellom I2i-enheten og gangen. Denne var svært tykk og reflekterte svært mye av lyset. Dette gjør at man får se et svakt speilbilde av seg selv i tillegg til bildet fra displayet. Vinduet reflekterer også på innsiden, noe som gjør at også brukeren får feedback på bildet som blir sendt.

3.3 Implementering

3.3.1 Software

Før rapporten skulle leveres fikk vi så vidt begynt med flyttingen av programvare fra testmiljøet til maskinene den skal kjøres på. Det eneste vi rakk var å forsøke var flytte serverprogramvaren over til serveren. Flytting til klient kunne vi ikke starte på, for klienten var ennå ikke levert.

Serveren er av litt eldre type, med 256 MB ram Dette er litt lite, men vi regner med det at vil fungere. I det vi startet flyttingen oppdaget vi problemer med vår egen programkode, og flyttingen ble derfor utsatt til vi fikk eksten kompetanse til å hjelpe oss med problemene.

Klientprogramvaren regner vi med skal gå greit å kjøre på den innkjøpte klienten, da det eneste denne trenger er en nettleser og støtte for java.

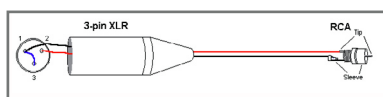
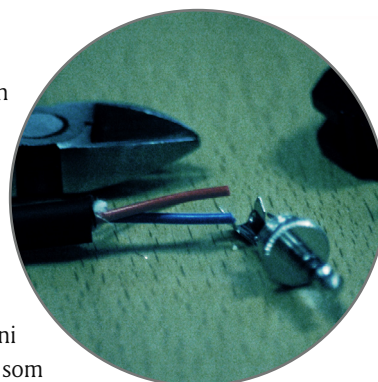
3.3.2 Sammenkobling



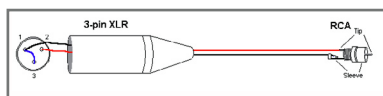
Ulike typer audio-kabler. Fra venstre: 3,5 mm. stereo jack, XLR hunn, XLR hann, og til slutt to RCA(phono).

Råmaterialer

For å koble sammen hele applikasjonen måtte det til mange kabler. Flere av disse kablene var ukonvensjonelle kablinger som f.eks. stereo 3,5 mm. jackplugg, som er pluggen på f.eks. hodetelefoner på bærbare CD-spillere. Denne skal sende stereosignal igjennom XLR-kontaktene på U-laben, for så å ende i to RCA-plugger inni matrisen. Vi tok turen til Fotophono, som er importører for kjente merkevarer som AKG, og leverer mye utstyr til bl.a. NRK. På tross at vi var to gutter som i deres målestokk skulle ha små kvanta, ble vi tatt vennlig imot. De hadde alt vi trengte, og i tillegg fikk vi gratis kursing hos en av deres teknikere i hvordan vi skulle lodde BNC-plugger. På toppen av det hele fikk vi låne nødvendig utstyr for flere tusen kroner. Snakk om service!



XLR til RCA - monosignal



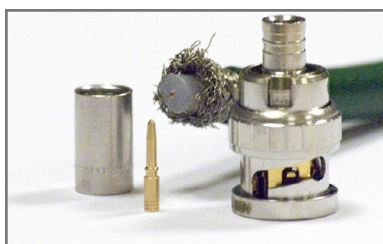
XLR til jack - stereosignal

Lydkabler

Til nå hadde vi bare «råmateriale» til kablene som skulle sy sammen prosjektet. Vi måtte finne ut hvordan de ulike kablene kunne loddes sammen. Ubalanserte sammenkoblinger er svært sjeldent brukt i profesjonelle miljøer – i alle fall er det få som vil vedkjenne seg bruke av det. Det er kanskje derfor det var relativt lite informasjon om dette på Internett, men til slutt fant vi en side som inneholdt informasjonen vi trengte og mer til.. Sammen med en loddebolt og noen kanner kaffe ble lodding av et tredvetalls kabler gjennomført på en natt.

Videokabler

Etter den uformelle kursingen hos Fotophono bød ikke produksjonen av videokabler på større problemer. Den var faktisk mye enklere og tok kortere tid enn audiokablene. I og med at vi brukte composite, var det bare et signal i tillegg til jord som skulle loddest. Kobberkjernen i kablene vi hadde kjøpt var litt mindre enn det som var ideelt for tangen vi hadde lånt. For å løse dette ble vi vist en liten «røverløsning» på kurset. Et svært tynt lag med loddetinn på kjernen før vi festet tippen. For å klemme til både tippen og hylsen til kabelen må man bruke en speiltang, som koster ca. 3000 kroner. Det finnes ingen standard for tykkelsen på pluggene, derfor må man ha en tang som passer akkurat typen plugg en har. Vi var så heldige å få låne en slik tang av Fotophono.



Delene en BNC plugg består av, i tillegg til en coax-kabel, som den skal festes på.



Delene settes så i sammen



til en brukende plugg.

3.4 Økonomi

I og med at vi i applikasjonen benyttet utstyr som var kjøpt inn før prosjektet startet, har vi ikke en oversikt over de totale kostnadene. Kostnadene presentert under er derfor utgiftene som påløp under prosjektets gang. Alle priser er innkl mva.

3.4.1 Rekneskap

Materialer, bygging I2i-konstruksjon	ca	2 500,- kr.
TV	ca	4 000,- kr.
Matrise	ca	9 500,- kr.
Klient-PC	ca	5 500,- kr.
Høytalere	ca	500,- kr.
Plugger	ca	1 500,- kr.
Samlet	ca	23 500,- kr.

3.4.2 Budsjett før byggingen startet (4. april)

DAEWOO DSC3210E, 32" flat widescreen-TV		5 000,- kr.
Kramer VS-606xl Multimedia Matrix	ca.	9 500 kr.
Argusspeil 42"	ca.	4 000,- kr.
Div. Materialer	ca.	3 000,- kr.
Kamera	ca.	2 000,- kr.
Samlet		23500,- kr.

I dette budsjettet hadde vi ikke regnet med en klient, da vi på dette stadiet ville klare oss med en av skolens litte eldre maskiner, men oppdragsgiver ønsket etterhvert i prosjektet å kunne benytte software som Marratech i applikasjonen, og ville derfor dekke denne kostnaden ut over budsjettet.

3.4.3 Tilleggs kostnader

Ved bygging av en slik applikasjon, må man i tillegg regne kostnader på anslagsvis

Beamsplitterspeil	ca	13 000,- kr.
Materialer for å mørklegge I2i	ca	500,- kr.
Merkostnader kabler	ca	1000,- kr.
Mikrofon	ca	500,- kr.
Server	ca	5 000,- kr.

3.5 Mål som ikke ble nådd

3.5.1 Brukskvalitetstester (ergonomi)

For å få en løsning som fungerer godt for brukeren, er det viktig å foreta en grundig evaluering av hvordan applikasjonen kommuniserer med brukeren. Ved å gjøre dette vil en avdekke styrker og svakheter når det gjelder effektivitet, hvor godt brukeren «husker» bruken fra gang til gang, og hvor mye det krever å lære seg applikasjonen. Disse testene kan brukes både ved utvikling av hardware og software, eller som i vårt tilfelle – en kombinasjon av disse. Metoder som kan brukes for å kartlegge dette er bl.a. spørreundersøkelser, feltundersøkelser, strukturerte brukertester.

3.5.2 Miljøet

Som en kan lese planleggingen av konstruksjonen, hadde vi planlagt å tilpasse mediatoren slik at den gled mer inn i miljøet. Vi hadde bl.a. tenkt å få lage et lite bord, slik at brukeren kunne sitte på barkraker under konferansen. Med hensyn til tid og økonomi, ble det avgjort ikke å inkludere dette i prosjektet.

3.5.3 Testing av software

Vi har ikke rukket å teste stabiliteten til softwaren over en lengre periode. Vi tror likevel at applikasjonen vil være relativt stabil, fordi for hver funksjon vi har implementert har vi testet den ganske grundig – og i tråd med den iterative arbeidsprosessen har vi gått tilbake i prosessen og testet og forbedret både med tanke på funksjonalitet og stabilitet.

3.5.4 Programmering

Hoveddelen av funksjonaliteten til softwaren ble ferdig til innlevering av rapporten, likevel var det enkelte funksjoner vi ikke fikk ferdige til rapporten skulle leveres:

- Innkommende samtaler: Vi har ikke fått ferddigtestet hvordan applikasjonen reagerer på innkommende samtaler. Koden er stort sett på plass, men den siste finpussen og testing er ikke gjennomført.
- Sikre stabilitet dersom fler tar applet'en i bruk, og foretar innstillinger. I og med at denne kan kjøre på alle maskiner med Internett-tilgang, er det en viss sansynlighet for at flere kan bruke klienten samtidig.

Vi vil jobbe med noen av disse punktene i uken frem mot presentasjonen.

3.5.5 Språk språkvask og typografi

Vi skulle gjerne brukt mer tid på språket og typografien i rapporten, men vi har forsøkt å finne en fornuftig balansegang mellom språk, typografi og innhold.

3.5.6 Dokumentasjon

I og med at applikasjonen var «oppe å gikk» først to dager før rapporten skulle leveres, har vi ikke rukket en grundig dokumentasjon av svakheter, oppstart- og vedlikeholdsrutiner til applikasjonene. Dette vil også jobbes med frem mot presentasjonen

4 Evaluering/ diskusjon

4.1 Prosjektgjennomføringen

4.1.1 Ressurser

Applikasjonen vi skulle bygge startet, for oss, som en meget vag antydning, og utviklet seg etterhvert til en meget kompleks enhet – med mange brikker som skulle falle på plass.

Hvilke finansielle ressurser prosjektet hadde tilgang til var, etter hva vi kan si i etterkant, et alt for lenge uavklart spørsmål. Dette har vi i prosjektperioden sett på som en svært lærerik utfordring, og er en svært god og viktig erfaring å ta med seg videre; Når rammene til et prosjekt på et felt er uklart –gjør dette at resten av rammeverket også er usikre. Overført til vårt prosjekt, kan man si at som en konsekvens av at de økonomiske rammene var uklare –kunne man heller ikke sette bastante rammer på verken tid eller hva vi skulle oppnå. I en slik situasjon kan i beste fall et slikt rammeverk antydes. Brikkene i økonomien begynte først å falle på plass i april, og de påfølgende innkjøpene ble levert prosjektet 19.april – med andre ord hadde vi heller kort tid på å softwareutvikling for enhetene som oppdragsgiver ikke allerede hadde.

Å være klar over hvilken kompetanse som er og ikke er i prosjektgruppen, samt hvilke ekstern kompetanse man har tilgang til, er også en viktig faktor for å gjennomføre et godt prosjekt. Et eksempel på dette i prosjektet, var at med erfaringen og kompetansen i utvikling av en såpass omfattende, komplisert og «sær» software, som gruppen inneholdt – ble kanskje læringskurven litt bratt. Dette er et hovedprosjekt, og mye av poenget er at man da skal lære av sine feil. Derfor vil vi i fremtidige prosjekter være bevisste i forholdet mellom kompetanse og mål –og at risikoen for å ikke lykkes stiger etterhvert som denne avstanden øker.

4.1.2 Innsatsen

Gjennom hele prosjektet har vi vært motiverte, men i starten var det litt lite «tak» i oss. Vi kunne godt ha utnyttet og arbeidet mer denne tiden. At man jobber best under press, har vært et beviselig faktum for gruppen. Fra en ikke alfor intensiv start i januar – har innsats og tid brukt i prosjektet økt i takt med at deadline nærmet seg. Å kunne jobbe jevnt og trutt mot et fjernt mål er en egenskap som enten er medfødt selvdisiplin, eller må erfares. Vi er nok i den siste gruppen –men gjennom prosjektet har vi fått en viss erfaring med at det er bedre å sikre en «flying start», enn å risikere en flyvende landing. Vi er nok ikke de eneste som har gjort og kommer til å gjøre denne erfaringen. De fleste hovedprosjektene rundt oss, har stort sett jobbet mer eller mindre døgnet rundt uken før innlevering.

4.1.3 Applikasjonen

Klienten, som er hjertet i applikasjonen, var ikke klar for rapporten skulle leveres. Videre har dette da ført til at vi heller ikke har noe grunnlag til å evaluere hvordan applikasjonen fungerer verken mot brukeren eller i «under skallet». At applikasjonen er langt ifra perfekt, er uansett helt sikkert. På mange områder er det fortsatt mye å hente på veien fra det virtuelle mot det fysiske møtet, men vi føler at vi har kommet noen steg videre –og at softwaren, som forhåpentligvis er ferdig til presentasjonen 27. mai, vil være et av de større stegene prosjektet har bidratt med.

4.1.4 Måloppnåelse

I og med at vi ikke har fått testet mediatoren – vet vi ikke helt hvor godt vi har oppnådd målet om å lage en applikasjon som er lettere å bruke enn de som eksisterer idag.

Målet vi har fått jobbet minst med er å tilpasse applikasjonen til miljøet rundt. Her har tiden ikke strukket til. Grunnen til dette dette er gjerne en kombinasjon av litt dårlig planlegging og at prosjektet har gapt over for mye.

Når det gjelder å utvikle en programvare som forenkler og senker terskelen for bruk av videokonferanser, tror vi at vår software vil være et steg i riktig retning.

I2i-enheten vi bygget fungerer tilfredstillende, og vil gjøre at brukerne føler møtet gjennom mediatoren som et «nærmere» møte ifh. til vanlige videokonferanser.

4.1.5 Læring

I prosjektets gang har vi ikke vært bevisste på all læringen vi har fått gjennom prosjektet. Men når vi nå ser tilbake blir vi overrasket over hvor mye vi lært, og vi vil faktisk påstå at vi har lært mye mer enn hva vi forventet ved prosjektstart.

For grafikeren, som fordyper seg i medieteknikk gjelder det konkrete ting som f.eks.

TVens oppbygging, forskjellige typer overføring av lyd og videosignaler, lodding av kabler, bruk av 3D studio Max. Programutvikleren har fått en solid kompetanse innenfor softwareutvikling i java, på godt og vondt. Han har også fått erfaring i og opplevd utfordringene med knytte softwaren sammen med hardware som kommuniserer Telnet, RS-232.

I tillegg til disse håndfaste punktene, har vi tilegnet oss minst like mye «svevende» kompetanse, som er vår betegnelse på kompetanse en ikke klarer å klart uttrykke med ord. Dette kan være innsikten i virtuelle møter, og hvordan vi reagerer på et slikt møte ifh. til et fysisk møte. En viktig læring, som vi vet vil få mye bruk for videre i karrieren – er erfaringene vi har gjort gjennomføringen av et såpass stort og komplekst prosjekt som prosjektet etterhvert ble. På forskjellige områder i prosjektet ble det brukt mye tid på planlegging, mens vi på andre områder tok ting mer «på sparket». I ettertid ser vi at gode forundersøkelser og nøyaktig planlegging gir resultater. Både i form av jevn og sikker progresjon, gode resultater. Et eksempel kan være ved innkjøp: Bruker man litt ekstra tid til å kartlegge det virkelige behovet og tråle markedet for å finne den beste og billigste løsningen – kan man på sikt spare både tid og penger i. I en stor og tung organisasjon som det HiG er, er det også mange formaliteter som må være i orden før innkjøp kan gjøres. Vi er fra dagliglivet vant til å kunne kjøpe ting når man trenger det, men formalitetene tar tid. Det er derfor viktig å ligge litt forut for sin tid, slik at ting kommer på plass for en trenger de –ellers går det med mye tid og frustrasjon til venting.

5 Konklusjon

5.1 Gruppen

I prosjektet har gruppen arbeidet svært godt sammen – kanskje for godt? Det har ikke vært noen form for krangler, og alle diskusjoner har stort sett vært saklige, og konstruktive. Når vi har mott motgang har vi vært flike til å dra lasset sammen, og ha en positiv instilling. Dette har kanskje vært viktig for prosjektet, men det er mulig at prosjektet hadde hatt et annet resultat dersom det ikke hadde vært så god stemning i gruppen. Det blir sagt at det er i oppoverbakke det går oppover, og at det er energi der det er spenninger. Ikke vet vi, men vi tror ikke det.

For det har vært et svært godt og trygt arbeidsmiljø med rom for individuelle behov og særtrekk hos oss begge. Ved prosjektstart bestemte vi oss for å ha en flat organisasjon, der hver og en av oss selv tok ansvar for sin del av kaken. Dette ansvaret har gruppen vært seg bevisst, og begge har kjempet til «the bitter end».

En skulle kanskje tro at man etter en så hektisk periode skulle være temmelig lei av hverandre, men prosjektet har vært med på å utvikle et vennskap som vil vare mye lengre en til skoleporten. Prosjektet har tatt oss med på svært mange opplevelser i begge endene av skalaen; Fra en stor maktesløshet og galgenhumor i det softwaren uken før innlevering bestemte seg for ikke å virke, selv når vi hentet frem gamle versjoner – til alle de positive opplevelsene fra eksursjonen til Stockholm, og gledene ved gjennombruddene vi på veien.

5.2 Virtuelle rom

Etter å ha gjennomført dette prosjektet har vi fått en innsikt i muligheter og problemområder rundt virtuelle møter i ikke-fysiske rom. Den mest alment kjente formen for dette er den tradisjonelle videokonferansen som via datamaskinen for alvor har begynt sitt inntog i vanlige hjem. Forskjellen mellom disse løsningene og større, dyrere løsninger er relativt stor – både når det gjelder kvalitet på overføringen og muligheter.

Vi tror at i takt med utviklingen på området, vil det i overskuelig framtid bli mer

aktuelt også for små- og mellomstore bedrifter, samt det for offentlige instanser å benytte videokonferanser som en erstatning for noen av de mindre møtene.

Videokonferanser har til nå stort sett begrenset seg til visninger på TVer og andre konvensjonelle enheter, men det foregår utvikling og forskning på alternative løsninger, som vi f.eks. «Det kommunikative rom», som er en del av doktorgradsavhandlingen Knudsen fullførte i 2004.

I2i-teknologien, som vi har benyttet i vårt prosjekt er en annen løsning, som er et steg i retning av å gjøre den teknologiske veggen mellom partene i det virtuelle rom transparent. Som et eksempel på at videokonferanser vil innta vår hverdag har vi f.eks prosjektet «Folkets Hub», som Mats Erixon har jobbet med. I dette prosjektet forsøker man å via fiberkabler å tilby folk i med avsidesliggende strøk litt av det samme kulturtilbudet som i byen.

6 Fremtid/visjon

I det vi nå skriver på siste kapittel i oppgaven, sitter vi med noen blandede følelser. Vi er selvsagt glade for at vi snart har fullført oppgaven, og dermed også ingeniørutdannelsen vår. Vi er også fornøyde med at prosjektet har et resultat som vi kan si oss fornøyde med og er stolte av. Vi har til en viss grad klart å mestre alle utfordringene vi har møtt på veien, og setter strek for et intensivt halvår med mange opp- og nedturer.

Likevel har avslutningen en viss bismak. Vi har ennå ikke fått fullført det som skulle være prikken over i'en –softwaren som skulle gjøre den store forskjellen mellom en komplisert enhet, og en oversiktlig enhet som de fleste skulle kunne benytte seg av. Men vi vil gjøre alt som står i vår makt for å få denne klar til presentasjonen om litt over en uke. Likevel vil softwaren, som resten av applikasjonen, ha svakheter, og etterhvert som den blir brukt – vil man helt sikkert avdekke flere svakheter.

Vi ønsker derfor at en ny generasjon «mediatorjenter» eller «mediatorgutter» vil ta opp tråden etter oss. Det er ennå store utfordringer, da spesielt på ergonomisiden, på veien mot en allmenn virtuell møteplass. Hvis vi skal tro noe om hvilke peronligheter som vil ta «jobben», tror vi det er personer med rikelig ballast fra arbeid med video og lyd –og som har en aldri så liten «Reodor» i seg. De må gå til oppgaven med åpent sinn, godt humør, være praktisk anlagt, og villig til å brette opp ermene for å ta et tak utenom datamaskiner og papirarbeid – alt dette i god Mats Erixon-ånd.

7 Kilder

Kap 1.6

http://www2.tisip.no/boker/java/forelesninger/html/Kap1_files/v3_slide0017.htm

Kap 2.3.1

http://www.3dmaxer.dk/forum/post.asp?method=ReplyQuote&REPLY_ID=92411&TOPIC_ID=11371&FORUM_ID=3

Kap 2.1.4

<http://www.dinside.no/php/art.php?id=112719>

Kap 2.11.2

<http://www.hjemmekino.no/guide/tvsignaler.asp>
<http://videoexpert.home.att.net/artic1/227yc.htm>

Kap2.12

<http://www.brian-patti.com/s-link/>

Kap 2.12.2

<http://www.brian-patti.com/s-link/>
http://www.interfacebus.com/Design_Connector_Control_S.html

Kap 2.2

«Vis meg ditt kroppsspråk- og jeg skal si deg hva du oppnår», Lindseth, Live Bressendorf. ISBN: 82-91238-07-3 (h.)

Kap2.15.2

google: define:telnet
<http://www.uninett.no/uninytt/1996-2/tcp.html>

2.16

<http://www.cs.uit.no/~aa/dist/doc/uninett/rapport/>
<http://www.uninett.no/multimedia/videokonf/standarder.html>
<http://www.ip24.no/dok.php?id=94>

Kap 3.1.3

<http://www.firepages.com.au/>

Kap 3.1.4

<http://www.bobyggogbolig.no/Articles/Misc/Article.asp?File=signaler.htm&UnderID=16>

Kap 3.3.2

<http://www.mediacollege.com/audio/>

Kap 5.2

http://www.amt.kth.se/projekt/folkets_hub/index.html

<http://www.gt.kth.se/research/kthlearninglab/krum/sidor/krum.html>

Ordliste

<http://www.hist.no/aft/sebra/kurs/ordforklaringer.htm>

<http://www2.filmweb.no/filmogkino/kinoteknikk/article.jhtml?articleID=34360>

Støttelitteratur

TV optics, Canon

Presence production, Claus Jørgen Schibsted Knudsen. 2004, Doktor avh.

Videomediert kommunikasjon, Claus Jørgen Schibsted Knudsen. 2003

8 Ordliste

Argus-speil

En type en-veis speil, som selges hos vanlige glassmestre.

Balansert linje

Metode for å overføre signaler ved hjelp av to ledere (+ og -) og en skjerm (jord), for å motvirke påvirkning fra eksterne støykilder.

Beamsplitter-speil

En dyrere, mer avansert en-veis speil som slipper gjennom mer lys enn et argus speil.

Codec

En sofistikert digital signalprosesserings-enhet som kan ta et analogt signal som input, konvertere det til et digitalt utgående signal og så komprimere signalet for det sendes til den andre parten. Ved mottakspunktet vil en tilsvarende codec reversere det innkommende digitale signalet til en analog presentasjon. Selve ordet er en sammentrekning av de engelske ordene code/decode (eller som noen eventuelt vil si: compressor/decompressor.) En codec's primære oppgave vil derfor være å få softwaren til å komprimere video- og audiosignalet. En codec består vanligvis av både hardware og software-komponenter. Vanligvis vil det etter en tid være behov for å oppgradere softwaren i codec'en.

Composite Video

Standard videosignal fra TV, videokamera eller DVD-spiller. Composite video gir vesentlig dårligere bildekvalitet enn S-VHS og RGB, ettersom composite video sender all bildeinformasjon som ett enkelt signal.

H.323 og andre lignende komprimeringer

Se kap 2.16.2

ITU-TS

(International Telecommunication Union -- Telecommunication Sector) har anbefalt en serie med standarder for å komprimere lyd

JAR Arkiv

Javas eget format for filpakking. Tilsvarende arkiv kan være cab eller zip.

Komprimering

Dette er en teknikk som reduserer representasjonen av informasjonen, dvs. kvaliteten, men ikke informasjonen i seg selv. Vanligvis består dette i å redusere antall bit som behøves for å kode et signal ved fjerne lange strenger av identiske

påfølgende bit som ikke endres i et samplingsintervall. Komprimering sparer overføringstid eller kapasitet ved at det kan benyttes nettverk som har mindre båndbredde. I tillegg sparer det plass om vi ønsker å lagre signalet på eksterne servere. Denne teknikken kan også benyttes til å redusere størrelsen på datafiler. Her vil komprimeringen være et kompromiss mellom filstørrelse og kvalitet. Høy komprimeringsgrad gir dårlig bildekvalitet og omvendt.

Molton

Tykt stoff som slipper gjennom minimalt med lys.

Multicast

Sending fra et punkt til mange (alle) punkter, hvor alle lytter på samme multicast-adresse. Multicasting er en ressursbesparende måte å kringkaste på, ettersom serveren som foretar sendingen kun sender ut én datastrøm, som så distribueres til mange brukere. Sammenlikn unicast.

Oppløsning

Det maksimale antallet bildepunkter (pixler) som kan vises samtidig på en skjerm, videokanon etc. Oppgis en skjermoppløsning på 640 x 480, betyr dette at skjermen kan vise 640 punkter i bredden og 480 punkter i høyden. Jo høyere oppløsning, jo mer skarpt og detaljert blir skjermbildet.

Protokoll

Dette er det samme som en definert kommunikasjonsstandard. Den forteller hvordan "boksene" på nettet skal snakke sammen.

Samtid (Realtime) streaming

Streamet video som lastes ned i sanntid og som automatisk justerer kvaliteten av videostreamen etter tilgjengelig båndbredde. Nettbrukerne med trege linjer vil få lavere bildekvalitet og/eller hakkete bevegelser. Real time streaming gjør det mulig å spole filmen fram og tilbake. Se også Progressiv streaming.

Teleconferencing

Forsyne telekommunikasjonslinjer med audio, video og grafiske egenskaper. Fysisk atskilte grupper kan "motes" ved å benytte toveis sanntidskommunikasjon over telenettet.

Unicast

Kringkastings-metode der serveren sender separate videostreamer til hver av deltakerne i en konferanse. Unicast er dermed en svært ressurskrevende måte å kringkaste på, ettersom belastningen på serveren og kravet til båndbredde er direkte proporsjonalt med antall deltakere. Motstykket til dette er multicast, hvor serveren sender én datastrøm til mange deltakere samtidig.

Kaskadekobling

Konferanser hvor de deltakende terminalene foretar multicast oppkobling til de terminalene som ikke har innebygd MCU.

Videokonferanse

En samling av teknologier som integrerer video med audio- og/eller dataoverføring




i sanntid over store fysiske avstander. Benyttes til blant annet møter og kompetanseheving mellom grupper som er fysisk atskilte over store avstander.

XLR-plugg

Profesjonell multikontakt med låsemekanisme. Normalt 3 pinner, for balanserte lydsignaler.

9 Vedlegg

1. CD

	Navn
▶	 Arbeid
▶	 Programmering
▶	 Prosjektgjennomføring