

Försättssida (färdigställs av Nada)

Namn: Jin Kjölberg

Svensk titel:

Känselgränssnitt i virtuella samarbetsapplikationer: Grafiska och haptiska designkriterier för en virtuell samarbetsmiljö anpassad för att stödja säker informationshantering och överlämnande av objekt.

Exjobbsämne: Människa-datorinteraktion (MDI)

Handledare: Eva-Lotta Sallnäs

Examinator: Yngve Sundblad

Uppdragsgivare: IPLab och CID, Nada, KTH

Datum: 2002-02-12

Examensarbetets titel

Känslgränssnitt i virtuella samarbetsapplikationer: Grafiska och haptiska designkriterier för en virtuell samarbetsmiljö anpassad för att stödja säker informationshantering och överlämnande av objekt.

Sammanfattning

Syftet med detta examensarbete var att ta reda på hur man bör utforma en virtuell samarbetsmiljö där återkoppling som stödjer känslsinnet ges, så kallad haptisk återkoppling, för att stödja säker informationshantering och överlämnande av objekt. Arbetet har innehållit en genomgång av relaterad forskning och litteratur, framtagning av relevanta designkriterier för det grafiska och haptiska användargränssnittet samt funktioner för miljön. Dessutom har en prototyp implementerats i två versioner, fri samt kontrollerad. De två versionerna hade samma grafiska och haptiska användargränssnitt. Den kontrollerade miljön innehöll utökad haptisk återkoppling och funktioner som syftade till att stödja en säker informationshantering och ett säkert överlämnande av objekt. Prototypen utvärderades genom en användarutvärdering som bestod av parvis uppgiftslösning i båda miljöerna samt en intervju. Resultaten visade att den fria miljön var något mer ändamålsenlig än den kontrollerade miljön när försökspersonerna inte visste skillnaderna mellan de två miljöerna. I båda miljöerna upplevdes dock interaktionen som intuitiv och naturlig.

The Master's thesis title

Haptic force feedback in collaborative virtual environments: Graphical and haptical design issues for a collaborative virtual environment designed in order to support safe information handling and handing over objects.

Abstract

The aim of this Master's project was to find out how to design a haptic collaborative virtual environment (CVE) in terms of graphical user interface, haptic force feedback and functionalities in order to increase the perceived safety and the actual safety in handing over an object between persons interacting in the environment. The work has consisted of a review of related research and literature, development of relevant graphical and haptical design issues and implementation of a prototype. The prototype was implemented in two versions, free and constrained. Both versions had the same graphical and haptical user interface. The constrained version was developed with additional haptic feedback and functionalities in order to support a safe hand over. To evaluate the application I designed and conducted a user study that consisted of problem solving in both the versions of the CVE and an interview. The results showed that the free version was slightly more effective than the constrained version when the user was not informed about the differences. The users perceived the interaction in both the versions as intuitive and natural.

Förord

Denna rapport beskriver ett examensarbete i människa-datorinteraktion som har avslutat mina studier i teknisk fysik vid Kungliga tekniska högskolan (KTH). Arbetet har utförts vid IPLab (Interaktions- och presentationslaboratoriet), Nada, KTH på uppdrag från IPLab och CID (Centrum för användarorienterad IT-design).

Mitt examensarbete har ingått som en del i ett större forskningsprojekt vid IPLab under ledning av professor Kerstin Severinson-Eklundh.Handledare har varit Eva-Lotta Sallnäs och examinator har varit Yngve Sundblad.

Jag vill passa på att rikta ett stort tack till Eva-Lotta Sallnäs för mycket givande diskussioner och synpunkter. Ett stort tack går också till Johan Hillström och Adam Nybäck vid Reachin Technologies AB för hjälp och råd med implementeringen. Jag vill även tacka CID för ekonomiskt stöd under arbetet samt vid deltagande i Eurohaptics 2001 och Reachin User Group Meeting 2001.

Jin Kjölberg

Stockholm, december 2001

Innehållsförteckning

KAPITEL 1 – INLEDNING	9
1.1 Examensarbetets syfte	9
1.2 Arbetets innehåll	10
KAPITEL 2 – RELATERAD FORSKNING OCH TEORI.....	13
2.1 Virtuella miljöer.....	13
2.2 Användbarhet.....	14
2.2.1 Förtroende.....	16
2.2.2 Samarbete	18
2.2.3 Multimodal interaktion	20
2.2.4 Haptik	22
2.2.5 Design av multimodala virtuella samarbetsmiljöer	24
KAPITEL 3 – PROTOTYPENS ÖVERGRIPANDE DESIGN.....	29
3.1 Generella designkriterier	29
3.1.1 Skillnad mellan fri och kontrollerad miljö	30
3.2 Specifika designaspekter	30
3.2.1 Användbarhet.....	30
3.2.2 Samarbete	30
3.2.3 Återkoppling	31
3.2.4 Funktioner	31
3.2.5 Arkitektur	31
KAPITEL 4 – PROTOTYPKONSTRUKTION.....	33
4.1 Ursprungliga krav	33
4.1.1 Användarscenarier	34
4.2 Analys.....	34
4.2.1 Utrustning	35
4.2.2 Miljöns utformning	37
4.2.3 Interaktion och återkoppling.....	38
4.2.4 Specificerade scenarier	39
4.3 Design	40
4.3.1 Systemarkitektur	41
4.3.2 Grafiskt användargränssnitt.....	41
4.3.3 Haptiskt användargränssnitt	42
4.3.4 Fri respektive kontrollerad miljö	43
4.4 Implementering	43
4.4.1 Datastrukturer	44
4.4.2 Grafiskt användargränssnitt.....	44
4.4.3 Haptiskt användargränssnitt	46
4.4.4 Svårigheter	47
4.5 Test.....	49
KAPITEL 5 – RESULTAT AV PROTOTYPUTVECKLINGEN.....	51
5.1 Användargränssnittet.....	51
5.1.1 Grafiskt användargränssnitt.....	51
5.1.2 Haptiskt användargränssnitt	52
5.2 Funktioner	53
5.3 Skillnader mellan den fria och den kontrollerade miljön	53
5.4 Datastrukturer	54
5.4.1 CVE-noden	54

5.4.2 Shelf-noden	54
5.4.3 Object-noden	55
5.5 Kända problem	55
5.5.1 Önskvärda ändringar	56
KAPITEL 6 – ANVÄNDARUTVÄRDERING	57
6.1 Metod	57
6.1.1 Design av studien	58
6.1.2 Försökspersoner	59
6.1.3 Utrustning	60
6.2 Resultat från användarutvärderingen	61
6.2.1 Kvantitativa resultat	61
6.2.2 Resultat från videoinspelningen	62
6.2.3 Resultat från intervjuerna	62
6.3 Analys av resultat från användarutvärderingen	65
6.3.1 Användbarhet	65
6.3.2 Samarbete	65
6.3.3 Grafiskt gränssnitt	65
6.3.4 Haptiskt användargränssnitt	66
6.3.5 Fri respektive kontrollerad miljö	67
6.4 Slutsatser från användarutvärderingen	67
KAPITEL 7 – AVSLUTNING	69
7.1 Sammanfattning och slutsatser	69
7.2 Haptik i framtiden	70
REFERENSER	71
BILAGA 1 – VRML-FILER	75
BILAGA 2 – HEADER-FILER	83
BILAGA 3 – UPPGIFTSLYDELSE FÖR FÖRSÖKSPERSONER (BLÅ AVATAR)	87
BILAGA 4 – INTERVJUFRÅGOR	91

Kapitel 1 – Inledning

Datorer och annan teknisk utrustning spelar en allt större roll i människans vardag. Vi använder teknik som hjälpmedel, leksaker och kommunikationsmedel. Ju mer teknik vi omger oss med desto viktigare är det att den är ändamålsenlig och lätt att använda och interagera med. Forskningsområdena som behandlar interaktion med datorer och maskiner heter människa-datorinteraktion, MDI (eng. HCI), respektive människa-maskininteraktion, MMI (eng. HMI).

Även distansarbete och därmed samarbete via datorer och nätverk har blivit allt vanligare. Forskningsområdet som behandlar datorstött samarbete förkortas CSCW som står för Computer supported cooperative work. Vid datorstött samarbete är de vanligaste kommunikationssätten text, telefon och video. Det finns kommersiella applikationer för till exempel telefon- och videokonferenser samt elektronisk post och dokumentdelning. Dessa miljöer baseras på tvådimensionell visuell samt auditiv information. Det finns även tredimensionella miljöer som ger stöd för delade dokument och manipulering av gemensamma objekt. Dessa miljöer kallas ofta för virtuella samarbetsmiljöer (eng. CVE) och ger möjlighet till att fokusera på deltagarna och deras egenskaper och uppgifter.

I virtuella samarbetsmiljöer har det tidigare varit vanligt med endast visuell och auditiv interaktion. Numera finns det även utrustning som stödjer interaktion via känsel-sinnet, så kallad haptisk interaktion. Ordet haptisk har sitt ursprung från grekiskans "haptiko's" som betyder "möjlig att förnimma med känselsinnet" [50]. Haptisk förklaras som "egenskapen hos en form att framstå särskilt tydlig för beröringssinnet, i motsats till optisk". Ett flertal studier har visat att den haptiska interaktionen kan ge användaren mervärden i form av ökad realism och effektivitet [2, 20, 36, 42]. Det har även visats att den påverkar ett samarbete positivt.

Detta examensarbete behandlar överlämnande av objekt i en haptisk virtuell samarbetsmiljö.

1.1 Examensarbetets syfte

Mitt första möte med haptisk interaktion var hösten 2000 i anknäytning till ett studiebesök vid företaget Reachin Technologies AB. Jag blev oerhörd fascinerad av denna form för interaktion och bokstavligen talat, känslan detta gav. Jag tog därför kontakt med Eva-Lotta Sallnäs, doktorand vid IPLab (Interaktions- och presentationslaboratoriet) vid KTH. Hon forskar inom området multimodal kommunikation i virtuella miljöer och har tidigare genomfört studier i haptiska samarbetsmiljöer. I anknäytning till hennes pågående forskningsprojekt önskade hon att utveckla en samarbetsmiljö för vidare studier av haptisk återkoppling.

IPLab är moderlaboratorium till kompetenscentret CID (Centrum för användarorienterad IT-design) vid KTH och bidrar till mycket av CIDs verksamhet. Företaget Saab är en av CIDs intressenter och i samarbete med dem diskuterades ett möjligt applikationsområde för haptiska samarbetsmiljöer. Saab Future Products arbetar med

utveckling av en prototyp för framtida ledningscentraler, Saab NetDefence¹. Systemet ska kunna användas för civila och militära ändamål. De önskade därför att ta reda på om och eventuellt hur haptisk återkoppling påverkar kommunikation och interaktion i en samarbetsmiljö.

En central uppgift i NetDefence är att överlämna och ta emot information. Denna uppgift har bland annat två aspekter. För det första är det viktigt att informationen som överlämnas är säker, det vill säga att informationen kommer från rätt källa och innehåller korrekta upplysningar. En annan aspekt är att själva överlämnandet sker säkert, det vill säga att mottagaren verkligen tar emot det sändaren ger och att information inte försvinner. I dag använder militären olika tvådimensionella grafiska figurer för att symbolisera information. Genom att kombinera dessa symboler på olika sätt kan man skapa och beskriva mer eller mindre komplex information. Det var därför önskvärt att utveckla en miljö som gav stöd för säker informationshantering och överlämnande av objekt.

Syftet med detta examensarbete blev därför att utveckla en generell, virtuell samarbetsmiljö för att kunna studera effekten av haptisk återkoppling vid överlämnande av objekt. Objekten skulle vara möjliga att kombinera på olika sätt. Uppgiften har bestått i att ta fram relevanta designkriterier för en sådan miljö, att implementera en prototyp av miljön samt att designa och genomföra en användarutvärdering av prototypen.

1.2 Arbetets innehåll

Det finns ett antal väldefinierade metoder för användarcentrerad utveckling. Det har dock utvecklats endast ett fåtal haptiska virtuella samarbetsapplikationer och det fanns därför ingen utprövad metod för framtagning av en sådan. Eberts beskriver fyra olika angreppssätt för design och utveckling av användargränssnitt: experimentellt, kognitivt, förutseende modellering och antropomorft [12]. Dessa angreppssätt kan användas var för sig eller integreras i olika delar av en process där syftet är att ta fram och utveckla användargränssnitt. I detta arbete har jag valt att följa Eberts utvecklingsprocess. I varje steg har jag själv valt en lämplig användarcentrerad metod, eftersom Eberts inte ger exempel på några sådana.

Utvecklingsprocess består enligt Eberts av följande steg:

- Problemanalys
- Övergripande design
- Prototypkonstruktion
- Användartester och -utvärderingar
- Slutliga användartester

Tabell 1 innehåller en översikt över de olika stegen i prototyputvecklingen samt deras innehåll. Dessutom beskrivs vilka arbetssätt och metoder jag valde att använda i respektive steg.

¹ Se Saabs hemsida, <http://www.saab.se>, för mer information.

Tabell 1. Översikt över utvecklingsprocessens innehåll och metoder.

Utvecklingssteg:	Innehåll:	Arbetsätt/Metod:
Steg 1: Problemanalys	Analysera problemet och ta reda på vad som är centralt i en haptisk virtuell samarbetsmiljö samt inhämta relevanta förkunskaper.	Uppgiftsanalys med antropomorft angreppssätt
Steg 2: Övergripande design	Baserad på analysen i steg 1 och arbetets syfte, bestämma vilka generella krav som ska finnas på utseende, återkoppling och funktioner i miljön.	Uppgiftsanalys.
Steg 3: Prototyputveckling	Med utgångspunkt i tillgänglig utrustning samt den övergripande designen, utveckla och implementera en prototyp av miljön.	Metod för objektorienterad mjukvaruutveckling
Steg 4: Användartester och -utvärdering	Formulera och genomföra användarutvärdering av den implementerade prototypen samt analysera resultaten och föreslå förbättringar och/eller ändringar.	Tänka högt samt riktad öppen intervju (semikontrollerat experiment)

Problemanalysen inleddes med en uppgiftsanalys av examensarbetets grundläggande syfte. Analysen resulterade i att ett antal olika relevanta ämnesområden utpekades. Sedan studerades publicerad litteratur inom varje område. Litteraturen sammanställdes och fungerade som bakgrundskunskap inför formuleringen av övergripande designkriterier. Vidare specificerades krav för en prototyp. Dessa krav fungerade sedan som underlag för analys och design samt implementering av miljön. Slutligen utfördes en användarutvärdering av systemet. Försökspersonerna fick då utföra ett antal uppgifter i miljön varefter de intervjuades.

Det finns ett antal metoder för utveckling av mjukvara och datasystem. De flesta metoder innehåller dock faserna kravspecifisering, analys, design, implementering och test [8]. Prototyputvecklingen lades därför upp enligt detta sätt.

Kapitel 2 – Relaterad forskning och teori

Människa-datorinteraktion (MDI) är ett tvärvetenskapligt forskningsområde som inkluderar kunskaper från många ämnesområden. Några av dessa är antropologi, datalogi, ergonomi, etnologi, filmvetenskap, grafisk design, industridesign, lingvistik, pedagogik, psykologi och sociologi. Datorstött samarbete (CSCW) är ett eget forskningsområde som är starkt knutet till MDI. Inom CSCW-forskning fokuserar man bland annat på hur människor kommunicerar och samarbetar med hjälp av datorer, det vill säga människa-dator-människainteraktion.

Examensarbetets syfte var att utveckla en generell och användbar virtuell samarbetsmiljö för att kunna studera effekten av haptisk återkoppling vid överlämnande av objekt. För att ta fram relevanta designkriterier för en sådan miljö gjordes en uppgiftsanalys. Uppgiftsanalys är en metod där man studerar de handlingar och/eller kognitiva processer en användare måste utföra för att kunna lösa en given uppgift [11, 51]. Syftet med metoden är bland annat att få en förståelse för informationsflödet i miljön samt ta fram lämpliga funktioner som bör finnas i miljön.

För att kunna överlämna ett objekt måste två eller flera personer samverka. En viktig aspekt att studera var därför samarbete. Centralt i ett samarbete är förtroendet mellan inblandade parter. I en överlämningsituation är det speciellt viktigt att båda parterna kan lita på varandra. Annars kan det vara svårt att genomföra överlämnandet effektivt och säkert. Det är dessutom viktigt att den aktuella miljön är användbar och ändamålsenligt utformad. Man måste därför diskutera hur en virtuell miljö ska utformas med hänsyn till realism, interaktion och återkoppling samt grafisk utformning.

Det aktuella problemet kunde därmed delas upp i fem huvudsakliga ämnesområden:

- virtuella miljöer
- användbarhet
- samarbete och förtroende
- multimodal interaktion, speciellt haptisk interaktion
- design av virtuella samarbetsmiljöer

För att inhämta tillräckligt med kunskaper inom de fem ämnesområden valde jag att söka och läsa publicerad litteratur. De viktigaste resultaten redovisas här.

2.1 Virtuella miljöer

Virtuell verklighet (VR) och virtuella miljöer kan definieras som ”datorgenererad skenvärld i vilken användare upplever sig vara och agera” [50]. Man kan skilja på skrivbordsbaserad VR (eng. desktop VR) och innesluten VR (eng. immersive VR). Ett ytterligare begrepp är utökad verklighet (eng. augmented reality) som är en blandning av reell och virtuell verklighet. Man befinner sig i och interagerar med den reella världen men har även möjlighet att manipulera virtuella objekt.

Skrivbordsbaserad VR innebär att man kan använda en vanlig dator med tillhörande bildskärm. Ytterligare utrustning kan vara stereoglasögon, 3D-möss eller utrustning

för haptisk återkoppling. Innesluten VR innebär däremot att man känner sig fullständigt "inuti" den virtuella miljön. Om man till exempel rör sig framåt kommer objekten man studerar att förstoras. Oftast innebär det användning av VR-hjälm eller liknande utrustning.

Virtuell verklighet har till tider fått mycket uppmärksamhet och förväntningarna på vad detta kan leda till har varit stora. Numera har man möjligen en mer realistisk syn på vad man kan åstadkomma med hjälp av denna teknik. Idag finns det till exempel kommersiella VR-applikationer inom spelindustrin, utbildning, design och formgivning, formella och informella möten samt underhållning. Bland annat på grund av Moores lag² har skrivbordsbaserad utrustning utvecklats kraftigt på mycket kort tid. Med denna utrustning kan man i flera sammanhang skapa en mycket effektiv virtuell miljö.

Capps m.fl. har definierat ett tillägg till Moores lag: "Det finns en tidpunkt efter vilken Moores lag fortfarande är giltig men inte relevant." De hävdar att man i framtiden inte kommer att begränsas av datorns beräkningskapacitet [10]. Därför kommer själva innehållet att bli nyckeln till framgång. En viktig faktor i detta är även hur användbara applikationerna som utvecklas är.

2.2 Användbarhet

Användbarhet är ett centralt begrepp inom MDI-området. Gleiss påpekar att definitionen av användbarhet i hög grad beror på vad och hur man mäter [16]. Communications Study Group, CGS, i London gjorde 1974 en användarstudie där begreppen effektivitet (eng. effectiveness) och acceptansgrad (eng. acceptability) användes som två dimensioner på användbarhet. Effektivitet betecknade hur väl man löste en given uppgift med en given apparat och acceptansgrad betecknade hur mycket man tyckte om att använda apparaten i fråga. Dessa två faktorer kan bero av varandra på mer eller mindre komplexa sätt. Dessa två faktorer var enligt CGS helt oberoende av varandra.

De ovan nämnda begreppen har tagits in i ett ISO-dokument CD 9241-11 "Usability statements". Den nya ISO-definitionen för användbarhet är [16]:

Usability: "A concept comprising the effectiveness, efficiency and satisfaction with which specified users can achieve specified goals in a particular environment".

Vidare definieras komponenterna *efficiency*, *effectiveness* och *satisfaction* på följande sätt:

Efficiency: "Measures of the accuracy and completeness of goals achieved relative to the resource (e.g. time, human effort) used to achieve the specific goals".

Effectiveness: "Measure of the accuracy and completeness of the goals achieved".

² Moores lag: Var 18:e månad fördubblas kapaciteten och halveras kostnaden för en given integrerad krets.

Satisfaction: “Measure of the comfort and the acceptability of the work system to its users and other people affected by its use”.

ISO-standaren skiljer därmed mellan efficiency som kan ses på som analogt med produktivitet och effectiveness som kan översättas med ändamålsenlighet. Dessa två begrepp betraktas som prestationsmått medan satisfaction som kan översättas med tillfredsställelse eller acceptansgrad kan ses som attitydmått.

Vid telekommunikation person till person har man enligt Gleiss kommit fram till att acceptansbegreppet även omfattar en aspekt som kallas personlig närvarokänsla (eng. personal presence). Denna aspekt varierar mycket i olika system och är därför en lämplig utgångspunkt för frågeformulär. Man ber försökspersonerna att på en skala markera till exempel hur varm, personlig, känslig och social miljön i fråga uppfattades [44]. Denna form av närvaro betecknas av andra som social närvaro i motsättning till virtuell närvaro. Virtuell närvarokänsla handlar om i hur hög grad man upplever sig vara i den virtuella världen trots att man befinner sig fysiskt på en annan plats [42]. Begreppet omfattar även hur man känner att man har kontroll över sin interaktion. Ett annat begrepp som kallas medvetenhet (eng. awareness) beskriver hur väl man får information om vilka aktiviteter som pågår i systemet. I miljöer där flera personer interagerar omfattar detta även information om andra personers aktiviteter [3, 15, 29, 43].

En ytterligare aspekt inom användbarhet är apparatens utseende. Gleiss ger exempel på att utseendet kan vara avgörande för om en produkt används eller inte. I detta sammanhang kan man även tala om hur apparaten känns att ta på och interagera med. Det betyder att användarens subjektiva värdering av faktorerna tillfredsställelse och ändamålsenlighet är vad som avgör en apparats användbarhet. Denna uppfattning får stöd av Karvonen som hävdar att webbsidors utformning och estetik påverkar människors förtroende för innehållet och avsändaren [25]. Hon visar även exempel på att utseendet påverkar hur vi uppfattar systemen. Om ett system visar upp bra prestanda så accepterar användaren ett tråkigt utseende. Men om prestanda är dålig och utseendet påkostat kan detta förvärpa användarens uppfattning av systemet: Användaren kan tro att utseendet ska försöka dölja den dåliga funktionaliteten.

Vad som uppfattas som snyggt och estetiskt beror dock på kulturell bakgrund, ålder och tidigare erfarenheter. Karvonen påpekar att framtida multimodala gränssnitt kommer att ställa stora krav på nytänkande kring gränssnittsdesign och att den estetiska aspekten kommer att vara mycket viktig i detta arbete.

Flera studier har dessutom visat att samtidig stimulering av känselsinnet och synsinnet kan öka ett systems användbarhet signifikant. Detta gäller både effektivitets- och acceptansaspekten. Till exempel har det visats att haptisk återkoppling i grafiska gränssnitt reducerar felfrekvensen [33, 34, 35]. Studier visar dessutom att stimulering av känselsinnet ger ökad närvarokänsla samt förbättrad prestation i samarbetsmiljöer [2, 20, 36, 42].

2.2.1 Förtroende

Att etablera förtroende är en viktig del i en samarbetsprocess. Dels måste man ha förtroende för systemet man använder och samarbetar i och dels för andra personer som interagerar i systemet.

Mänskligt förtroende

Muir diskuterar aspekter kring människa-maskininteraktion som måste tas hänsyn till i design av beslutsstöd [32]. Hon påpekar att det är viktigt att användaren får förtroende för systemet för att det ska användas. Det är också viktigt att användaren får en korrekt uppfattning om systemet för att undvika över- eller underskattning av dess kapacitet. Eftersom det finns få systematiska studier som behandlar förtroende mellan människor och maskiner utgår Muir från studier som har modellerat förtroende mellan människor.

Muir hänvisar till ett antal olika definitioner av förtroende mellan människor i litteraturen. Hon utgår från dessa och föreslår följande definition av förtroende mellan människa och dator:

”Förtroende är förväntningen från en medlem av systemet, på en bibehållenhet av naturliga och moraliska ordningar, på en teknisk kompetent prestation och ett förtroendefullt ansvar från en medlem av systemet, och är relaterat till men inte nödvändigtvis isomorft³ med, objektiva mått av denna kvantifiering.”

Hon behandlar också dynamiken i förtroendet mellan människa och maskin. Förtroendet är tidsberoende och även beroende av användningen av systemet. Hon hävdar att för att kunna avgöra om en person är tillförlitlig eller inte, måste personen i fråga ha uppträtt i en situation där det var möjligt att vara otillförlitlig. Det vill säga han eller hon måste ha passerat ett test. Vidare är det viktigt att ta reda på hur förtroendet kan behållas och växa samt vilka faktorer som bidrar till att förtroendet försvinner och hur det eventuellt kan återskapas. Enligt Muir finns det mycket liten kunskap om detta både när det gäller förhållanden mellan människor och mellan människor och maskiner.

Muir hävdar dessutom att en person måste bestämma sig för om han eller hon vill lita på ett system från början. Hur förtroendet utvecklas beror på hur systemets funktionalitet passar ihop med användarens förväntningar. Människor väljer dock oftast att från början ha förtroende för nya sociala förhållanden. Det beror på att det är mindre mentalt krävande att lita på, än att inte lita på förhållandet i fråga.

Förtroende i fleranvändarapplikationer

Jones m.fl. diskuterar förtroende i datorsystem som stödjer samarbete, så kallad groupware [24]. De hävdar att tidigare har groupware-utvecklingen varit helt teknikstyrd och aspekter som förhållanden mellan människor, roller och sociala regler har inte tagits hänsyn till i designen.

³ Två företeelser eller domäner sägs vara isomorfa om de trots materiella olikheter uppvisar samma struktur i relevanta avseenden [49].

För att bättre kunna ta hänsyn till sociala förhållanden i framtida groupware-applikationer har de försökt att definiera begreppet förtroende mellan individer i en sådan miljö. De anser att förtroende är en nyckelfaktor i frågan om effektivitet i aktiviteter både inom gruppen och emellan grupper. Förtroende kan dessutom formaliseras och användas i utveckling och utvärdering av system för datorstött samarbete. I sin artikel presenterar de en matematisk modell för detta som de kallar Trust.

De påpekar att det finns lite forskning om förtroende i datorsystem och kombinerar därför olika synsätt på förtroende från tidigare litteratur. Deras definition av förtroende baseras sedan på följande:

- användbarhet spelar roll för ett förtroendefullt beteende, dvs. att användningen av ett arbetssammanhang är direkt relaterat till sannolikheten att en individ ska samarbeta med ett annat i det givna sammanhanget
- individer antar att andra individer uppvisar ett beteende liknande sitt eget
- förtroende handlar om att hantera komplexa situationer och risker
- förtroende gör det lättare att förutse andra personers framtida beteende, givet att det finns en gemensam historia där personerna har visat varandra förtroende och att man använder det förtroendet för att resonera kring framtida handlingar
- det kan finnas olika nivåer av förtroende

Jones m.fl. föreslår vidare ett antal byggstenar på vilka Trust baseras. Dessa kan tjäna som en gemensam grund och vokabulär i diskussioner kring förtroende i system för datorstött samarbete:

- människor, dvs. individer och grupper som interagerar i miljön
- sammanhanget i vilket människorna interagerar
- grundläggande förtroende, t.ex. baserat på tidigare erfarenheter
- kunskap om andra människor i miljön
- generellt förtroende oberoende av sammanhangen
- förtroende i sammanhanget baserat på tidigare erfarenheter i samma sammanhang
- vikten av samarbetet, specifik för varje människa och sammanhang
- användningsvärdet av samarbetet dvs. förväntat värde, specifik för varje människa

För flera detaljer kring den matematiska modelleringen hänvisas till artikeln i sin helhet.

I litteraturen har jag hittat endast några exempel på studier där förtroende mellan människor i virtuella miljöer har behandlats. Jensen m.fl. har till exempel undersökt beroendet mellan förtroende och kommunikationsmodalitet [23]. Studien visade att valet av kommunikationsmodalitet var viktigt för att uppnå stöd för önskad aktivitet. Försökspersonerna tyckte mer om och litade mer på den andra deltagaren om de hade möjlighet till att kommunicera med varandra. Detta mättes med hjälp av enkäter där försökspersonerna skulle värdera sin motspelare i förhållande till *trevlighet* (trevlig, snäll, vänlig, varm) och *trovärdighet* (ärlig, rättfärdig, pålitlig, uppriktig).

I virtuella miljöer kan människor representeras på olika sätt. Hur man utformar representationen kan också påverka hur personen bakom representationen uppfattas och

därmed förtroendet för personen i fråga. Detta behandlas mer utförligt under rubriken *Representation av människor*.

2.2.2 Samarbete

Man kan skilja på samarbete i reella miljöer, till exempel möten ansikte mot ansikte, och samarbete i virtuella miljöer, till exempel datorspel. Man kan också skilja på olika typer av samarbete, till exempel informationsdelning, problemlösning och undervisning. Faktorer som hur många personer som samarbetar, deras förhållande till och kännedom om varandra, hierarkier, status och roller påverkar också hur ett samarbete fungerar. För att designa användbara virtuella samarbetsapplikationer kan det vara nyttigt att studera hur människor samarbetar i reella miljöer, med andra ord använda ett antropomorft angreppssätt.

Samarbete i reella miljöer

Olson m.fl. har analyserat reella designmöten på två olika arbetsplatser [37]. Syftet med mötena var att ta fram och diskutera designkriterier för en mjukvara. I mötena deltog mellan tre och sju personer och tidsåtgången varierade mellan en och två timmar. Alla deltagarna kände varandra sedan tidigare och hade vana från grupparbeten. De var placerade kring ett bord och hade endast papper, penna, wyteboard o.dyl. som hjälpmedel. Deltagarna hade olika kompetenser och i de flesta mötena deltog en nyanställd.

Resultaten visade att två huvudgrupper av aktiviteter uppstod: relevant designaktivitet och administrationsaktivitet. Ungefär 40 procent av tiden användes till att diskutera design. Medan 30 procent av aktiviteterna var sammanfattningar och genomgångar. Även om de sistnämnda aktiviteterna hade en viss anknytning till design, verkade det som om deras funktion huvudsakligen var att koordinera och generera diskussion om vidare arbete. Gester och ansiktsuttryck användes också i hög grad som kommunikationsmedel och var en del av koordineringen.

Trots stora skillnader i de två företagens arbetssätt och tradition, visade resultaten att designmöten är ganska organiserade. Det var dessutom vanligt att de olika deltagarna gick in i olika roller. Detta är ett sätt att minska den kognitiva belastningen för varje deltagare.

Lenman m.fl. har i en rapport behandlat funktionsbehov och designkriterier för virtuella samarbetsmiljöer [29]. Utredningen avser endast formella möten liknande de möten studerade av Olson m.fl. Enligt rapporten har ett reellt formellt möte fyra olika huvudsyften: informationsbearbetning, planering, utförandansvarsfördelning samt regelverk.

Det formella mötets form beror enligt Lenman m.fl. på hur många deltagare mötet har, vilken relation deltagarna har till varandra, hur ofta de träffas, deras motivation i mötet samt hur beslutsprocessen ser ut. En av mötets grundläggande funktioner är att skapa en gruppidentitet. Vidare ska det fungera som en scen för individerna i gruppen samt definiera deras roll inom gruppen och i verksamheten för övrigt. Detta gör det möjligt för deltagarna att skapa en gemensam referensram och förståelse för gruppens syfte, något som stimulerar till ett större socialt medvetande och en kreativ miljö.

Lenman m.fl. hävdar även att ett formellt möte är bästa forumet för att fatta beslut och tydliggöra ledarrollen.

Tre psykosociala komponenter som anses viktiga för att grupper ska kunna utföra sina uppgifter är kommunikation, koordinering och samarbete. Vid kommunikation spelar den använda modaliteten stor roll. Kommunikation via en rik modalitet (till exempel ansikte mot ansikte eller ljud och bild tillsammans) gör det lättare att lösa komplexa problem. Det påpekas också i rapporten att olika modaliteter är ändamålsenliga för olika typer av problem och uppgifter.

Samarbete i virtuella miljöer

I datorstött samarbete har man som tidigare nämnts fokus på hur två eller flera personer kommunicerar och samarbetar med hjälp av datorer. Mjukvara som stödjer detta kallas ofta för "groupware". Vanligt förekommande groupware-applikationer är till exempel e-post, synkron textkommunikation (tjatt), elektroniska anslagstavlor, gruppkalendrar och videokonferenser. De nämnda groupware-applikationerna är oftast tvådimensionella, textbaserade virtuella miljöer. När man talar om virtuella samarbetsmiljöer menar man ofta tredimensionella miljöer där varje person har en tredimensionell grafisk representation. Under rubriken *Design av multimodala virtuella samarbetsmiljöer* kommer sådana miljöer att diskuteras specifikt.

Två aspekter som är viktiga när det gäller datorstött samarbete är den geografiska dimensionen samt samarbetets form. Geografiskt kan deltagarna befinna sig på samma fysiska plats eller på olika platser, distribuerat. Man kan även skilja på om arbetet sker samtidigt (synkront) eller vid olika tider (asynkront). I litteraturen beskrivs kombinationer av dessa aspekter ofta som en tid/rum-matris [11, 13].

Bild 1 är ett exempel på en tid/rum-matris. E-post är till exempel ett asynkront distribuerat kommunikationssätt. Användarna kan skriva, skicka och läsa sina meddelande vid valfria tider och på valfri plats. Videokonferens är däremot ett synkront och distribuerat kommunikationssätt eftersom det kräver att användarna kommunicerar i realtid. Möten ansikte mot ansikte är ett exempel på synkront arbete på samma plats, medan en anslagstavla är ett exempel på asynkron kommunikation på samma plats.

	Samma plats	Olik plats
Samma tid	Samtal ansikte mot ansikte	Telefon Videokonferens
Olik tid	Virtuell/reell anslagstavla	Brev E-post

Bild 1. Tid-Rum-matris

Kupst m.fl. har tagit fram viktiga kriterier för virtuella samarbetsmiljöer baserade på egna erfarenheter av skrivbordsutrustning för videokonferenser o.dyl. [26]. De hävdar

att dagens utrustning ”... inte ger tillräckligt stöd och flexibilitet för att kunna designa, integrera och etablera samarbetsmiljöer som ska fungera på flera platser och plattformar inom företagen, på ett ekonomiskt och effektivt sätt”. De påpekar dock att en datorstött samarbetsmiljö som stödjer användning av gemensamma dokument och informationsdelning inte ska ses som en fullständig ersättning för rumsbaserade videokonferenser.

Salvador m.fl. har sammanställt arbete från en workshop med temat ”Groupware för design och utvärdering” [43]. Resultatet blev en mall för design av groupware-applikationer samt en identifiering av vilka aspekter som skiljer groupware från en-användar-applikationer. Fem aspekter som överensstämmer väl med tidigare refererad litteratur togs fram: människor och deras roller i en grupp, artefakter som produceras och används i interaktionen, uppgifter och aktiviteter, interaktiva situationer samt interaktiva sociala protokoll. Dessutom ställdes följande krav på groupware-applikationer:

- uppgiftsfokusering
- tillförlitlighet/förtroende
- låg kognitiv belastning
- användbarhet (hög acceptansgrad)
- stöd för medvetenhet i systemet

2.2.3 Multimodal interaktion

Som tidigare nämnts har det visats att ett systems användbarhet kan ökas när flera sinnen samverkar vid interaktionen. För att designa användbara interaktiva gränssnitt måste man därför ta hänsyn till de mänskliga sinnen och deras möjligheter och begränsningar.

Känslsinnet

För flera hundra år sedan namngav Aristoteles de fem sinnen syn, hörsel, känsel, lukt och smak. Begreppet sinne kan definieras som ”kroppsligt organ med förmåga att omvandla fysiska retningar till syn-, hörsel-, känsel-, lukt- eller smakupplevelser” [50]. Kroppen har även olika system som kombinerar flera sinnen. Ett exempel är balanssinnet som bygger på en kombination av visuell information, information förmedlad via känselkroppar i huden, muskler, sensor och leder, samt själva balansorganet i innerörat.

Känslsinnet stödjer två olika former av perception, taktil perception och kinestetisk perception. Den taktila perceptionen utgår från information från nervcellerna i huden och förmedlar upplevelser som vibrationer, tryck och temperatur. Kinestetisk perception baseras däremot på information från muskler, sensor och leder och deras förhållande till varandra.

Haptisk perception är ett begrepp som inkluderar både kinestetisk och taktil perception. Några exempel på haptisk perception är att känna på tyger, identifiera fasta objekt samt deras spatiala form och orientering i rummet eller en tandläkares undersökning av tänderna. Haptisk perception bidrar mycket till social och sexuell kommunikation och estetisk uppskattning av både konst och dagliga livet [6].

En av haptikens största fördelar jämfört med sinnen som syn och hörsel, är att känsel-sinnet är bilateralt [5]. Det betyder att man kan inhämta information från samt manipulera eller kontrollera ett objekt samtidigt. Med hjälp av synen kan man till exempel endast observera ett objekt, medan den haptiska återkopplingen ger möjlighet till att få information om bland annat strukturer, form och tyngd och samtidigt deformera eller flytta ett objekt.

Studier av multimodala gränssnitt

Barfield m.fl. gör en jämförelse av människans förmåga att använda olika modaliteter i en virtuell miljö [1]. Den beskriver utförligt de auditiva, visuella och taktil/kinestetiska (haptiska) modaliteterna och hur respektive modalitet stimuleras i miljön. Det framkommer bland annat att utan kinestetisk och taktil återkoppling saknar en virtuell miljö naturlighet och i flera tillfällen för detta till att användaren uppvisar sämre resultat.

Som tidigare refererats har Jensen m.fl. undersökt hur olika kommunikationskanaler påverkar samarbetet mellan personer i en virtuell miljö där följande situationer studerades: ingen kommunikation, synkron textkommunikation, text-till-tal och tal [23]. Resultatet visade bland annat att tal var den viktigaste kommunikationskanalen. Även Eberts [12] refererar till försök som visar att kommunikation blir mest effektiv om rösten används som kommunikationsmodalitet. Inga av dessa studier har dock undersökt hur en eventuell haptisk återkoppling kan påverka kommunikationen och samarbetet.

Basdogan m.fl. [2] samt Sallnäs m.fl. [42] har studerat hur haptisk återkoppling påverkar samarbetet i lösandet av en fysisk samarbetsuppgift. Basdogan m.fl. använder syn och känsel som kommunikationskanaler, medan Sallnäs m.fl. även använder tal. Sallnäs studie visar att haptisk återkoppling signifikant ökar den virtuella närvarokänslan men endast antyder en ökad social närvarokänsla. Basdogan m.fl. använder uttrycket samhörighet för att beskriva fenomen liknande social närvaro. I deras studie har de visat att den haptiska återkopplingen faktisk ökar även den sociala närvarokänslan. Skillnaden i påvisat social närvarokänsla kan bero på att den auditiva kommunikationen var dominerande i förhållande till den haptiska i Sallnäs studie.

Oakley m.fl. har studerat samarbete i miljöer som använder delade skärmar, det vill säga möjlighet till att dela dokument och samtidigt manipulera dokumentet [36]. De presenterar fem olika sätt att kommunicera haptisk: trycka, dra, lokalisera, ta tag i samt gestikulera. Kommunikationen utfördes med en grafisk representation av användarna som kallas avatar. I skrivbordsmetaforen är avataren oftast utformat som en pil och benämns muspekare eller markör.

Studien visade att även vid denna form för samarbete tillför haptisk återkoppling ökad spatial närvarokänsla, engagemang och naturlighet i interaktionen, jämfört med en icke-haptisk miljö. Även användbarheten ökade signifikant i den haptiska miljön trots att den upplevdes som mer fysiskt krävande. De flesta försökspersonerna tyckte däremot att det var svårt att använda den haptiska kommunikationen. Försökspersonerna hade möjlighet att kommunicera verbalt men hade ingen visuell kontakt. De visade också en tendens att verbalt varna andra deltagare innan en haptisk interaktion utfördes, även om denna inte påverkade personerna i fråga. Det kan enligt Oakley

m.fl. tyda på att personerna var ovana att få denna form av återkoppling samt att beröring upplevs som en mycket personlig aktivitet.

Överlämnande

I en studie där överlämnande av ett objekt studerades specifikt hävdar Ishii m.fl. att arbetets effektivitet influeras av det fysiska och mentala förhållandet mellan deltagarna och mellan deltagarna och objektet [20]. I försöket studerades en överlämningsuppgift i syfte att undersöka hur haptisk återkoppling påverkar effektiviteten. En situation där båda deltagarna hade kontakt med objektet studerades. Resultatet visade att den haptiska återkopplingen gav en mycket högre effektivitet samt en ökad känsla av intimitet mellan deltagarna. Ishii m.fl. hävdar att den haptiska återkopplingen hindrar deltagarna från att försöka flytta objektet i olika riktningar.

2.2.4 Haptik

Haptiska gränssnitt har som tidigare beskrivits en hel del fördelar jämfört med visuella och visuella-auditiva gränssnitt. Även om det kan vara svårt att återge en fullständigt naturtrogen interaktion har haptisk återkoppling visat sig vara mycket användbart i flera applikationer. I spelbranschen har så kallad force-feedback-enheter som ger kraftåterkoppling, använts i många år och även spelkontroller med inbyggd vibrator. Inom följande områden finns det i dag utvecklade applikationer och pågående forskning rörande haptiska gränssnitt [39, 40, 45]:

- medicin, t.ex. träning av kirurger⁴, hjälpmedel för funktionshindrade [21, 22]
- underhållning, t.ex. spel [9] och simulatorer
- utbildning, t.ex. studier av fysiska och kemiska fenomen [49] visualisering av komplexa data [38]
- industri, t.ex. i CAD- och andra utvecklingsmiljöer, prototypkonstruktion
- grafisk konst, t.ex. virtuella utställningar och museer, datorstödd formgivning⁵

För att förstå utmaningarna som ligger bakom haptisk återkoppling behöver man ha klart för sig vad man försöker simulera. Människans fysiska, kognitiva och perceptuella begränsningar måste alltid tas hänsyn till i utveckling av alla former av gränssnitt.

Mänsklig haptik

Den mänskliga handen består av 19 ben sammankopplade av en mängd friktionsfria leder, sensorer och muskler. Det hela täcks av mjuka vävnader och hud. Detta system har hela 22 frihetsgrader. Känslsinnet består som tidigare nämnts av ett stort antal receptorer och nerver i huden, leder, sensorer och muskler. Dessa stimuleras av mekanisk, termisk och kemisk aktivitet och ger upphov till taktil och kinestetisk perception [5, 45].

I litteraturen finns det redogörelser för hur små skillnader det mänskliga sinnet kan upptäcka när det gäller att uppleva strukturer, positioner, tryck, krafter o.dyl. (eng. Just Noticeable Difference, JND) [5, 47, 48]. Detta är faktorer man måste ta hänsyn till i

⁴ Reachin, <http://www.reachin.se/> (december 2001)

⁵ FreeForm, <http://www.sensable.com/> (december 2001)

utveckling och användning av haptisk utrustning. Även hur stora krafter en människa kan skapa och kontrollera samt bandbredden för respektive modalitet är viktigt att ta hänsyn till [48].

Haptisk utrustning

Knoll från Bell Labs var den första som på 1960-talet demonstrerade haptisk återkoppling i en virtuell miljö [40]. Sedan dess har det utvecklats en hel del interaktionshårdvara som stödjer haptisk återkoppling och den kan sägas ha två grundläggande funktioner:

- att mäta position och kontaktkrafter (samt ändringar över tiden) från användarens hand (eller annan kroppsdel) samt
- att för användaren återge kontaktkrafter och position (eller rumslig och tidsmässig fördelning).

Man kan skilja på utrustning som försöker återge en naturlig haptisk interaktion (till exempel med handen) och utrustning som interagerar via ett redskap. Vidare skiljer man på om utrustningen riktar kraften mot kroppen eller mot "världen" till exempel kroppsbasead respektive bordsbasead utrustning. Detta påverkar vilka krafter som kan genereras och därmed den haptiska återkopplingen [5].

Exempel på haptisk interaktionshårdvara som finns tillgängligt är bordsbasead utrustning som styrspakar och möss, samt kroppsbasead utrustning som handskar och taktill utrustning. Biggs m.fl. har gjort en översiktlig genomgång av några haptiska interaktionshårdvarors egenskaper, deras kostnad samt producenter [5]. Vidare finns det flera prototyper och icke-kommersiella haptiska interaktionshårdvaror som har använts i studier och försök [9, 20, 33].

Massie m.fl. beskriver designkriterier, kinematik och mekanik bakom det haptiska gränssnittet PHANTOM som utvecklades år 1993 vid Massachusetts Institute of Technology, USA [31]. Det produceras i dag av SensAble Technologies, Inc.

Det finns i dag ingen utrustning som fullständigt kombinerar både taktill och kinestetisk återkoppling av kraft [35]. PHANTOM sägs ge haptisk återkoppling men ger huvudsakligen kinestetisk kraftåterkoppling. Man kan dock känna vibrationer till följd av ytors struktur och detta kan uppfattas som taktill stimulans.

Datorhaptik

Datorhaptik som är analogt med datorgrafik, definieras som kunskapen om att skapa och rendera haptisk stimulans till den mänskliga användaren [40, 45]. Haptisk rendering kan beskrivas som processen att beräkna krafter baserade på interaktionsenhetens position. Salisbury m.fl. beskriver vilka element som måste finnas med i ett gränssnitt för att man aktivt ska kunna utforska och uppleva haptiska objekt [41]. Några exempel på detta är fri rörelse, bibehållen kontaktkraft, ytor med och utan friktion samt texturer. Även Srinivasan m.fl. beskriver liknande aspekter inom datorhaptiken [45].

För att kunna återge en realistisk och kontinuerlig återkoppling bör den haptiska renderingen uppdateras med en frekvens på 1000 Hz. Detta motsvarar bandbredden för det mänskliga taktila systemet [45]. En visuell grafisk miljö behöver endast renderas med en frekvens på 60 Hz för att återges på ett naturligt sätt.

I anknytning till ett ökande behov av distanssamarbete och användning av Internet, har haptisk återkoppling i nätverk blivit mer och mer aktuellt. Man har i flera försök simulerat nätverk för att undvika problem kopplat till fördröjning [2, 42]. Man kan till exempel använda en dator för att köra applikationen medan försökspersonerna befinner sig i olika rum. Ett annat sätt att kringgå svårigheter är att låta endast en användare i taget interagera med miljön [9, 20].

Oakley m.fl. hävdar att ett nätverk av två datorer på ett avstånd av några meter och med en kommunikationsfrekvens på 100 Hz, är tillräckligt för att effektivt använda haptisk kommunikation i en samarbetsmiljö [36]. För kommunikation över längre avstånd till exempel transatlantiska, kommer man till exempel att vara begränsad av ljusets hastighet. För att kunna skapa distribuerade virtuella miljöer med auditiv, visuell och haptisk återkoppling behöver det därför utvecklas standardiserade protokoll som inkluderar haptik explicit [46]. Även standarder för hur man hanterar fördröjningar i haptisk information krävs.

2.2.5 Design av multimodala virtuella samarbetsmiljöer

För att öka prestationen i en virtuell miljö är det viktigt att betrakta navigeringens komplexitet samt graden av närvarokänsla [47]. Det är dessutom viktigt att tänka på att ökad interaktion medför ökad kognitiv belastning som kan påverka prestationen negativt. Skillnader mellan användare måste också tas hänsyn till, till exempel mellan nybörjare och erfarna användare. I tillägg kan användarens personlighet spela roll, speciellt i komplexa sociala situationer som samarbetsapplikationer. Basdogan m.fl. redovisar resultat som antyder att hur man uppfattar en samarbetspartners kön kan bero på återkopplingen i miljön [2]. Till exempel tror fler användare att deras samarbetspartner är en man om haptisk återkoppling ges jämfört med endast visuell återkoppling.

Stanney m.fl. hävdar att det i dag inte finns några passande designmetaforer för virtuella miljöer [47]. Metaforer från traditionell MDI som till exempel skrivbord och menyer, baseras endast på tvådimensionell visuell återkoppling. Det är viktigt att gränssnittet uppmanar till effektiv interaktion [7, 47]. En bra metafor för virtuella miljöer bör därför stödja både visuell, auditiv och haptisk information. De föreslår vidare att man bör använda ett antropomorft tillvägagångssätt vid framtagning av designmetaforer för virtuella miljöer, då detta bidrar till mer intuitiva gränssnitt.

Arkitektur

Hur man utformar arkitekturen i en virtuell miljö beror på vad man vill utföra i miljön. Datorspel använder sig ofta av hela världar med flera rum eller platser. Vill man skapa stöd för mötesverksamhet kan det vara lämpligt att utforma världen som ett sammanträdesrum med bord, stolar, wyteboardtavlor o.dyl. Ska man däremot visualisera data kan det vara lika effektivt att använda en abstrakt representation [3]. Man ska dock komma ihåg att detaljer både kan utgöra belastning och stöd för en användare [29].

Den rumsliga strukturen påverkar hur lätt eller svårt det är att navigera i den virtuella miljön. En annan faktor är hur användaren "ser" miljön. Olika möjligheter är till exempel genom avatarens "ögon", första person-vy (eng. first person view). En första person-vy är ett uttryck som ofta används i datorspel. Det betyder att användaren ser

världen från sin representations position, det vill säga att kameran som återger det användaren ser placeras i representationens position. En annan form för vy är tredje person-vy. Då placeras kameran så att spelaren kan se sin egen representation i bild och samtidigt dennes relation till världen. De olika synvinklarna påverkar synfältets storlek samt upplevelsen av den egna och andra personers aktivitet.

Hindmarsh m.fl. har studerat hur synfältets storlek påverkar interaktionen i en objekt-fokuserad miljö [18, 19]. I den aktuella miljön är avataren modellerad som en människa och kameran är placerad precis bakom avatarens axel. Deltagarna ser därmed axlarna och huvudet på sin representation bakifrån samt världen i den riktning avataren är vänt. Detta ger användaren ett utökat synfält och ger återkoppling om avatarens tillstånd. Problem relaterade till det begränsade synfältet på 55° horisontellt samt 45° vertikalt, var bland annat svårighet i att uppfatta andras aktivitet samt att lokalisera objekt som det refererades till verbalt. Deltagarna hade dessutom svårt att föreställa sig den andra deltagares synfält. Detta ledde också till flera missförstånd mellan deltagarna. För att underlätta gemensamma referenser och undvika missförstånd kan man till exempel tydliggöra avatarens synfält med hjälp av ljussättning [14], det vill säga ljusmarkera det område som den andra användaren ser (se bild 2).

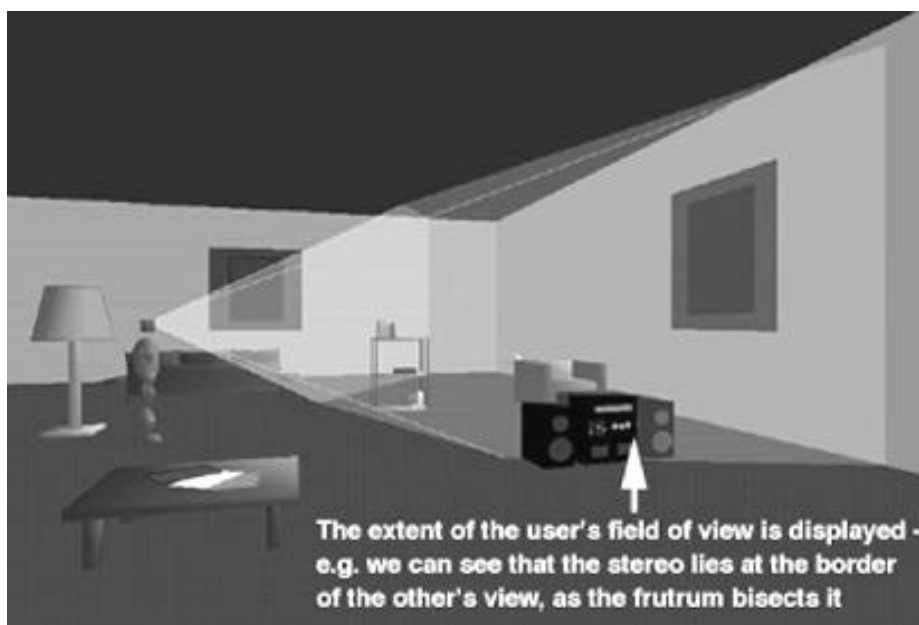


Bild 2. Ljusmarkering av en användares synfält.

Bilden är hämtad från artikeln *Object-Focused Interaction in Collaborative Virtual Environments* av Hindmarsh m.fl. [14].

Ett annat sätt att öka informationen om andras aktivitet och vad de refererar till är att låta alla deltagare ha samma representation av miljön [2, 42]. Detta fungerar bäst om miljön är begränsat till ett rum eller en plats och antalet deltagare är relativt litet. Ytterligare ett sätt är att tydliggöra aktiviteter både via avataren och det manipulerade objektet [19]. Detta kan till exempel göras med hjälp av färgmarkering av objekt eller en visuell koppling mellan objekt och avatar, det vill säga genom att fokusera på aktiviteten snarare än individen som utför aktiviteten.

Representation av människor

I den reella världen är kroppen ett av människans viktigaste kommunikationsmedel och används nästan alltid tillsammans med andra modaliteter [17, 38]. Vi använder gester, poser, huvudrörelser o.dyl för att medvetet eller omedvetet uttrycka känslor, intresse, identitet, tillgänglighet, personliga egenskaper o.s.v. För att kommunikationen i en virtuell miljö ska bli effektiv är det därför viktigt att ta vara på den icke-verbala kommunikationen.

Benford m.fl. har belyst ett antal faktorer som bör beaktas när det gäller hur deltagarna i VR-miljöer ska representeras [4]. De menar att en användare bör vara lämpligt förkroppsligad i miljön för att vara effektivt representerad inför sig själva och andra deltagare. En avatar kan enligt Benford m.fl. förmedla information om bland annat följande:

- tillgänglighet och närvarograd
- placering och riktning i förhållande till andra i rummet
- identitet
- pågående aktivitet samt var uppmärksamheten är riktad och vilket objekt som manipuleras
- aktivitetshistoria
- gester och ansiktsuttryck
- personliga egenskaper

Andra aspekter man måste ta hänsyn till är hur sanningsenligt och naturtroget avataren ska utformas, hur den ska representeras i olika medier samt om man kan befinna sig på flera platser samtidigt. Varje användare kan, beroende på sin datorcapacitet, vilja variera sin egen representation av andras avatarer med hänsyn till bland annat detaljnivå.

Benford m.fl. hävdar dessutom att det inte alltid är önskvärt att återge människor naturtroget utan att en enkel form som "blockies", två avlånga boxar formade som ett T, är tillräcklig för att ge uttryck för närvaro, placering och orientering. Hur viktigt det är med naturtroga avatarer är dock något som bör studeras mer.

Bowers m.fl. visar exempel på hur en enkel blockie-avatar används i en virtuell miljö [7]. För att stödja närvaro, orientering, tillgänglighet, egenskaper samt enkla gester gavs avataren ett öga samt öron och den kunde läggas ner. Användarens namn placerades ovanför avataren. Studien visar att avatarerna trots en enkel utformning och begränsade mänskliga egenskaper, har social betydelse i interaktionen mellan deltagarna och inte endast är ett navigeringsverktyg. Detta strider mot annan VR-forskning som förespråkar så naturtroga avatarer som möjligt.

Salem m.fl. [38] och Guye-Vuillème m.fl. [17] hävdar till exempel att en naturtrogen återgivning av deltagarna ökar användbarheten och möjliggör en mer naturlig interaktion i en virtuell miljö. De menar att det dessutom är ett krav för att kunna använda icke-verbala kommunikation effektivt. På samma sätt som Benford m.fl. [4] beskriver de en avatars uppgift relaterat till bland annat identitet, närvaro, uppmärksamhet och tillgänglighet.

Bowers m.fl. påpekar däremot att det är viktigt att avataren endast återspeglar faktiska egenskaper. Det betyder till exempel att den utrustas med öron endast om användaren har ljudåterkoppling. Vidare utveckling av avatarernas egenskaper är önskvärt men bör anpassas till datorkapacitet och nätverksprestanda.

McCarthy m.fl. har studerat avatrar i läromiljöer [31]. De påstår att en fullkroppsåtergivning av instruktören vore mest realistiskt men inte nödvändigtvis mest effektivt, både när det gäller datorkapacitet och användbarhet. Att endast använda en hand som avatar gav inte tillräckligt med icke-verbal kommunikation. Men det visade sig att en halvkroppsavatar med möjlighet att oberoende röra kropp, huvud och ögon, gav lämplig icke-verbal kommunikation i en given situation för lärande.

Ett problem relaterat till användarens representation är interaktionen mellan användaren och avataren. Hur kan man på ett effektivt sätt styra sin avatar utan att ta bort uppmärksamheten från uppgiften man utför? På detta område behöver det utvecklas verktyg för inmatning av instruktioner från användaren. Den kommersiella applikationen Active Worlds använder idag menyer för att välja beteende hos avataren. Salem m.fl. föreslår att man bör använda fördefinierade rörelser som aktiveras när bestämda nyckelord förekommer i en skriftlig kommunikation [38].

Kapitel 3 – Prototypens övergripande design

Bakgrunden till examensarbetets uppgiftsformulering var som tidigare nämnts en diskussion med Saab om utvecklingen av deras prototyp NetDefense. Prototypen är ett koncept som baseras på ett nätverk, både kommersiella och militära, tillåter individuella enheter att stödja varandra. Ledningscentralen för systemet ska vara utformad som ett rum innehållande olika former för datorstöd. Utrustningen ska stödja ett samarbete mellan personer från olika delar av organisationen samt från olika organisationer. Personerna i rummet ska kunna använda datorstödet vid samarbete för att koordinera resurser samt för att sprida information vidare till andra enheter. Ledningscentralen ska kunna användas i situationer som är tidskritiska, som kräver synkront arbete, som pågår över kortare eller längre tid eller som är mycket målfokuserade. Det var därför viktigt att den framtagna miljön var användbar och väl designad för detta ändamål.

I en situation liknande den ovan nämnda, är det önskvärt med en mycket säker informationshantering. Anledningen till detta är bland annat att underlätta och säkerställa att rätt beslut tas samt att rätt order utförs. Ett exempel är hur säker informationen upplevs vara, det vill säga om informationen kommer från rätt källa och är tillförlitlig. Ytterligare ett exempel är säkerheten i att rätt order verkligen har överlämnats och mottagits.

Det svenska försvaret använder i dag grafiska objekt och figurer med fördefinierad innebörd för att underlätta sin kommunikation. Objekten kan kombineras på olika sätt så att komplex och entydig information ska kunna utväxlas. Den miljö som har tagits fram i detta examensarbete, är tänkt att kunna användas för att studera hur den haptiska återkopplingen påverkar hur man upplever säkerheten i överlämnandet av denna typ av symboliserad information. Även faktorer som faktisk precision samt effektivitet kan studeras.

Eftersom miljön i första hand ska användas för att stödja överlämnande har störst vikt lagts vid design av objekten som ska överlämnas samt funktioner kring själva överlämnandet.

3.1 Generella designkriterier

Den framtagna miljön ska användas för samarbete. Det kräver att minst två personer kan interagera i miljön samtidigt. Eftersom arbetet kan vara tidskritisk är det mycket viktigt att miljön är intuitiv och användarvänlig. Ett önskemål är dessutom att miljön ska kunna anpassas efter olika situationer och därför bör miljön vara generell och ska kunna användas för flera olika scenarier. Miljön ska dessutom stödja ett säkert överlämnande. Det bör därför utvecklas en version av miljön som innehåller funktioner som bidrar till en säkrare hantering av objekten, en kontrollerad miljö. Denna miljö utvecklas parallellt med en okontrollerad, fri miljö. För att kunna studera hur den haptiska återkopplingen påverkar interaktionen behöver miljön finnas i en haptisk och en icke-haptisk version. De generella designkriterierna sammanfattas i tabell 2.

Tabell 2. Sammanfattning av generella designkriterier för miljön.

Generella designkriterier

- Den virtuella samarbetsmiljön ska fokusera på upplevd samt faktisk säkerhet i överlämnande.
 - Två personer ska kunna interagera samtidigt och/eller var för sig i miljön.
 - Den virtuella miljön ska vara intuitiv, generell och anpassningsbar.
 - Miljön ska skapas i två olika versioner, fri samt kontrollerad, och varje version ska representeras med haptiskt/visuellt gränssnitt samt visuellt gränssnitt.
 - Exempel på uppgifter som ska kunna utföras i miljön är: överlämna ett objekt, skapa ett sammansatt objekt av flera enkla objekt samt överlämna det sammansatta objektet.
-

3.1.1 Skillnad mellan fri och kontrollerad miljö

Den kontrollerade miljön utvecklas i syfte att ge användaren utökat stöd för objekt-hantering och överlämnande av objekt. Som tidigare nämnts är det till exempel i system liknande NetDefence, viktigt att information inte försvinner eller överlämnas till fel person. Den kontrollerade miljön bör därför ha egenskaper som säkerställer att informationen, i detta fall objekten, inte kan tappas bort. Genom att kontrollera när användaren kan ta tag i eller sätta ifrån sig objekt kan man till exempel se till att det alltid är någon som har ansvaret för informationen och att den inte försvinner.

3.2 Specifika designaspekter

Utöver de generella designkriterierna kan det vara nyttigt att framhäva några aspekter som är speciellt viktiga att ta hänsyn till.

3.2.1 Användbarhet

En användbar miljö kännetecknas av en hög grad av tillfredsställelse och ändamåls-enlighet som beskrevs i kapitel 2. Relaterat till säkerhet i överlämnandet kan de två begreppen översättas till upplevd säkerhet och faktisk säkerhet. Den upplevda säkerheten är ett subjektivt mått på hur säkert användaren upplever överlämnandet och om användaren är tillfredsställd med utförandet av uppgiften. Den faktiska säkerheten är ett objektivt mått på hur effektivt uppgiften är löst.

För att miljön ska upplevas som intuitiv är det viktigt att man faktiskt kan göra det man förväntar sig kunna göra i miljön. Dessutom är det viktigt att miljön är konsekvent och lätt att lära sig. En säker hantering av objekten kan stödjas genom att det läggs till funktioner som styr hur och när man kan interagera med objekten.

3.2.2 Samarbete

I ett samarbete sker det hela tiden en utvärdering av samarbetspartners interaktion. Resultatet av detta påverkar förtroendet i miljön samt hur man beter sig. Det är därför viktigt att båda användarna kontinuerligt får information om den andra personens aktivitet, den egna aktiviteten samt påverkan på miljön. Även val av kommunikationskanaler påverkar samarbetet.

3.2.3 Återkoppling

Interaktionen ska vara intuitiv och kännas effektiv. Den bör därför baseras på direktmanipulation. Det betyder att all interaktion ska vara synlig hela tiden. Ytterligare visuell återkoppling som till exempel färgmarkering av objekt i vissa tillstånd eller dylikt är inte aktuellt. Detta beror på att det är den haptiska återkopplingen som är det centrala. Av samma anledning ges det heller ingen specifik ljudåterkoppling. För övrigt ska det vara möjligt att känna interaktionen med alla ytor i miljön. Det är dessutom önskvärt att det i den kontrollerade miljön ges utökad haptisk återkoppling för att stödja objekthanteringen.

3.2.4 Funktioner

Det ska vara möjligt att ta tag i, hålla i, släppa samt flytta ett objekt. Man ska kunna kombinera flera enkla objekt och skapa ett sammansatt objekt och man ska kunna ta tag i, hålla i, släppa samt flytta det sammansatta objektet. Det ska vara möjligt att överlämna ett enkelt och/eller ett sammansatt objekt. Det är dessutom önskvärt att givaren ska kunna avbryta ett överlämnande genom att dra till sig objektet.

3.2.5 Arkitektur

Miljön är tänkt att fungera som ett virtuellt rum för formella möten. Miljön bör därför likna ett sammanträdesrum utan störande personliga föremål. Rummet ska begränsas av tak, golv och väggar. I rummet bör det finnas platser där man kan placera objekt. Objekten kan ha olika utseende och form. Det bör även finnas plats där användarna kan skapa sammansatta objekt. Varje användare bör ha en enkel grafisk representation som ger tillräcklig information om närvaro, aktivitet och placering.

Miljön är tänkt att vara generell och anpassningsbar och det bör därför vara möjligt att förändra rummets och objektens egenskaper. Förändringarna kan dock endast göras av applikationsutvecklaren och inte av användaren. Det kan av den anledning vara onödigt att ha en hög detaljnivå i miljön.

Kapitel 4 – Prototypkonstruktion

Att utveckla en prototyp är och bör vara en iterativ process. Robusta och väl designade system ska alltid kunna ändras beroende på nya krav eller tillgång på utrustning. Mindre ändringar bör kunna genomföras utan att hela systemet måste designas och implementeras om. Det är också ofta som ställda krav inte går att uppfylla inom ramen för tillgänglig utrustning, kompetens och tid. Det finns därför många möjliga källor till frustration och förseningar inom systemutvecklingsprocesser.

I utvecklingsprocessen för denna prototyp har jag genomfört följande steg:

- fastställt ursprungliga objektiva och subjektiva krav samt användarscenarier baserade på den övergripande designen
- analyserat de ursprungliga kraven och scenarierna
- utifrån analysen designat miljön med anseende på grafiskt användargränssnitt, haptiskt användargränssnitt samt funktioner i miljön
- implementerat designen i programspråken VRML, Python och C++, med hjälp av Reachin API
- testat att miljön hanterar ställda krav

Under pågående utveckling har enstaka krav fått ändras av olika anledningar. Svårigheter med att implementera det ursprungliga kravet eller önskvärda förenklingar i miljön har varit några orsaker. Implementeringen har därför gjorts iterativt. Det har dock hela tiden funnits en körbar och fungerande version av miljön. Testerna av objektiva krav har utförts manuellt parallellt med implementeringen. Som avslutning på arbetet utfördes en mindre användarstudie för att utvärdera miljöns användbarhet och funktionalitet. Studiens innehåll och genomförande beskrivs i kapitel 6. Detta kapitel beskriver innehållet i varje utvecklingsfas från ursprungliga krav vidare till analys samt design och implementering och slutligen test.

4.1 Ursprungliga krav

De generella designkriterierna som togs presenterades i den övergripande designen, kan specificeras och delas upp i följande objektiva och subjektiva krav:

Objektiva krav:

- Man ska kunna hålla i, lyfta och flytta ett objekt.
- Man ska kunna skapa sammansatta objekt av enkla objekt.
- Man ska kunna överlämna ett enskilt eller ett sammansatt objekt mellan två användare i miljön.
- Man ska kunna avbryta ett överlämnande.
- Man ska få kontinuerlig haptisk och visuell återkoppling.

Subjektiva krav:

- Interaktionen med och i miljön ska ske naturligt och så realistiskt som möjligt.
- Miljön ska vara användbar och intuitiv.
- Miljön ska stödja ett säkert överlämnande av objekt.

- Miljön ska stödja samarbete mellan två användare.

4.1.1 Användarscenarier

För att tydliggöra de objektiva kraven skapas användarscenarier. De kan fungera som en mall för framtida tester. När scenarierna kan genomföras på ett tillfredsställande sätt uppfyller applikationen de ställda kraven. Detta förutsätter dock att användarscenarierna är genomtänkta och täcker alla eventuella situationer som systemet ska klara av att hantera. I analysfasen kan man sedan förfinas scenarierna. De subjektiva kraven testas bäst i en användarstudie med noviser.

Nedan följer fyra exempel på viktiga scenarier som användes i prototyputvecklingen. Systemets användare betecknas med versalerna A respektive B, medan objekt betecknas med gemenerna a, b och c.

Scenario 1: Skapa och ändra ett sammansatt objekt

- A väljer ett objekt a och flyttar det.
- A väljer sedan ett annat objekt b och skapar ett sammansatt objekt av a och b.
- A tar bort b ur det sammansatta objektet. Det sammansatta objektet består nu av ett enkelt objekt a.

Scenario 2: Överlämna ett objekt

- A väljer ett objekt a och ger det till B.
- B tar emot a och placerar ner a.

Scenario 3: Skapa och överlämna samt ta i sär ett sammansatt objekt

- A skapar ett sammansatt objekt av objekten a, b och c.
- A ger det sammansatta objektet till B.
- B placerar ner det sammansatta objektet och placerar a, b och c på var sin plats.

Scenario 4: Avbryta ett överlämnande

- A ger ett objekt a till B.
- B tar emot objekt a, A håller fortfarande i objekt a.
- A ångrar överlämnandet och avbryter.

4.2 Analys

Analysfasen gör man för att få bättre förståelse för innehållet i kraven. Man försöker få en översikt över systemet och förbereder inför designen. Detta görs genom att man översätter den generella beskrivningen av kraven till mer systemspecifika termer. De objektiva kraven måste möjligen anpassas efter den utrustning som finns att tillgå och användarscenarierna måste förfinas. I analysfasen behandlades följande aspekter på prototypen: utrustning, miljöns utformning, återkoppling i miljön samt specificerade användarscenarier.

4.2.1 Utrustning

I tabell 3 redovisas en översikt över utrustningen som krävdes och fanns till förfogande för prototyputvecklingen.

Tabell 3. Översikt över utrustning som användes i utvecklingsarbetet.

Utrustning:	
Hårdvara	<ul style="list-style-type: none">• DELL Precision med 2 st. Pentium3 420 MHz• Diamond FireGL2 grafikkort• 2 st. Reachin Display (2a) med PHANTOM Desktop
Plattform	<ul style="list-style-type: none">• Windows 2000 Professional
Mjukvara	<ul style="list-style-type: none">• Reachin API 3.0• Borland C++ Builder 5.0

Reachin Display (2a)

Reachin Display (2a) består av en aluminiumram som är specialdesignad för att kunna samlokalisera handrörelse och synintryck, det vill säga haptik och grafik (se bild 3).

Ramen fungerar som hållare för en 17 tums bildskärm, en stereosändare från Stereographics samt en halvgenomskinlig spegelplatta. Bildskärmen har en upplösning på 1280 x 720 bildpunkter och en uppdateringsfrekvens på 120 Hz. Som positioneringsenhet används Magellan/SpaceMouse som ger sex frihetsgrader, samt PHANTOM Desktop som är den haptiska enheten. Magellan/SpaceMouse har inte använts i detta arbete.

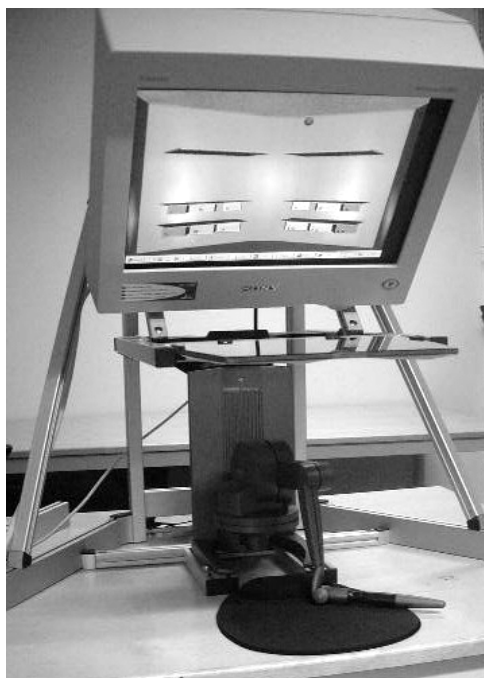


Bild 3. Reachin Display (2a).

Phantom Desktop

Den haptiska interaktionsenheten PHANTOM Desktop (i fortsättningen refererad till som *Phantomen*), har en positionsnoggrannhet på 0.02 mm. Arbetsvolymen på 16 x 13 x 13 cm är ett mått på markörens räckvidd i en och samma miljö. Enheten kan generera krafter upp till 6.4 Newton. Användaren interagerar med miljön genom att hålla i en penn-liknande robotarm (se bild 4). Denna ger positionering i sex frihetsgrader (translation i x, y- och z-led samt rotation kring x, y- och z-axeln) samt haptisk återkoppling i tre frihetsgrader (translation i x-, y- och z-led). Interaktionsenheten har även en knapp som kan användas för interaktion utöver positionering.



Bild 4. Interaktion med PHANTOM Desktop.

Reachin API

Reachin API är ett objektorienterad API (Application Programming Interface) skrivet i C++. Det tar hand om den haptiska renderingen och använder OpenGL för grafik-renderingen. På detta sätt integreras haptik och grafik i ett enda scenograf-API⁶. APIet tar även hand om stereografik och kommunikationen mellan den haptiska enheten och datorn.

Haptiken kräver som tidigare nämnts en uppdateringsfrekvens på 1000 Hz för att kunna återge en naturlig känsla. Sjunger uppdateringsfrekvensen kan till exempel ytor kännas mjukare än det var tänkt. Det betyder att den haptiska enheten uppdateras för sent. Man hinner trycka för långt in i objektet innan den reagerar med att generera en motkraft så att man trycks ut ur objektet.

Uppdateringsloopen på 1000 Hz kallas realtidsloopen. APIet använder även en grafisk loop med uppdateringsfrekvens 60 Hz. Det betyder att ögat behöver endast 60 bilder i sekunden för att uppfatta rörelsen som sammanhängande. Realtidsloopen tar alltså hand om den haptiska renderingen och körs på en av de två processorerna. Den andra processorn kör grafikloopen, operativsystemet o.dyl.

⁶ En dokumentation av Reachin API-noder finns på <http://www.reachin.se/support/documents/referenceManual/> (december 2001)

Reachin API använder VRML som scengrafmall, men följer inte VRML-standaren fullt ut. För applikationsutveckling med hjälp av Reachin API, krävs programmeringskunskaper i VRML, Python samt C++.

Utrustningens begränsningar

Virtuella samarbetsmiljöer förutsätter ofta att användarna befinner sig på olika platser. Det finns för närvarande ingen nätverksstandard för överföring av haptisk information. De båda haptiska enheterna har därför seriekopplas till en gemensam dator. För närvarande är det maximala avståndet mellan de två enheterna 1,5 m då det uppstod kraftiga störningar vid längre kabel.

Eftersom applikationen körs på en dator krävs en bildskärmsdelare. Det resulterar i att båda användarna ser samma grafiska representation av miljön. Man har därmed ingen möjlighet att ge varje användare var sin vy, till exempel en första person-vy.

Phantomen har endast en knapp som kan användas för interaktionen, till skillnad från en vanlig datormus som ofta har tre knappar. Det reducerar antalet möjliga funktioner. Phantomens arbetsvolym begränsar dessutom miljöns storlek om man vill ha en viss detaljnivå. Den maximala kraften som enheten kan generera betyder mycket för vilka haptiska egenskaper som kan implementeras i applikationen. Detta märks speciellt om man vill lyfta tunga objekt. Phantomen erbjuder endast en interaktionspunkt och det styr hur väl man kan utforma interaktionen i miljöerna.

Om man vill använda Reachin API för applikationsutveckling begränsas implementeringen till programspråken VRML, Python och C++.

4.2.2 Miljöns utformning

Att överlämna objekt kan göras i många olika sammanhang. Omgivningens utformning kan därför bero av den kontext man vill uppnå i miljön. Om objekten till exempel ska symbolisera dokument, kan omgivningen återges som en mer formell, kontorsliknande miljö. Symboliserar objekten leksaker kan miljön utformas mer personligt. För att utveckla en generell miljö bör man därför skapa en omgivning som inte innehåller onödiga föremål eller störande objekt.

Arbetets ursprungliga idé om säkert överlämnande av objekt, har arbete i en formell kontext som utgångspunkt, nämligen ledningscentraler. Ett rum med några hyllor eller bord att placera objekten på kan därför skapa tillräckligt med sammanhang eller kontext och samtidigt vara generell.

Objekten ska kunna symbolisera vad som helst. De bör därför ha en enkel utformning med möjlighet till att ändra visuella och haptiska egenskaper som form, färg, struktur, vikt o.dyl.

För att skapa en realistisk och intuitiv miljö bör objekten och hyllorna utformas så att man ser vad de föreställer. Till exempel bör hyllorna placeras intill väggen och inte mitt i rummet så länge de inte ska fungera som en rumsavdelare.

Syftet med utvecklingen av prototypen var bland annat att skapa en miljö där man kunde överlämna objekt. Ett krav var att miljön skulle stödja ett säkert överlämnande. För att studera hur miljöns funktioner och regler kan stödja den upplevda samt den faktiska säkerheten i överlämnandet, bör två versioner av miljön skapas. Den fria miljön försöker i högsta grad eftersträva en naturlig och intuitiv interaktion med objektet. Den kontrollerade versionen bör däremot innehålla funktioner eller regler som underlättar hanteringen och stödjer ett säkert överlämnande av objektet.

4.2.3 Interaktion och återkoppling

I dagliga livet kan man utforska ett objekt på många olika sätt. Lederman och Klatzky har fastställt ett antal utforskningsprinciper (eng. exploration principles, EP) som beskriver hur en människa går till väga för att undersöka olika egenskaper hos ett objekt [28]. Vill man undersöka tyngden "väger" man objektet i handen, vill man undersöka strukturen stryker man fingrarna längs ytan (taktil perception). När man vill lyfta ett objekt tar man oftast tag i objektet med två eller flera fingrar eller med två händer om objektet är stort. Det är dels för att balansera objektet, men också för att få ett intryck av objektets storlek och position i förhållandet till kroppen. Man kan till exempel känna att tummen och pekfingret befinner sig på ett visst avstånd ifrån varandra och därmed bedöma objektets tjocklek.

PHANTOM Desktop erbjuder som tidigare nämnts endast en interaktionspunkt. Det betyder att man inte kan greppa om ett objekt för att lyfta det. Idag är dock de allra flesta vana med "drag-and-drop"-metaforen man använder för interaktion i grafiska gränssnitt. Att använda knappen på PHANTOM Desktop borde därför vara tillräckligt intuitivt för att simulera att greppa.

För att kunna ta tag i ett objekt måste man självklart beröra objektet om miljön ska vara någorlunda realistisk. Frågan återstår om det ska vara möjligt att lyfta ett objekt oavsett var på objektet man greppar. En första ide var att sätta handtag på objekten för att tydliggöra var man var tvungen att beröra objektet för att greppa det. Efter diskussioner med min handledare bestämde vi dock att ta bort handtagen då det gör interaktionen med objekten mindre intuitiv och därmed blir också miljön mindre användbar.

En annan frågeställning som kom upp var att om man placerar flera objekt ovanpå varandra, hur ska man då kunna flytta alla objekt på en gång? Från början fanns planer på att ha en funktion som skapade sammansatta objekt och som satt på ett gemensamt handtag för hela objektet. Men detta bidrar till en mer komplicerad miljö och mindre intuitiv interaktion. När man i verkliga livet placerar klossar ovanpå varandra måste man ta tag i klossen längst ner för att lyfta med sig samtliga klossar. Detta användes sedan som mall för hur man flyttar sammansatta objekt i miljön.

För att objekten ska vara realistiska måste man ta hänsyn till gravitationen och kollisiondetektion. Det betyder att objekten måste falla till marken eftersom de inte ska sväva. Vidare måste objekten kollidera med varandra samt med hyllor och väggar, det vill säga inte kunna gå igenom varandra eller väggar.

Vid ett överlämnande måste båda användarna känna sin egen och den andras påverkan på ett gemensamt objekt. Det vill säga att tyngden fördelas mellan de två samt

att om en användare flyttar objektet ska den andra följa med, om den andra fortfarande håller i objektet. För att ångra ett överlämnande i verkligheten kan man rycka det åt sig. På samma sätt borde det fungera i miljön. Man kan dock tänka på hur mycket det ska vara möjligt att dra åt sig, det vill säga hur stora krafter man kan få använda. PHANTOM Desktop har här en mekanisk begränsning på 6.4 Newton vilket för en vuxen människa är en relativt liten kraft.

Ett krav för miljön är att man använder direktmanipulation. Det vill säga att man hela tiden ser vad man gör. Det betyder att man också hela tiden ska känna det man interagerar med. Alla objekt ska därmed vara haptiska.

Användaren kan representeras av en enkel geometrisk form (avatar). Enligt tidigare studier har det visat sig att en sfär ger användaren tillräckligt med information om egen och andras närvaro, aktivitet och identitet [42]. Eftersom PHANTOM Desktop endast ger kraftåterkoppling i 3 frihetsgrader (translation i x-, y- och z-riktning, ingen rotation) kan det vara en fördel att avataren är symmetrisk med hänsyn till rotationen. Annars kan användaren tro och förvänta sig att rotationskrafter kan ges.

För att kunna studera effekten av haptisk återkoppling måste dessutom både den fria och den bundna miljön finnas i en icke-haptisk version. Eftersom vi vill studera effekten av den haptiska återkopplingen specifikt önskas inga övriga återkopplingar som ljud eller utökad visuella markeringar.

4.2.4 Specificerade scenarier

De fyra scenarierna som presenterades i kapitel 4.1.2 blev senare förfinade och mer detaljerade. Varje del av respektive scenario specificerades genom att beskriva de funktioner som användes, handling som krävdes för att utföra funktionerna samt vilken haptisk återkoppling som skulle ges (se Scenario 1-Scenario 4).

Scenario 1: Skapa och ändra ett sammansatt objekt

Funktion	Handling	Haptisk återkoppling
Ta tag i	A berör a, trycker ner och håller nere knappen	Känner a:s tyngd
Flytta	A håller a, flyttar a till valfri plats	Känner kollision mellan a och hyllan/bordet
Släppa	A släpper knappen	Tyngd försvinner
Skapa sammansatta objekt	A tar tag i b, flyttar b, placerar b ovanpå a, släpper b	Känner b:s tyngd samt kollision mellan a och b
Ändra det sammansatta objektet	A tar tag i b, flyttar b, placerar b på valfri plats	Känner b:s tyngd samt kollision mellan b och hyllan/bordet

Scenario 2: Överlämna ett objekt

Funktion	Handling	Haptisk återkoppling
Ge objekt	A tar tag i a, håller	A känner a:s tyngd
	B tar tag i a, håller	A och B känner och delar a:s tyngd, känner varandras förflyttningar
Ta emot objekt	A släpper a	A känner ingen tyngd, B känner hela a:s tyngd
	B flyttar a till valfri plats	B känner kollision mellan a och hylla/bord

Scenario 3: Skapa och överlämna samt ta i sär ett sammansatt objekt

Funktion	Handling	Haptisk återkoppling
Skapa sammansatt objekt	A placerar a, b och c ovanpå varandra	A känner alla objekts tyngd samt kollision mellan objekten
Ge sammansatt objekt	A tar tag i det understa objektet, a, håller	A känner sammanlagda tyngden av alla objektens tyngd
	B tar tag i a, håller	A och B känner och delar det sammansatta objektets tyngd
Ta emot sammansatt objekt	A släpper a och B flyttar det sammansatta objektet till valfri plats	B känner hela det sammansatta objektets tyngd, känner kollision mellan objektet och hylla/bord
Ta i sär ett sammansatt objekt	B tar tag i c, respektive b och a och flyttar de till var sin plats	B känner alla objektens tyngd samt kollision mellan objekt och hylla/bord

Scenario 4: Avbryta ett överlämnande

Funktion	Handling	Haptisk återkoppling
Ge ett objekt	A ger a till B	A och B känner och delar a
Ångra	A ångrar, drar åt sig a med en viss kraft	A känner B:s interaktion med objektet
	B tappar greppet om a	A känner hela a:s tyngd

4.3 Design

Syftet med designfasen är att förbereda för implementeringen. Man försöker dela in systemet i delsystem och specificerar delsystemens funktioner. I ett traditionellt systemutvecklingsprojekt är det vanligt att rita diagram över systemets arkitektur. Man kan till exempel rita upp alla delsystem och markera hur de kommunicerar med varandra samt hur de är uppbyggda. I detta arbete har designfasen även bestått av att specificera den grafiska och haptiska utformningen av miljön.

4.3.1 Systemarkitektur

Reachin API är som tidigare nämnts uppbyggt efter scenografprincipen. Man kan använda APIets innebyggda noder men även skapa nya noder och använda dessa i egna applikationer. Miljön kan delas upp i olika delsystem. Man har till exempel objekten, hyllor/bord som objekten kan placeras på samt själva rummet. Den framtagna prototypen bör därför innehålla tre nya noder eller delsystem: objekt, hyllor och rum. Dessa noder kan skapas genom att redan existerande noder byggs på med nya eller modifierade egenskaper och funktioner. För att skapa nya eller utöka noderna krävs programmering i C++.

4.3.2 Grafiskt användargränssnitt

Interaktionen bygger som tidigare beskrivet på direktmanipulation. Det är även ett mål att interaktionen ska vara så realistisk som möjligt. Det är därför viktigt att både den visuella och haptiska återkopplingen utformas med detta i åtanke. Det betyder att alla kollisioner mellan alla fysiska föremål måste återges visuellt samt att objekten måste följa gravitationens lagar.

Objekten

Det bör finnas ett antal objekt med olika egenskaper som till exempel färg, storlek och textur. Det bör dessutom vara möjligt att skilja de olika objekten åt med endast visuell återkoppling. Önskvärda former är boxar, cylindrar samt koner. Det bör vara möjligt att enkelt ändra de olika egenskaperna. Objekten placeras ut på hyllorna och måste därför ha en position från början.

En begränsning som bör göras redan i designfasen är att objekten inte kan rotera. Det betyder en reducering av miljöns realism men underlättar kraftigt för kollisionsdetektionen. Traditionellt sett är kollisionsdetektion ett mycket beräkningskrävande problem i datorgrafik. Det finns speciella program som är gjorda för att beräkna fullkroppskollisionsdetektion, till exempel Karma från Math Engine⁷. Detta finns dock inte inbyggt i Reachin API.

Hyllorna

Hyllorna bör ha visuella egenskaper som storlek, färg och textur. Det bör dessutom vara möjligt att ändra dessa egenskaper. Hyllorna placeras intill väggarna och bör på samma sätt som objekten ha en position. Eftersom hyllorna är statiska kommer de att ha samma position hela tiden.

Rummet

Rummet bör vara symmetriskt kring yz-planet, det vill säga spegla högra och vänstra sidan. Detta görs för att de två användarna ska ha "var sin" likadan sida. (Se till exempel bild 5 eller bild 6.) Rummet begränsas av en bakre vägg, två sneda sidoväggar, två raka sidoväggar samt en genomskinlig främre vägg. Den främre väggen görs genomskinlig för att man ska kunna se interaktionen. Anledningen till att den ritas ut över huvudet är att användaren ska kunna känna att det är en vägg där, samt att objekten ska kunna studsas mot väggen. Även taket ritas ut genomskinligt för att ge

⁷ För mer information se <http://www.mathengine.com>

rummet en öppnare känsla. Rummets djup och bredd måste anpassas efter Phantomens räckvidd som är 16 cm i x-led samt 13 cm i både y- och z-led.

Både golv och väggar bör ha en textur för att göra rummet mindre sterilt. Hyllorna och borden placeras intill de sneda sidväggarna. Vinkeln mellan den bakre väggen och den sneda sidväggen görs större än 90 grader för att man lättare ska kunna se hyllornas innehåll.

Avatarerna

Avatarerna utformas som sfärer. Radien på sfären får inte vara större än objektens höjd, bredd eller djup. Om så inte är fallet kan användaren inte greppa eller utforska objektet på lämpligt sätt.

Färger

Genom att ge objekt och avatrar liknande färgnyanser ämnar man påverka användaren till att se på hyllorna eller objekten som "sina". De två avatarerna bör dock ha en lätt åtskiljbar färg. Blå och grön är ett bra val eftersom man då tar hänsyn till användare med röd/grön färgblindhet. Hälften av objekten ges därmed blånyanserade färger medan resterande objekt ges grönyanserade färger.

4.3.3 Haptiskt användargränssnitt

På samma sätt som med det grafiska användargränssnittet, är det viktigt att interaktionen känns kontinuerlig. Det betyder att alla rummets fysiska föremål, det vill säga väggar, golv, tak, hyllor, objekt och avatrar, ska ha en yta som användaren kan känna.

Rummet

Väggarna och golvet bör ha en viss styvhet och friktion. Väggarna bör dock vara styvare än golvet, då golvet kan bestå av till exempel en matta. Taket ska vara utan friktion och men ha en viss styvhet.

Hyllorna

Hyllorna bör ha en stor friktion för att ge användaren en uppfattning av att objekten inte kan halka av, och de ska vara ganska styva.

Objekten

Objekten bör ha olika haptiska egenskaper som till exempel ytstruktur, styvhet och tyngd.

Delade objekt ska kännas delade, det vill säga man ska känna en annan användares påverkan på objektet. Till exempel att tyngden delas mellan två användare som håller ett objekt eller att en användare dras med om den andra flyttar objektet och båda håller i det.

4.3.4 Fri respektive kontrollerad miljö

Två versioner av miljön, en fri och en kontrollerad, har utvecklats. I den kontrollerade miljön finns det vissa begränsningar på hur objekten kan bete sig. Det var dessutom önskvärt att ytterligare haptisk återkoppling lades till i miljön. Det grafiska gränssnittet är likadant i de båda miljöerna.

Funktioner

I den kontrollerade miljön kan objekt endast greppas och släppas om de är placerade på en hylla, ovanpå ett annat objekt eller när en annan användare håller i objektet. Med andra ord kan de inte släppas i lösa luften eller placeras på golvet. Anledningen till att de inte får placeras på golvet är att de då kan ses som borttappade, vilket inte får ske i den kontrollerade miljön. Det ska heller inte vara möjligt att skjuta ner objekt från hyllorna.

Återkoppling

Objekten ska i den kontrollerade miljön vara magnetiska för att underlätta för användaren att lokalisera dem. Det är dessutom önskvärt att användaren känner ett "klick" eller en liten dragningskraft in mot objektet när det greppas.

När ett objekt placeras ovanpå ett annat ska det kännas magnetiskt genom att det centreras över mittpunkten till objektet under. Detta görs för att ge användaren ytterligare återkoppling samt försäkra användaren om att objektet är placerat.

4.4 Implementering

Implementeringen gjordes som tidigare nämnts i programspråken VRML, Python och C++ med hjälp av Reachin API. Alla tre programspråken var för mig ovana språk och jag valde därför att begränsa programmeringen så långt det var möjligt till VRML och C++.

Prioriterad ordning för implementeringen var att i en första omgång implementera en fungerande fri miljö, det vill säga att man kunde

- ta tag i ett objekt,
- släppa ett objekt,
- bygga sammansatta objekt,
- lyfta sammansatt objekt samt
- överlämna enskilt eller sammansatt objekt.

Nästa steg var att implementera den kontrollerade miljön, det vill säga

- magnetiska objekt,
- funktioner så att det endast var möjligt att ta tag i och släppa objektet som stod i hyllan, ovanpå annat objekt eller på bordet eller som någon annan höll,
- magnetism mellan objekten när sammansatta objekt skapades samt
- göra det omöjligt att skjuta ner objekt från hyllorna.

De nya noderna skapades med C++-kod (se Bilaga 2). Modelleringen av alla föremåls grafiska och haptiska egenskaper gjordes med relativt enkel VRML-kod (se Bilaga 1).

4.4.1 Datastrukturer

För att kunna flytta objekt var det nödvändigt att objekten var möjliga att greppa. I en första implementering användes därför noden GraspableTransform. Denna nod innebar att objekt kunde greppas med hjälp av en knapptryckning. Nackdelen var dock att objekt skapade med denna nod inte kunde påverkas av krafter eller röra sig själva. Det blev därför naturligt att välja noden Dynamic för att modellera objekten.

Greppfunktionen skapades med hjälp av ett virtuellt gummiband. När avataren berörde objektets yta och användaren samtidigt tryckte ner knappen, skapades ett gummiband mellan avataren och objektets origo. Genom att öka gummibandets styvhet kändes detta som om man greppade objektet och kunde hålla det stabilt. När knappen släpptes, togs gummibandet bort och därmed släpptes objektet. För varje användare som höll i objektet skapades det ett eget gummiband.

Enligt designkriterierna skulle det endast vara möjligt att lyfta ett sammansatt objekt genom att ta tag i det understa objektet. För att hålla ordning på vilka objekt som var ovanpå varandra hade varje objekt två variabler som innehöll en pekare till det eventuella objektet under respektive ovanpå. När man rörde vid ett objekt uppdaterades en vektor som innehöll alla objekt ovanpå det aktuella objektet. Vid förflyttning av ett sammansatt objekt blev objekten ovanpå förflyttade i förhållande till det greppade objektets position.

Den kontrollerade miljön skapades för att stödja en säker objekthantering och ett säkert överlämnande. Det ställdes till exempel specifika krav på när man kunde ta tag i och släppa ett objekt, det vill säga när man kunde skapa och ta bort gummibandet. Genom att lägga in en variabel i objektet som kontrollerade om det var placerat på en hylla, på golvet eller ovanpå ett annat objekt kunde detta styras. Objekten hade även två variabler som ändrades beroende på om objekten inte hölls av någon eller hölls av en respektive två användare.

För att stödja objekthanteringen var det i den kontrollerade miljön möjligt att hålla objektet utan att ha knappen intryckt. Ville man släppa objektet var man tvungen att ge det till den andra användaren eller placera det på en hylla eller ovanpå ett annat objekt.

Den haptiska loopen måste uppdateras med en frekvens på 1000 gånger i sekunden (1 kHz) för att det ska kännas realistiskt. Den grafiska loopens däremot behöver endast en frekvens på 60 Hz. Det betyder att alla egenskaper som hade att göra med den haptiska återkopplingen placerades i realtidsloopen. Det gällde bland annat beräkningen och genereringen av alla krafter samt kollisiondetektionen.

4.4.2 Grafiskt användargränssnitt

Rummets väggar samt golv och tak modellerades helt och hållet med hjälp av noder som fanns i Reachin API. Väggarna fick dock ingen textur då det blev tillräckligt mjukt intryck med färg och rätt belysning.

Hyllorna bestod i första versionen av ett hyllplan med sidoväggar samt tre skiljeväggar. Bordet modellerades som ett bord med fyra ben, ett i varje hörn, samt med en underhylla (se bild 5). För att underlätta kollisionsdetektionen valde jag i samråd med handledare att förenkla hyllorna och bordets utseende. Både bordet och hyllorna blev därför reducerade till ett hyllplan. Enda skillnaden var att bordet blev djupare än hyllorna (se bild 6).

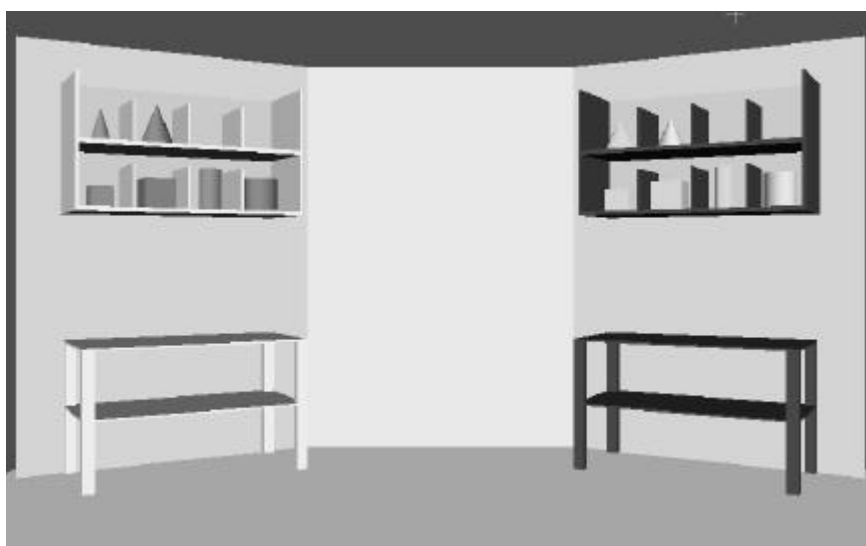


Bild 5. Version 1.

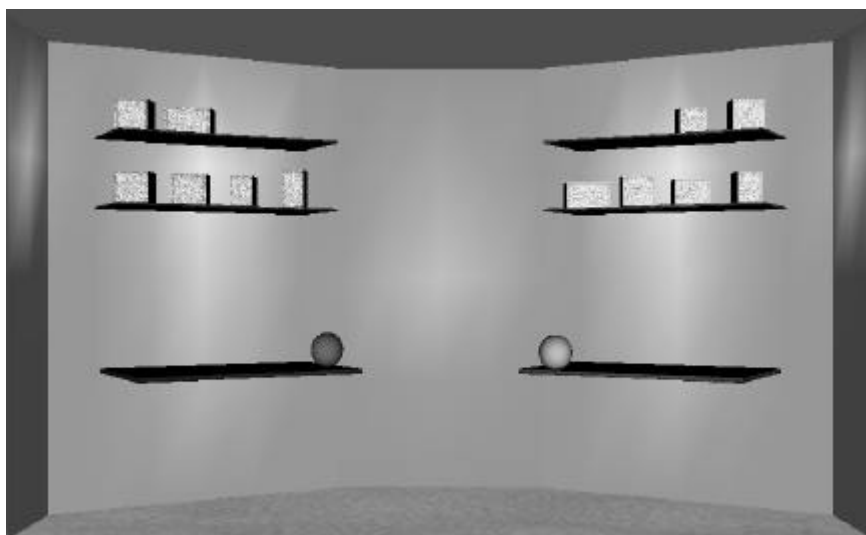


Bild 6. Version 2.

Det var till en början önskvärt att objekten hade olika geometriska former, till exempel kon, cylinder och box. Återigen var det för att underlätta kollisionsdetektionen som alla objekt modellerades som boxar. Traditionellt är det vanligt att använda sfärer som förenklade kollisionsobjekt då de har samma avstånd i alla riktningar. I den model-

lerade miljön fanns det dock sneda väggar och det blev därför enklast med boxar. Man kunde då använda boxens hörn för att detektera om boxen kolliderat med de sneda väggarna.

Det blev redan i designen bestämt att objekten inte skulle kunna rotera. Om man använder sfärer är det visuellt inget större problem, men att boxar inte kan rotera skapar en mindre realistisk miljö. Det innebar bland annat att boxar som stod med endast en liten del av hörnet på en hylla, inte tippade även om objektets tyngdpunkt låg utanför hyllan.

De två avatarerna modellerades som sfärer i var sin färg, blått respektive grönt. Färgtonerna återspeglades i objektens färger. Objekt på vänstra sidan hade tre olika blåfärger och objekt på högra sidan hade tre olika gröna färger. Objekten hade dessutom tre olika storlekar på varje sida. Avatarernas storlek gjordes mindre än boxarnas sidor så att det inte skulle bli något problem med att beröra ytorna. För att underlätta identifiering av och referering till objekten fick varje objekt en textur med en siffra (se bild 8).

4.4.3 Haptiskt användargränssnitt

Alla ytor gjordes kännbara genom att de gavs en yta när de modellerades i VRML-koden. Det var dock endast objekten som modellerades med en haptisk bumpmap-textur. De olika texturerna gjordes ganska lika för att man i användarstudien skulle kunna undersöka om det var möjligt att känna så små skillnader. Ett första förslag var att göra upphöjda trianglar, fyrkanter o.dyl. på boxarnas sidor. Detta gick dock inte då avatarerna var allt för stora jämfört med objektens storlek och höjden på bumpmap-texturen.

Objekten modellerades med lite olika styvheter. Detta märktes dock inte så väl på grund av storleksförhållandet mellan avatar och objekt. Det återspeglades dock i den känslan man fick när ett objekt greppades. Den genererade kraften från Phantomen beräknas utifrån hur långt in i objektet man har tryckt. Ju mjukare objekt ju längre in i objektet kan man trycka. När man greppade ett objekt placerades avataren på utsidan av objektets yta oavsett hur långt in i objektet man hade tryckt. Det fick som konsekvens att ju styvare objektet var desto starkare kraft kände man när man greppade ett objekt. Det var möjligt att skala bort denna kraft eller stänga av den haptiska ytan, men det prioriterades bort i den implementerade prototypen.

Ett objekts tyngd kan lättare bestämmas om man ger användaren haptisk återkoppling än med endast visuell information. Objekten fick därför olika vikter oberoende av färg och storlek. För att simulera gravitation lades det på en konstant kraft neråt för alla objekt. För att det skulle se realistiskt ut visuellt blev denna kraft relativt stor och skalades därför bort när objekten stod ovanpå ett annat objekt, på golvet eller på en hylla.

Genom tester visade det sig att den simulerade gravitationen medförde att objekten blev för tunga att lyfta. Den totala kraften neråt blev större än vad Phantomen kunde återge. Av den anledningen skalades objektets vikt ner när de greppades och lyftes. Det innebar också att om användaren balanserade ett objekt ovanpå sin avatar utan att

greppa det (knappen inte intryckt), kändes objektet tyngre än om det hade varit greppat (knappen intryckt).

I designen var det önskvärt med magnetiska objekt, dels för att hjälpa användaren att hitta objekten och dels för att göra det lättare att stanna kvar vid objektet. Reachin API innehåller både en magnet-nod och en nod för magnetiska ytor. Dessa kunde dock inte användas. Magnet-noden påverkade båda Phantomerna, det vill säga interaktionsenheterna, i miljön. Det innebar att om en användare närmade sig ett magnetiskt objekt skulle båda användarna dras till objektet. Det var dock möjligt att delvis komma undan detta problem genom att reducera magnetens styrka. Noden för den magnetiska ytan skapade en yta som var magnetisk endast när man hade berört objektet en gång. Den gjorde det alltså inte lättare att lokalisera objektet.

Vid byggande av sammansatta objekt kunde man dock simulera magnetism. Genom att placera det greppade objektets mittpunkt rakt ovanför det undre objektets mittpunkt så fort de två objekten hade kontakt i horisontellt led (y-led), kändes det som om objekten var magnetiska i förhållande till varandra. Denna känsla blev starkare ju större objekten var och därför gjordes objekten något större, jämför till exempel version 2 med version 3 i bild 7 respektive 8.

Användarna gavs möjlighet att känna på varandras avatarrer. Denna känsla skapades genom att generera en kraft som stötte bort avatarrerna när de närmade sig varandra. Det var nödvändigt att göra på detta sätt eftersom användarens representation i miljön inte kan ha en haptisk yta.

4.4.4 Svårigheter

Under implementeringen dök det upp ett antal problem av olika storlek och betydelse. De flesta problemen var relaterade till att objekten var dynamiska det vill säga kunde röra på sig, samt att det var två användare (Phantomer) i miljön.

Som tidigare nämnts måste man programmera kollisionsdetektion om man arbetar med objekt i rörelse. För att objekten ska uppvisa ett realistiskt beteende måste man se till att de studsar mot väggar och varandra. Kollisionsdetektion är en välkänd utmaning inom datorgrafik och mycket forskning läggs ner på området. För att underlätta beräkningarna och programmeringen ändrades modelleringen av hyllor och objekt.

Att använda två Phantomer i samma miljö skapade också några problem. Det första var att hårdvaran/mjukvaran inte klarade av att de olika enheterna var seriekopplade på ett längre avstånd. Det reducerade alltså möjligheterna till att ha två användare i vart sitt rum.

Ett annat problem kopplat till Reachin API var att noder i miljön påverkade båda de två Phantomerna, till exempel Magnet-noden. Om man till exempel vill stänga av den haptiska ytan för en användare när denna greppar ett objekt kommer heller inte den andra användaren att kunna känna någon yta. Några av dessa problem kan reduceras eller undvikas genom att ändra eller utöka noderna. Detta prioriterades dock bort.



Bild 7. Version 2.

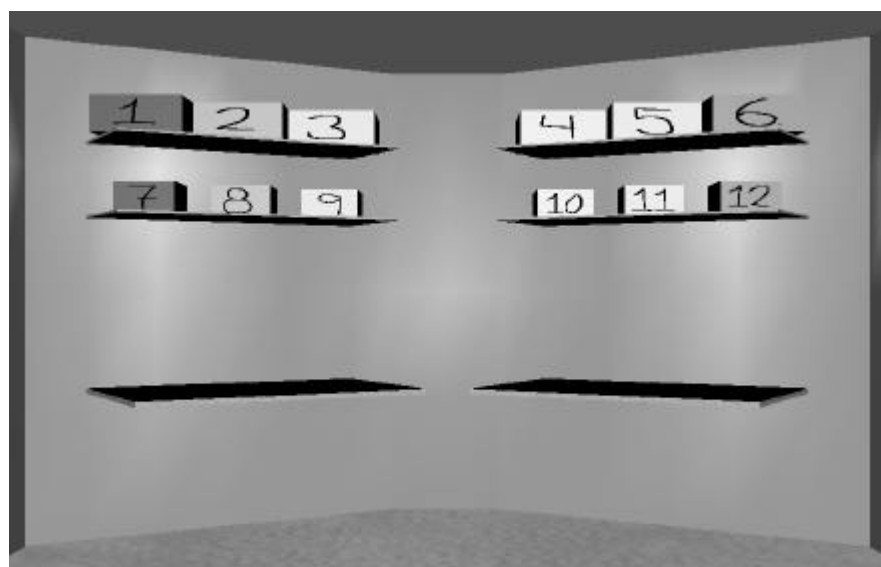


Bild 8. Version 3.

4.5 Test

Testerna utfördes manuellt under implementeringen. Det vill säga att jag testade fysiskt med Phantomen om den implementerade funktionen kunde utföras eller om den rätta återkopplingen gavs. Av denna anledning fanns det alltid en fungerande version av miljön. Bild 9 visar exempel på en överlämningsituation i miljön.



Bild 9. Interaktion i miljön: Överlämna ett sammansatt objekt.

Kapitel 5 – Resultat av prototyputvecklingen

Den implementerade prototypen finns i två versioner, en fri och en kontrollerad. De båda versionerna har samma grafiska användargränssnitt. De funktionella skillnaderna mellan miljöerna redovisas i nästa stycke. Det finns för närvarande ingen fungerande icke-haptisk version av miljön. Prototypen innehåller tre nya noder. Dessa noder är dels en utökning av befintliga noder i Reachin API och dels nyutvecklade noder.

5.1 Användargränssnittet

Både det grafiska och det haptiska användargränssnittet bygger på att interaktionen ska kännas och upplevas som naturlig. Interaktionen har modellerats enligt direktmanipuleringsprincipen som är grundläggande för grafiska gränssnitt. Den förutsätter att alla objekt återges kontinuerligt och tillåter fysisk interaktion samt att användaren får omedelbar återkoppling på sina egna och andra användares handlingar. Objektens grafiska och haptiska egenskaper redovisas i tabell 4.

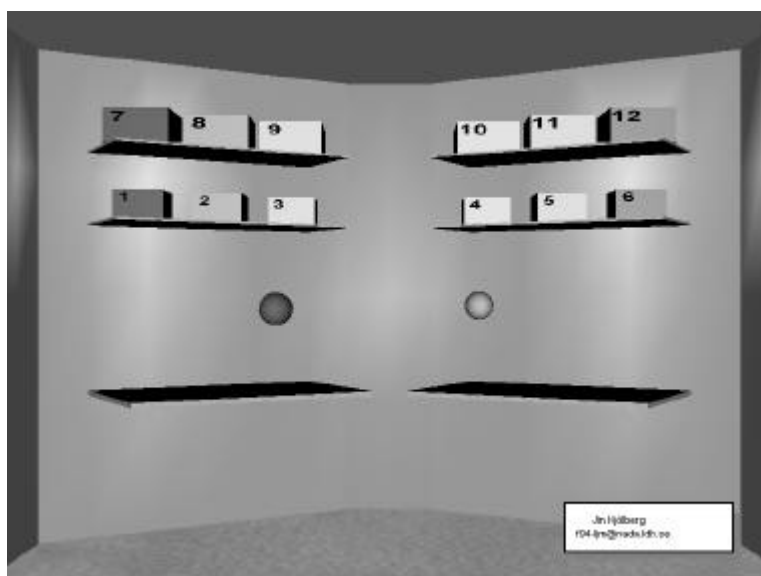


























Bild 10. Prototypens användargränssnitt.

5.1.1 Grafiskt användargränssnitt

Det grafiska användargränssnittet modellerades med VRML-kod (se bilaga 1) och består av följande komponenter (se bild 10):

- Ett symmetriskt rum avgränsat av fem väggar, ett golv samt en genomskinlig vägg och genomskinligt tak.
- Sex hyllor varav två något djupare än de övriga.
- Tolv box-formade objekt i två olika storlekar och sex olika färgnyanser.
- Två sfär-formade avatarer med var sin färg.

Tabell 4. Objekttegenskaper.

Visuell textur	Haptisk textur	Storlek [cm]	Färg [RGB]	Vikt [kg]	Styvhet [N/m]
		20x12x20	0,35-0,05-1,00 (mörkblå)	0,2	100
		20x12x20	0,25-0,50-1,00 (mellanblå)	0,3	300
		20x12x20	0,05-0,75-1,00 (ljusblå)	0,2	100
		20x12x20	0,25-1,00-0,50 (ljusgrön)	1,0	500
		20x12x20	0,35-1,00-0,50 (mellangrön)	0,5	100
		20x12x20	0,10-0,50-0,50 (mörkgrön)	0,3	100
		26x14x26	0,35-0,05-1,00 (mörkblå)	0,5	200
		26x14x26	0,25-0,50-1,00 (mellanblå)	0,2	100
		26x14x26	0,05-0,75-1,00 (ljusblå)	1,0	500
		26x14x26	0,25-1,00-0,50 (ljusgrön)	0,5	600
		26x14x26	0,35-1,00-0,05 (mellangrön)	0,2	500
		26x14x26	0,10-0,50-0,10 (mörkgrön)	0,4	100

5.1.2 Haptiskt användargränssnitt

Det haptiska användargränssnittet specificerades i VRML-filerna (se bilaga 1) och bidrar med följande egenskaper:

- Alla objekt och föremål har kännbara ytor.
- Objekten har egenskaper som vikt, ytstruktur och styvhet.
- Avatarerna påverkar varandra med krafter som gör att användarna kan känna på den andres avatar.

5.2 Funktioner

Följande funktioner implementerades både i den fria och den kontrollerade miljön:

- Objekten följer i princip de fysiska lagarna med undantag för rotation.
- Objekt krockar med varandra samt övriga föremål i miljön.
- Alla objekt kan greppas, hållas, släppas och flyttas.
- Sammansatta objekt kan byggas genom att flera objekt placeras ovanpå varandra.
- Enkla och sammansatta objekt kan överlämnas från en användare till en annan.

5.3 Skillnader mellan den fria och den kontrollerade miljön

De två versionerna av miljön, fri och kontrollerad, har samma grafiska användargränssnitt. Det är endast några haptiska egenskaper samt funktioner som skiljer. Man väljer vilken version man vill använda genom att sätta en variabel "functional" i CVE.cpp-filen till "true" respektive "false" innan man kompilerar koden och kör programmet.

De funktionella skillnader i de två implementerade versionerna redovisas i tabellform i tabell 5.

Tabell 5. Skillnader mellan fri och kontrollerad miljö.

Aktivitet	Kontrollerad miljö	Fri miljö
Skapa sammansatt objekt	Det greppade objektets mittpunkt placeras automatiskt rakt ovanför det understa objektets mittpunkt så fort objektens horisontella ytor har kontakt (y-led).	Det greppade objektet placeras där det släpps, men om man berör ett objekt kommer eventuella ovanstående objekt att flyttas så att de har sin mittpunkt ovanför det berörda objektets mittpunkt (x- och z-led).
Ta tag i eller släppa objekt	Kan endast göra detta <ul style="list-style-type: none">• om objektet står på en hylla (objektets mittpunkt måste vara på hyllan),• om objektet är placerat ovanpå ett annat objekt eller• om objektet hålls av en annan användare.	Kan göra detta när som helst.
Putta objekt ner från hyllan	Ska inte vara möjligt.	Kan göra detta med alla objekt.

5.4 Datastrukturer

Bild 11 visar en förenklad grafisk framställning av hur de nyutvecklade noderna förhåller sig till Reachin API-noderna. Varje nod redovisas översiktligt i detta kapitel. För detaljer hänvisas till bifogade header-filer (Bilaga 2). Fullständig och körbar kod kan fås genom kontakt med IPLab⁸ eller författaren⁹.

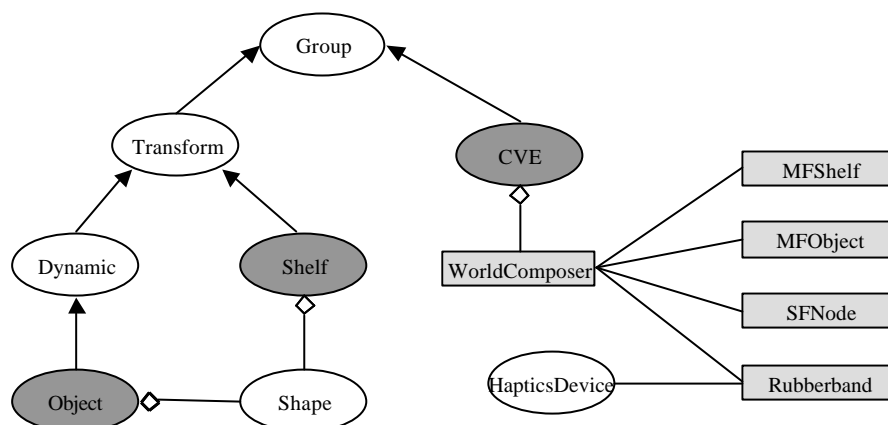


Bild 11. Översikt över prototypens nodstruktur.

Noder med vit bakgrund finns i Reachin API, noder med grå bakgrund är specifika för denna applikation. Fyrkantiga fält betecknar strukturer i CVE-noden.

5.4.1 CVE-noden

CVE-noden fungerar som en simulator för hela miljön. Strukturerna MFShelf och MFOBJECT innehåller alla hyllor respektive objekt som har skapats i VRML-filerna shelves.wrl respektive objects.wrl. Strukturen SFNode innehåller själva rummet som specificeras med grafiska och haptiska egenskaper i filen cve.wrl. Rubberband är den struktur som tar hand om funktionaliteter som att greppa och släppa objekt. Den innehåller flera egna strukturer och innehåller kopplingen till den haptiska enheten. WorldComposer fungerar som en samordnare för alla strukturerna och skapar en gemensam CVE-nod. Nodens innehåll anges i filen world.wrl. Strukturer i CVE-noden visas som fyrkantiga fält i bild 11.

5.4.2 Shelf-noden

Shelf-noden består av en Shape-nod och specificeras i filen Shelf.wrl. Den innehåller endast tre fält som ger möjlighet att bestämma hyllans storlek, färg och textur. Hyllans form är en box och det är dennas sidor som kan specificeras. Eftersom Shelf-noden ärver sina egenskaper från Transform-noden har den redan ett fält för att sätta positionen.

⁸ Se <http://www.nada.kth.se/iplab/> för kontaktinformation.

⁹ E-post: f94-ljm@nada.kth.se

5.4.3 Object-noden

Object-noden innehåller på samma sätt som Shelf-noden, tre fält för storlek, färg och textur. Även den består av en Shape-nod som specificeras i filen Object.wrl. Noden innehåller även en del andra attribut och metoder som används för att bestämma objektens beteende och funktionalitet. Eftersom Object-noden ärver egenskaper från Dynamic-noden, kan objekten påverkas av krafter och röra sig fritt. Det har inte lagts på någon tröghet på objekten eftersom de inte ska rotera.

5.5 Kända problem

I den implementerade miljön finns ett mindre antal kända problem. Dessa förklaras här nedan och det ges samtidigt exempel på möjliga anledningar.

När man flyttar sammansatta objekt tappar objekten ovanpå det greppade objektet, kontakten med varandra och kan trilla av. Detta beror på att metoden `moveComposedObjects()` anropas i realtidsloopen. Problemet uppstår dock sällan vid långsam förflyttning.

Avatarerna når inte i hela rummet, speciellt inte längst fram i hörnen. Det är dock inte något problem att få tag på objekten. Problemet kan åtgärdas genom att skala miljön, men detta får vidare konsekvenser för detaljnivån i miljön.

Det är möjligt att trycka objekten igenom hyllorna. Detta orsakas av att kollisionsdetektionen med hyllorna inte har implementerats att objektet ska flyttas ut av hyllorna när de krockar.

Om man håller ett objekt precis under en hylla på vilken ett objekt är placerat, kommer objektet på hyllan att fastna i det greppade objektet under hyllan. Man kan där efter dra det översta objektet genom hyllan. Detta orsakas delvis av kollisionsdetektionen, men också av att hyllorna är relativt tunna så att objekten hinner få kontakt i y-led och därmed fastna i varandra.

När man greppar ett objekt pressas man ut av objektet. Detta har som tidigare nämnts samband med ytans styvhet. Kraften kan skalas bort och det är även möjligt att stänga av den haptiska ytan.

Ibland hamnar avataren inuti objekten. Detta har möjligen att göra med proxyns storlek eller position. Proxyn är den sfäriska form som Reachin API använder som referens för beräkning av representationens position och interaktion med den haptiska ytan. Applikationsutvecklaren kan själv bestämma proxyns storlek, det vill säga proxyns radie. Den bör dock överensstämma med den grafiska representationens storlek, i detta fall har proxyn samma radie som avataren.

Phantomen som representeras av den gröna avataren vibrerar mycket, speciellt när den befinner sig långt framme i miljön och lyfter ett objekt.

5.5.1 Önskvärda ändringar

Följande ändringar är önskvärda i den implementerade miljön:

- Ökad friktion mellan objekten för att ha möjlighet att flytta med sig flera objekt utan att greppa.
- Ökad och/eller specificerad haptisk återkoppling när man greppar objekt i båda versionerna.
- Icke-haptisk miljö.
- Möjlighet att ångra ett överlämnande.
- Varning till användaren om han eller hon använder för stora krafter.

Kapitel 6 – Användarutvärdering

När man arbetar med användarcentrerad utveckling och användaren inte aktivt är med i designprocessen, är det speciellt viktigt att göra användarstudier. Resultaten från studierna kan användas som ett test på att systemet uppfyller de ställda kraven samt som en utgångspunkt för ändringar i nästa iterationssteg av utvecklingsprocessen.

Applikationens huvudsyfte var att fungera som en haptisk virtuell samarbetsmiljö där man kunde skapa sammansatta objekt från enstaka objekt samt överlämna enstaka eller sammansatta objekt mellan personerna i miljön. Syftet med utvärderingen var att ta reda på hur naturlig miljön upplevdes att arbeta och samarbeta i, eventuella ergonomiska aspekter samt att testa systemets funktionalitet och stabilitet. Det var även önskvärt att studera i vilken grad de två olika versionerna fri och kontrollerad, upplevdes som och fungerade olika.

För att få fram spontana reaktioner och kommentarer från användarna användes metoden *Tänka högt* i samband med lösande av praktiska uppgifter i miljön. *Tänka högt*-metoden är ett mycket effektivt sätt att få kvalitativ data från en användare [51]. Användarna blev därför ombudda att verbalt förmedla alla sina handlingar, reaktioner och funderingar under arbetet. På detta sätt kunde man få fram omedvetna synpunkter eller reaktioner. Dessa skulle användaren möjligen inte kommit ihåg att berätta själva efter testet eller tyckt var av intresse.

De subjektiva aspekterna kommer bäst fram när försökspersonerna själva kan få uttrycka sig fritt. Därför användes en riktad öppen intervju. Denna form av intervju läggs upp så att intervjuaren belyser en vid fråga med några olika frågeområden och följer upp med följdfrågor inom områdena. På detta sätt kan försökspersonerna lotsas in på ett ämnesområde, emedan det fortfarande finns utrymme för att diskutera närliggande aspekter eller andra intressanta områden som dyker upp under intervjun.

De flesta data som kom fram från både den praktiska delen samt intervjun var av kvalitativ typ. För att kunna dra slutsatser och kommentera resultaten måste man göra en kvalitativ analys. Det betyder att man försöker reducera den information man samlat in för att få ut en syntes. Exempel på information kan vara uttalanden från intervjuerna eller från tänka högt-situationen. Dessa rådata kommer senare att utgöra underlag analys och diskussion. Utifrån kvalitativa data kan det i flera sammanhang vara mer intressant att studera användarnas likartade eller olikartade beteenden snarare än hur många som utförde uppgiften på det ena eller andra sättet [27].

6.1 Metod

I användarutvärderingen önskade jag att studera ett antal olika dimensioner av miljön, systemets funktionalitet och stabilitet, interaktionens naturlighet, samarbete samt ergonomi. I tabell 7 presenteras varje dimension med några frågor som beskriver vad de innebär.

Tabell 7. Studerade dimensioner.

Studerade dimensioner (typ av rådata)	
Funktionalitet (objektiv)	<ul style="list-style-type: none">• Kan man skapa ett sammansatt objekt av två eller flera enstaka objekt?• Kan man överlämna ett enstaka objekt?• Kan man överlämna ett sammansatt objekt?
Stabilitet (objektiv)	<ul style="list-style-type: none">• Är miljön konsekvent, beter sig objekten konsekvent?• Uppträder Phantomerna konsekvent?• Uppstår det fel i applikationen, eventuellt när?• Avslutas applikationen omotiverat, varför?
Naturlighet och samarbete (subjektiv)	<ul style="list-style-type: none">• Beter miljön och dess innehåll sig som man förväntar sig?• Hur svårt/lätt är det att förstå vad man kan/inte kan göra i miljön?• Hur intuitiv är miljön?• Känns miljön bra att arbeta i?• Kan man utföra uppgifterna så att man själv är nöjd med resultatet?• Upplevs miljön som konsekvent?• Får man en känsla av att samarbeta med en person?• Påverkar miljön förtroende för samarbetspartnern?
Ergonomi (subjektiv)	<ul style="list-style-type: none">• Var interaktionen ansträngande eller uttröttande?• Var interaktionen för ensidig rörelsemässigt?• Var man tvungen att anpassa sig till och/eller efter utrustningen?• Spelade användningen av stereoglasögonen någon roll för interaktionen?

För att studera systemets funktionalitet och stabilitet användes objektiva rådata. Dessa inhämtades genom observation av försökspersonernas interaktion vid lösandet av praktiska uppgifter i de båda versionerna av miljön. De övriga dimensionerna krävde subjektiva rådata. Dessa inhämtades från försökspersonernas uttalanden i tänka högt-situationen samt intervjuer.

6.1.1 Design av studien

Studien genomfördes med försökspersoner som arbetade parvis eftersom den aktuella miljön var en samarbetsmiljö. Eftersom resultaten baserades på kvalitativa data var det tillräckligt med ett mindre antal försökspersoner. Det var dock önskvärt att försökspersonerna var noviser samt hade olika utbildning eller yrken. Med noviser menades i detta sammanhang en person som inte tidigare hade provat på haptisk återkoppling.

Praktisk uppgift

De två försökspersonerna fick var sin uppgiftslydelse med endast mindre skillnader. Lydelsena var anpassade efter vilken interaktionsenhet de använde (se Bilaga 3). Uppgifterna bestod bland annat av att identifiera det tyngsta objektet bland sex objekt, identifiera objekt med ytor samt överlämna olika kombinationer av objekt. Svaret på dessa uppgifter skulle skrivas ned i uppgiftslydelsen. En uppgift var av mer social karaktär där försökspersonerna var tvungna att samarbeta och själva bestämma hur

uppgiften skulle tolkas. Försökspersonerna fick själva avgöra när de var klara med varje uppgift. Totalt bestod den praktiska delen av tio uppgifter.

Uppgifterna löstes dels i den fria miljön och dels i den kontrollerade. Alla uppgifter var möjliga att lösa i båda miljöerna. Vartannat par började därför med den fria respektive den kontrollerade miljön för att se om det var någon skillnad mellan dem.

Under den praktiska delen hade försökspersonerna möjlighet att prata med varandra eftersom de satt i samma rum. För att samla in subjektive rådata blev de uppmanade om att tänka högt så mycket som möjligt under interaktionen. Försökspersonerna hade dessutom möjlighet till ögonkontakt (se bild 12 och 13). För att kunna analysera försökspersonernas interaktion blev bildskärmen till försökspersonen som representerades med en blå avatar, samt all verbal kommunikation inspelad på videoband. Videokameran placerades snett bakom försökspersonens högra axel.



Bild 12

Bild 13

Utrustningens uppställning vid lösande av den praktiska uppgiften.

Intervju

De två försökspersonerna intervjuades tillsammans och hade möjlighet att komma med kommentarer om upplevelsen av miljön och interaktionen. Försökspersonerna fick frågor om deras spontana reaktioner samt mer specifika frågor (se Bilaga 4). Det fanns dessutom utrymme för att diskutera ytterligare frågor som dök upp under intervjun. Intervjun spelades in på bandspelare.

6.1.2 Försökspersoner

Studien innehöll fem par, det vill säga totalt tio personer. De var fördelade på sex kvinnor och fyra män. Medelåldern var 30,5 år medan medianåldern var 27 år. Tabell 6 visar en översikt över försökspersonernas demografiska fördelning. K står för kvinna och M för man. Kolumnen Relation beskriver parvis försökspersonernas relation till varandra.

Tabell 6. Demografi.

Försök	Kön	Ålder	Yrke/utbildning	Novis	Relation
1	K	25	Studerande (internationella relationer)	Ja	Kompisar
	K	24	Studerande (språk)	Ja	
2	K	36	Biomedicinsk analytiker, studerande (civilingenjör)	Ja	Okända
	M	31	Systemutvecklare	Nej	
3	K	60	Sekreterare	Ja	Gemensam arbetsplats
	K	27	Doktorand (MDI)	Nej	
4	M	28	Journalist	Ja	Okända
	K	27	Sjuksköterska	Ja	
5	M	23	Studerande (civilingenjör)	Ja	Kompisar
	M	24	Studerande (civilingenjör)	Ja	

6.1.3 Utrustning

Vid lösandet av de praktiska uppgifterna använde varje användare en Reachin Display (2a) med Phantom Desktop som haptisk interaktionsenhet. De hade samma visuella representation av miljön (se bild 15) och använde stereoglasögon (se bild 14) under interaktionen.



Bild 14. Stereoglasögon.



Bild 15. Användarens vy.

Bildskärmen samt allt tal spelades in med en digital videokamera och vid intervjun användes en diktafon.

6.2 Resultat från användarutvärderingen

Den praktiska delen innehöll fem uppgifter där försökspersonerna skulle ange ett svar i uppgiftslydelsen. Tillsammans med antal avbrott och den använda tiden kunde detta sammanfattas som kvantitativa resultat. Kommentarer från tänka högt-situationen samt intervjun gav i första hand kvalitativa resultat. Ytterligare studier av videoinspelningarna skulle dock kunna ge möjligheter till att ta fram fler kvantitativa resultat.

6.2.1 Kvantitativa resultat

Tabell 8 redovisar kvantitativa resultat från den praktiska delen av utvärderingen och kan tolkas på följande sätt:

- Första kolumnen beskriver vilket försök resultatet avser samt i vilken ordning de två versionerna fri (F) samt kontrollerad (K) användes.
- De fem följande kolumnerna redovisar de resultat användarna har angett i sina uppgiftslydelser som svar på uppgift 1, 3a, 3b, 3c och 7 (se Bilaga 3).
- Den nedersta raden visar den objektivt rätta lösningen på respektive uppgift.
- De två kolumnerna till höger visar antal avbrott i applikationen under den pågående interaktionen samt hur lång tid försökspersonerna använde på att lösa uppgifterna.

Tabell 8. Kvantitativa resultat från den praktiska delen.

Uppg. Försök	1	3a	3b	3c	7	Antal avbrott	Tid [min]
1 (F-K)	1 1	7 7	5 5	2+3 2+3	5 4	4	45
2 (K-F)	1 9	7 7	5 5	11+12 11+12	6 6	6	55
3 (F-K)	1 1	7 7	5 5	2+4 9+10	6 11	7	55
4 (K-F)	1 1	7 7	5 5	5+9, 7+12 5+9, 7+12	6 5	1	30
5 (F-K)	9 8	7 7	5 5	3+10 3+10	11 5	1	55
Rätt svar	9	7	5	1+6, 2+3, 4+9, 8+10, 11+12	4		

Uppgift 1 samt uppgift 7 bestod i att identifiera vilket objekt bland sex möjliga som var tyngst. Endast två respektive en av tio svarade rätt. Däremot var det sju av tio som valde ett specifikt objekt som tyngst i uppgift 1. I uppgift 7 var det något jämnare fördelat bland de valda objekten. Men svaren är endast fördelade på tre av fem möjliga felaktiga alternativ.

Däremot hittade samtliga försökspersoner det objekt med mest ojämn respektive jämnast yta, uppgift 3a och 3b. Endast två av fem par hittade två objekt med lika jämna/ojämna ytor, uppgift 3c.

6.2.2 Resultat från videoinspelningen

Försökspersonerna fick inga instruktioner om hur de kunde interagera med objekten. Genom att studera videoinspelningarna från studien framgick det tydligt att de flesta försökspersonerna relativt snabbt hittade ett sätt att ta tag i och släppa objekten. Ofta kom försökspersonerna fram till detta genom ett visst samarbete och verbal kommunikation. I den kontrollerade miljön tog det dock längre tid innan försökspersonerna upptäckte hur de kunde släppa objekten. Många försökspersoner gav själva uttryck för att de inte kunde "bli av med" objekten.

Det var i stort sett inga försökspersoner som tappade något objekt i överlämnandet. Några försökspersoner tyckte att när de placerade objekten på en hylla och gav plats för den andra personen var detta ett tillräckligt överlämnande. Alla försökspersonpar klarade dock av och genomförde överlämnanden där båda användarna höll i objektet samtidigt.

Många försökspersoner använde sin representation för att hänvisa rumsligt i miljön, till exempel peka på objekt och hyllor. Det var också en tendens att försökspersoner som kände varandra från tidigare interagerade mer med varandras representationer än de som var okända.

För att undersöka objektets vikt var det vanligast att försökspersonerna tog tag på ovansidan av objektet och lyfte rakt upp. Vid några tillfällen samarbetade försökspersonerna genom att pressa sina avatrar mot objektet på motstående sidor för att kunna lyfta. Detta var oftast innan de upptäckte möjligheten att själva lyfta ett objekt.

Flera försökspersoner gav uttryck för att de inte hade upptäckt att objektets ytor hade en struktur innan de i uppgiftslydelsen blev uppmanade att känna efter. För att studera ytan drog försökspersonerna sin representation fram och tillbaka över objektets yta.

Från videoinspelningarna får man fram att avbrotten i de flesta fall berodde på att användarna använde för stora krafter så att Reachin API av säkerhetsskäl avslutade applikationen. Det var inte implementerat någon varningssignal så att användaren visste om kraften var på gränsen till vad Phantomen klarade av. I vissa fall berodde avbrotten på att programmet inte kunde hantera den uppstådda situationen, till exempel att alla objekt hade hamnat utanför avatrarernas räckvidd.

6.2.3 Resultat från intervjuerna

Intervjuerna samt talet från videoinspelningarna gav kvalitativa data. Nedan finns en sammanställning av några av de synpunkter och reaktioner som framkom i intervjuerna.

Upplevd skillnad mellan fri respektive kontrollerad miljö

Endast ett par, försökspersonpar nummer 5, uppgav att de spontant märkte en skillnad mellan miljöerna. Övriga par uttryckte dock att det var annorlunda till exempel svårare att interagera i den ena miljön. Den fria miljön uppfattades oftast som lättast och mest verklighetstrogen. Ett stort antal försökspersoner tyckte att det var svårt att bli av med objekten i den kontrollerade miljön, det vill säga släppa objekten. En person uttryckte att han eller hon trodde det berodde på ett fel i programmet.

Interaktionen

Alla försökspersoner upplevde interaktionen som relativt naturlig eller realistisk. Några kommenterade dock det orealistiska i att lyfta objekt genom att trycka på en knapp eller ta tag i sidan på ett objekt. Även objektens oförmåga att trilla om tyngdpunkten var utanför hyllan uppfattades som orealistiskt. En person påpekade att den haptiska återkopplingen man fick när man greppade objektet störde uppfattningen av objektets tyngd.

En kommentar som beskriver miljöns intuitiva utformning var "... det förvånansvärda är att man inte reagerar på vad man gör. Man bara gör det man förväntar sig kunna... ". En annan kommentar var att "... programmet fungerade bättre än vad jag hade trott... ". Dessa kommentarer gällde generellt för applikationens funktionalitet och interaktion.

Några försökspersoner upplevde att alla objekt var magnetiska. Ett antal försökspersoner kommenterade också att de trodde de blev påverkade av objektets nummer, storlek och färg. De trodde att objekt med högre siffror, mörkare färg eller större storlek skulle vara tyngre än objekt med lägre siffra, ljusare färg eller mindre storlek.

Uppgiftslösning

Samtliga försökspersoner tyckte det gick bra att lösa uppgifterna samt att de var nöjda med den egna insatsen. Några gav uttryck för att uppgiftslydelsen var något vag och gav utrymme för tolkningar. Av denna anledning var det i vissa tillfällen svårt att själv avgöra om uppgiften var rätt utförd. En person uttryckte att uppgifterna var för enkla, och ville ha haft klurigare uppgifter eftersom själva överlämnandet var så lätt.

Samtliga försökspersoner tyckte att interaktionen var rolig och lätt. Den svåraste uppgiften var enligt försökspersonerna att känna skillnad på objektets tyngd. Även att hitta två likadana ytor uppfattades som svårt.

För att kontrollera att ett objekt överlämnades uppgav de flesta försökspersonerna att de använde den visuella återkopplingen. Även den verbala kommunikationen var viktig. Många gav också uttryck för att de kände efter om de hade kontroll över objektet eller om samarbetspartnerns hade tagit tag i det, eftersom det märktes.

Samarbete och närvaro

Samtliga försökspersoner upplevde att de var tillsammans i miljön. De allra flesta uttryckte att det hade varit mindre roligt att vara ensam i en liknande miljö. Att man kunde interagera med en annan person, både haptiskt, visuellt och verbalt gjorde att interaktionen och samarbetet blev roligare.

Att man kunde prata med varandra uppfattades av försökspersonerna som mycket viktigt i samarbetet. Om de hade varit placerade i vart sitt rum med telefonkontakt kunde det ha likaställts med den faktiska situationen enligt försökspersonerna själva.

En person gav uttryck för att den sociala känslan var liten med en så enkel avatar eftersom den inte gav uttryck för specifika egenskaper hos användaren. Hade man själv kunnat välja sin avatars utseende skulle detta ha bidragit till en större identitetskänsla enligt försökspersonen.

Andra försökspersoner uttryckte däremot att de kunde identifiera sig med sin avatar. Många kände dessutom att objekt med samma färgskala som deras representation hörde till dem. Till exempel uppfattades användare som representerades av en blå avatar att de blåfärgade objekten var deras.

Förtroende och säkerhet

Samtliga försökspersoner tyckte att det kändes tryggt att lämna ifrån sig ett objekt. Flera uttryckte en viss skepsis vid första försöket, men när de såg att det var möjligt att överlämna objekt, var detta lätt. Som tidigare nämnts använde de flesta i första hand den visuella återkopplingen eller den verbala kommunikationen som försäkring för att uppgiften var löst. Den haptiska återkopplingen spelade dock en viss roll eftersom flera påpekade att man kunde känna om den andra personen hade tagit tag i objektet.

Försökspersonerna i de par där man kände varandra från tidigare sade att de litade på sin samarbetspartner och såg ingen anledning till att denna inte skulle ta emot objektet. De var mer skeptiska till systemets förmåga att hantera överlämnandet än till den andra personen.

Om man tyckte att det var lättast att ge eller ta emot objekt berodde delvis på hur man uppfattade ansvaret i överlämnandet. Av de som tyckte det var lättare eller säkrare att ta emot var en anledning att då hade man kontroll över fortsättningen. En annan anledning var att försökspersonen ansåg att det var den som gav ifrån sig objektet som hade ansvaret för att mottagaren kunde ta emot det. Om mottagaren inte fick tag i objektet var detta alltså sändarens fel. En person som tyckte att det var säkrast att ge ifrån sig objekt uppgav att anledningen var att man kände när den andra hade tag och då kunde man släppa.

Utrustningen och ergonomi

De flesta försökspersonerna tyckte Phantomen var en rolig interaktionsutrustning. Det var dock delade meningar om hur man ville hålla den. Några ville använda den som ett pekdon och ville därför hålla den åt fel håll så att änden pekade neråt (se bild 3 för korrekt grepp). En annan tyckte den såg ut som en vev och ville därför hålla den som en vev, det vill säga greppa runt den med hela handen.

Merparten av försökspersonerna upplevde att det inte var tröttnande att utföra uppgifterna i den haptiska miljön. Några kommenterade dock att de trodde det skulle vara jobbigt att arbeta i miljön en hel dag. En person upplevde att arbetet orsakade spänningar i kroppen samt att pekfingret fungerade som stöd i alla lyft och att man därför blev trött i detta när man jobbade en längre tid.

Ett försökspersonpar påpekade att den lutande skärmen kunde uppfattas som något hotande och att man därför valde att sitta lite längre ifrån skärmen än nödvändigt. En annan kommentar var att man möjligen skulle känna sig lite låst till en viss arbetsplats eller arbetsställning.

Endast en försöksperson tyckte att stereoglasögonen orsakade trötthet i ögonen. Ett antal kommenterade att glasögonen var lite stora samt att det var svårt att läsa uppgiftslydelsen med dem på. I övrigt fungerade de bra.

Ett irritationsmoment i miljön var delvis vibrationer från speciellt den ena Phantomen samt ljud från robotarmens motor. Ljudet uppfattades dock av några som auditiv återkoppling vilket uppskattades.

Framtida tillämpning

De flesta försökspersonerna hade problem med att föreställa sig en lämplig situation att använda applikationen i. De trodde att haptiken inte skulle tillföra någon säkerhet i en virtuell miljö. De tyckte dessutom att e-post fungerade tillräckligt säkert för överlämnande av dokument och liknande information enligt deras uppfattning. Samtliga försökspersoner uttryckte en större skepsis till andra former för säkerhet, till exempel kryptering av meddelanden och inloggningsprocedurer. Flera uttryckte dessutom att de inte skulle vilja överlämna viktig information i en offentlig miljö överhuvud taget. Med offentlig miljö menade de flesta en plats för synkron textkommunikation (tjatt) eller liknande på Internet.

6.3 Analys av resultat från användarutvärderingen

Både de kvantitativa och kvalitativa resultaten var av stort intresse. Nedan följer en analys av några av de mest intressanta resultaten.

6.3.1 Användbarhet

Grundat på resultat från av både den praktiska delen av utvärderingen och intervjun kan miljön karakteriseras som intuitiv och användbar. Alla försökspersonerna klarade av att utföra uppgifterna vid stort sett första försöket. Endast ett fåtal överlämnanden misslyckades. Även tillfredsställelsen var hög då samtliga försökspersoner upplevde att de kunde utföra uppgifterna så att de själva var nöjda. Utan undantag tyckte försökspersonerna att interaktionen hade varit rolig. De flesta tyckte även att båda versionerna av miljön var realistiska och naturliga. Den fria miljön upplevdes dock som lättast att arbeta i.

6.3.2 Samarbete

Utan att få instruktioner om hur objekten kunde manipuleras eller vilka funktioner som fanns i miljön, kunde försökspersonerna utföra uppgifterna relativt effektivt. De två försökspersonerna använde ofta den verbala kommunikationen för att klargöra hur olika uppgifter kunde utföras. Försökspersonerna uttryckte själva att det hade varit mindre roligt ifall de inte hade haft möjlighet att prata med varandra under försöket. De trodde själva att vikten av att sitta i samma rum så länge de hade audiokontakt, var negligerbar. Detta överensstämmer med tidigare studier som visar att den auditiva kanalen är mycket viktig vid samarbete [12, 23].

6.3.3 Grafiskt gränssnitt

Det grafiska gränssnittet spelar en stor roll i hur man upplever en miljö. Försökspersonerna använde avatarernas och objektens färger för att definiera vilken användare som "ägde" objekten. Det verkade dessutom som om försökspersonerna förväntade

sig att de två versionerna hade samma egenskaper eftersom de hade samma visuella utformning.

Visuella egenskaper som storlek och färg påverkade möjligen hur objekten upplevdes haptiskt. Eftersom objekten hade liknande färger och samma visuella ytstruktur, uppfattades de som gjorda av samma material. Det var därför några försökspersoner som tyckte det var konstigt att de skulle väga olika mycket. Andra personer påpekade vikten av färgens nyans, objektens storlek samt numreringen för hur objekten upplevdes.

6.3.4 Haptiskt användargränssnitt

Alla försökspersoner lyckades att hitta det objekt med jämnast respektive ojämnast yta. Att jämföra två ytor var däremot svårare och endast två av fem par hittade två objekt med samma textur. En faktor som bidrog till att försvåra uppgiften var att Reachin API använder samma textur på alla sidor av en box. Det innebär att två sidor på samma objekt kan kännas olika ifall de har olika storlek eftersom texturen skalas efter boxens mått. Av denna anledning skulle det kanske i framtiden vara mer användarvänligt att använda med kuber istället för boxar.

En annan anledning till svårigheten kan ha varit avatarernas storlek i förhållande till texturen. Ju mindre radie interaktionsenheten har desto lättare är det att känna små skillnader. Det är därför möjligt att det med en mindre, det vill säga mer noggrann avatar, hade varit lättare att känna strukturella skillnader. Detta kan jämföras med att känna på ett objekt med boxningshandskar eller plasthandskar i verkliga livet.

Endast tre av tjugo svar var rätt när det gäller fastställande av det tyngsta objektet. Ett intressant fenomen var att sju av tio personer angav i uppgift 1 ett specifikt objekt som tyngst även om detta var fel. Det aktuella objektet var av liten storlek men med en mörk färg. Några försökspersoner uttryckte att de trodde de lurades av objektens färg, att mörka objekt var tyngre än ljusa. Detta förklarar dock inte varför inga försökspersoner valde det stora objektet med den mörka färgen eftersom detta vägde mer än dubbelt så mycket som det felaktigt valda objektet.

En liknande tendens visades i uppgift 7, men där var de felaktiga svaren fördelade på tre av fem möjliga objekt. Ett av dessa objekt hade en mörk färg medan de två andra hade en mellanfärg. Ett objekt var stort medan två var små. Totalt sett var det alltså fler försökspersoner som valde samma felaktiga objekt i både uppgift 1 och 7, än det totala antalet rätta svar. Tabell 9 visar antal försökspersoner (FP) av totalt tio, som upplevde ett visst objekt som tyngst bland sex möjliga.

Tabell 9. Objektens vikt samt försökspersonernas svar i uppgift 1 och 7.

Objekt nr	Uppgift 1 Vikt [kg]	Antal FP	Objekt nr	Uppgift 7 Vikt [kg]	Antal FP
1	0,2	7	4	1,0	1
2	0,3	0	5	0,5	3
3	0,2	0	6	0,3	4
7	0,5	0	10	0,5	0
8	0,2	1	11	0,2	2
9	1,0	2	12	0,4	0

Skillnaderna i vikterna var relativt små och det är svårt att känna små viktskillnader. Men det tyngsta objektet var dubbelt så tungt som det näst tyngsta och borde därför kunna identifieras. Antalet felaktiga svar kan dels bero på att objektets vikt skalades när det lyftes. En försöksperson uttryckte också att den haptiska återkopplingen som gavs vid själva greppandet av objektet var störande för uppfattningen av objektets vikt. Det visuella intrycket från färg och storlek på objektet spelar naturligtvis också roll för hur man bedömer objektet.

Försökspersonernas sätt att utforska objektets ytstruktur och vikt överensstämde väl med de utforskningsprinciper som man använder i verkliga livet [28].

6.3.5 Fri respektive kontrollerad miljö

Försökspersonerna fick ingen information om vilka skillnader som fanns i de två miljöerna. De fick endast veta att de skulle testa en annan miljö. Anledningen var för att se om skillnaderna var märkbara och om det påverkade effektiviteten och interaktionen.

Ingen av försökspersonerna uppfattade direkt någon spontan skillnad mellan miljöerna. De hade samma grafiska gränssnitt och det är möjligt att detta gav intrycket att det var samma miljö. Ett par kom fram till efter lite reflektion, att det fanns en skillnad. De kunde även förklara vilka dessa skillnader var. Detta par var det enda som upplevde den kontrollerade miljön som säkrare att överlämna objekt i.

Trots att försökspersonerna inte upplevde någon spontan skillnad mellan miljöerna, upplevdes generellt den fria miljön som lättast att arbeta i. Det framkom under intervjun efter att försökspersonerna fick möjlighet att ytterligare reflektera över den egna interaktionen. Anledningen till att den fria miljön upplevdes som lättast var antagligen av att den var mer intuitiv eftersom den inte innehöll några kontrollerande parametrar. Försökspersonerna uppfattade det dessutom som något irriterande att inte kunna släppa objektet. Det är möjligt att den kontrollerade miljön hade varit mer ändamålsenlig om försökspersonerna hade känt till vilka begränsningar eller hjälpmedel den innehöll, det vill säga fått veta vilka skillnader som fanns i de två versionerna.

6.4 Slutsatser från användarutvärderingen

Miljön kan sägas vara intuitiv och användbar och den är väl lämpad för att överlämna objekt i. Den kontrollerade miljön är dock mest ändamålsenlig om användarna känner till hur miljön stödjer arbetet.

Det är svårt att känna små viktskillnader samt hitta objekt med likadana ytstrukturer så länge de skiljer sig mindre åt. Vid extrema skillnader kan objekt lätt identifieras utifrån ytstrukturen. Det är dock viktigt att tänka på avatarens storlek i förhållande till strukturens storlek.

Ett effektivt samarbete antas vara beroende av möjligheten till att prata med varandra under arbetet. Enligt försökspersonerna hade det varit tråkigare att utföra uppgifterna eller liknande uppgifter om de var själva i en motsvarande miljö.

Att kunna känna på varandras representation samt alla objekt i miljön, gjorde interaktionen roligare och mer naturlig. Det var dock den visuella återkopplingen som oftast användes som primär försäkring på att en uppgift var genomförd, till exempel att överlämna ett objekt. Även den verbala kommunikationen var viktig i denna process.

Phantomen ansågs vara ett trevligt interaktionsverktyg dock med delade meningar om hur den borde hållas. Stereoglasögonen var i stort sett oproblematiskska att använda. Utrustningen i sin helhet ansågs dock uppmana till fokuserat arbete och kan vara svårt att kombinera med annat vanligt skrivbordsarbete som till exempel att läsa ett dokument.

Kapitel 7 – Avslutning

7.1 Sammanfattning och slutsatser

Syftet med detta examensarbete var att ta fram designkriterier och utveckla en generell, virtuell samarbetsmiljö. Miljön skulle anpassas för att kunna stödja ett säkert överlämnande av objekt. Den skulle även användas för att kunna studera effekten av haptisk återkoppling vid överlämnande av objekt. Arbetet har bestått av en större genomgång av relevant litteratur, utveckling och implementering av en prototyp samt en användarutvärdering.

Efter problemanalysen och inhämtning av relevant bakgrundskunskap fastställdes först övergripande designkriterier. De blev sedan specificerade och delades upp i objektiva och subjektiva krav. De objektiva kraven kunde testas under implementeringen, men för att testa de subjektiva kraven behövdes en användarutvärdering med noviser, det vill säga ovana användare.

Baserat på testerna utförda under implementeringen samt användarstudien kan man konstatera att miljön uppfyller ställda objektiva krav förutom möjligheten att avbryta ett överlämnande. Denna funktion är fortfarande inte implementerad i miljön. När det gäller de subjektiva kraven visade användarutvärderingen att försökspersonerna uppfattade båda miljöerna som intuitiva och naturliga. Den fria miljön upplevdes dock som något lättare att arbeta i. Både graden av tillfredsställelse och ändamålsenlighet var hög, och man kan därmed säga att miljön är användbar.

Det kan dock diskuteras huruvida den kontrollerade miljön stödjer ett säkert överlämnande mer än den fria miljön. Det var ytterst få försökspersoner som upptäckte skillnaden mellan miljöerna även om flera uttryckte att det fanns en viss skillnad. Den fria miljön uppfattades som något lättare att interagera i. Det är dock troligt att den kontrollerade miljön hade upplevts som mer ändamålsenlig, om försökspersonerna hade fått information om vilka regler som gällde i miljön. Det enda paret av försökspersoner som upplevde den kontrollerade miljön som säkrare att överlämna objekt i hade upptäckt dessa regler.

Den virtuella miljön stödjer samarbete i form av haptisk och visuell kommunikation. Försökspersonerna hade under utvärderingen även möjlighet att tala med varandra, något som visade sig vara mycket viktigt i samarbetsprocessen. Det hade därför varit intressant att genomföra en liknande studie med endast syn och känsel som återkopplings- och kommunikationsmodalitet. Detta kräver dock att försökspersonerna sitter i olika rum.

Av olika anledningar har det inte tagits fram en icke-haptisk version av miljöerna. Detta beror delvis på tidsbrist samt svårigheter med att ”stänga av” haptiken i Reachin API. För att kunna studera effekten av just haptisk återkoppling är det dock en nödvändighet att en sådan version implementeras.

Som resultat av arbetet kan man konstatera att det i en samarbetsmiljö är viktigt att ta hänsyn till kommunikations- och återkopplingsmodalitet när samarbetet ska optimeras. När den virtuella miljön utformas är det viktigt att ha ett väldefinierat grafiskt användargränssnitt då egenskaper som färg, storlek och bildtextur påverkar hur objekt upplevs. Men även det haptiska användargränssnittet är av stor betydning, till exempel storleksförhållandet mellan den haptiska strukturen och användarens avatar samt objektens vikt och styvhet. Det är dessutom viktigt att anpassa de haptiska egenskaperna i miljön efter interaktionsenhetens egenskaper och begränsningar.

7.2 Haptik i framtiden

Haptik är i dag ett relativt okänt begrepp bland allmänheten. Jag tror dock att känselgränssnitt har en stor potential till att skapa nya former för människa-datorinteraktion och människa-dator-människainteraktion där användaren står i centrum. Det slående med haptiska gränssnitt är hur intuitiv interaktionen blir och att den för de allra flesta orsakar en "wow-effekt" när man för första gången använder dessa.

Den haptiska utrustningen är i dag mycket kostbar och det dröjer antagligen flera år innan utrustning som PHANTOM finns vid varje dator. Det finns dock redan i dag enklare haptisk utrustning som möss och spelkontroller som använder sig av haptisk återkoppling för att effektivisera interaktionen och göra den mer användarvänlig.

I framtiden kommer det att vara applikationers innehåll och utformning som blir den avgörande faktorn för en produkts användning och därmed lönsamhet. Multimodala gränssnitt tillför flera dimensioner till interaktionen och därmed möjligheter till ökad användbarhet. De öppnar också för att använda metaforer kopplade till flera sinnen. Känselgränssnitt ger till exempel möjligheten att fysiskt simulera uttryck som "tunga beslut" och "glider ur händerna".

Det återstår mycket forskning och produktutveckling på området och inte minst en hel del utveckling av kommersiella applikationer. Ett steg på vägen är att försöka tänka nytt och inse att människan är komplex i sitt sätt att interagera både med den reella och virtuella verkligheten, och speciellt i förhållande till andra människor. En viktig fråga att ta ställning till vid utveckling av virtuella miljöer är vad som är deras egentliga syfte. Är de förknippade med en önskan om att simulera den reella verkligheten, eller ska den virtuella miljön fungera som en möjlighet till och stöd för interaktion som inte är genomförbar i den reella världen? Har man svaret på den frågan har man kommit långt på väg mot en användbar applikation.

Referenser

- [1] Barfield, W., Hendrix, C., Bjomeseth, O., Kaczmarek, K. A. and Lotens, W. (1995). *Comparasion of Human Sensory Capabilities with Technical Specifications of Virtual Environment Equipment*, Presence, Vol. 4, No. 4, Fall 1995, pp. 329-356.
- [2] Basdogan, C., Ho, C.-H., Srinivasan, M. A. and Slater, M. (2000). *An Experimental Study on the Role of Touch in Shared Virtual Environments*, ACM Transactions on Computer-Human Interaction, Vol. 7, No. 4, December 2000, pp. 443-460.
- [3] Benford, S., Bowers, J., Fahlén, L., Greenhalgh, C., Mariani, J. and Rodden, T. (1995). *Networked Virtual Reality and Cooperative Work*, Presence, Vol. 4, No. 4, Fall 1995, pp. 364-386.
- [4] Benford, S., Bowers, J., Fahlén, L., Greenhalgh, C. and Snowdon, D. (1995). *User Embodiment in CVE*, Proceedings of ACM CHI'95, pp. 242-249.
- [5] Biggs, J. and Srinivasan, M. A. (2001). *Chapter 5. Haptic Interfaces*, Handbook of Virtual Environment Technology (Kay M. Stanney, Ed.), Lawrence Erlbaum Associates, Inc. (in press). Tillgänglig 2001-06-18 på URL http://touchlab.mit.edu/publications/2001_001.pdf
- [6] Boff, K. R., Kaufman, L., Thomas, J. P. (1986). *Chapter 31 Tactual Perception*, Handbook of Perception and Human Performance, Volume II Cognitive Processes and Performance, John Wiley & Sons, Inc.
- [7] Bowers, J., Pycock, J. and O'Brian, J. (1996). *Talk and Embodiment in Collaborative Virtual Environments*, Proceedings of CHI'96, April 1996, pp. 58-65.
- [8] Bruegge, B. och Dutoit, A.H. (2000). *Object-Oriented Software Engineering: Conquering Complex and Changing Systems*, Prentice Hall. ISBN 0-130489725-0.
- [9] Buttolo, P., Oboe, R., Hannaford, B. and McNeely, B. (1996). *Force Feedback in Shared Virtual Simulations*, MICAD, France, 1996.
- [10] Capps, M., Watsen, K. and Zynda, M. (1999). *Cyberspace and Mock Apple Pie. A Vision of the future of Graphics and Virtual Environments*, IEEE Computer Graphics and Applications, November/ December 1999, Volume 19, Issue 6, pp. 8-11.
- [11] Dix, A., Finlay, J., Abowd, G. and Beale, R. (1997, 1993). *Chapter 13: Groupware*, Human-Computer Interaction. New York: Prentice Hall, Inc.
- [12] Eberts, R. E. (1994). *User Interface Design*. New Jersey: Prentice Hall, Inc. ISBN 0-13-140328-1.
- [13] Ellis, C. A., Gibbs, S. J. and Rein, G. L. (1991). *Groupware: Some Issues and Experiences*, Communications of the ACM, Vol. 34, No. 1, pp. 9-28.
- [14] Fraser, M., Glover, T., Vaghi, I., Benford, S., Greenhalgh, C., Hindmarsh, J. and Heath, C. (2000). *Reavealing the Realities of Collaborative Virtual Reality*, Proceedings of CVE 2000, pp. 29-37.

- [15] Fuchs, L., Pankoke-Babatz, U. and Prinz, W. (1995). *Supporting Cooperative Awareness with Local Event Mechanism: The Groupdesk System*, Proceedings of ECSCW 95, September 10-14, Stockholm, Sweden, pp. 247-262.
- [16] Gleiss, N. (1991). *Vad är användbarhet?* Särtryck ur Tele 2/91, Telia AB.
- [17] Guye-Vuillème, A., Capin, T. K., Pandzic, I. S., Thalman, N. M and Thalman, D. (1998). *Nonverbal Communication Interface for Collaborative Environments.*, Proceedings of CVE'98, pp. 105-112.
- [18] Hindmarsh, J., Fraser, M., Heath, C., Benford, S. and Greenhalgh, C. (1998). *Fragmented Interaction: Establishing Mutual Orientation in Virtual Environments*, Proceedings of CSCW 98, Seattle WA, pp. 217-226.
- [19] Hindmarsh, J., Fraser, M., Heath, C., Benford, S. and Greenhalgh, C. (2000). Object-Focused Interaction in Collaborative Virtual Environments, ACM Transactions on Computer-Human Interaction, Vol. 7, No. 4, December 2000, pp. 477-509.
- [20] Ishii, M., Nakata, M. and Sato, M. (1994). *Networked SPIDAR: A Networked Virtual Environment with Visual, Auditory and Haptic Interactions*, Presence, Vol. 3, No. 4, Fall 1994, pp. 351-359.
- [21] Jansson, G. (2000). *Basic Issues Concerning Visually Impaired People's Use of Haptic Displays*, International Conference on Disability, Virtual reality and Associated Technologies, Alghero, Sardinia, Italy.
- [22] Jansson, G., Billberger, K., Petrie, H., Colwell, C., Kornbrot, D., Fänger, J., König, H., Hardwick, A. and Furner, S. (1999). *Haptic Virtual Environments for Blind People: Exploratory Experiments with two Devices*, The International Journal of Virtual Reality, Vol. 4, No. 1, 1999, pp. 10-20.
- [23] Jensen, C., Farnham, S. D., Drucker, S. M. and Kollock, P. (2000). *The Effect of Communication Modality on Cooperation in Online Environments*, Proceedings of ACM CHI 2000, Vol. 1, pp. 470-477.
- [24] Jones, S. and Marsh, S. (1997). *Human-Computer-Human Interaction: Trust in CSCW*, ACM SIGCHI Bulletin, Vol. 29, No. 3, 1997, pp. 36-40.
- [25] Karvonen, K. (2000). *The Beauty of Simplicity*, Proceedings of the 2000 International Conference on Intelligent User Interfaces 2000, Arlington VA, USA, pp. 85-90.
- [26] Kupst, S., Mehravari, N., Olson, M. and Rush, S. (1997). *Military and Civilian Applications of Virtual Co-Location and Collaborative Environments and Associated Technologies*, Proceedings of MILCOM 97, Vol. 1, 2-5 Nov. 1997, pp. 218-222.
- [27] Lantz, A. (1993). *Intervjumetodik*, Studentlitteratur. ISBN 91-44-38131-X.
- [28] Lederman, S.J. and Klatzky, R.L. (1987). *Hand Movements: A Window into Haptic Object Recognition*, Cognitive Psychology, 19, pp. 342-368.
- [29] Lenman, S., Sallnäs, E.-L., Serenius, B., Sundblad, O., Uhlin, T., Wadman E.-M. och Winroth, U. (1999). *Formella VR-möten*, Technical report TRITA-NA-D9914, CID-58, KTH, Stockholm, Sweden.

- [30] Massie, T. H. and Salisbury, J. K. (1994). *The PHANTOM Haptic Interface: A Device for Probing Virtual Objects*, Proceedings of the ASME Winter Annual Meeting, Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, Chicago, IL, November, 1994.
- [31] McCarthy, L., Stiles, R., Johnson, L and Rickel, J. (1998). *Enabeling Team Training in Virtual Environments*, Proceedings of CVE 1998, pp. 113-121.
- [32] Muir, B. M. (1987). *Trust Between Humans and Machines, and the Design of Decision Aids*, International Journal of Man-Machine Studies, Vol. 27, No. 5/6, 1987, pp. 527-539.
- [33] Münch, S. and Dillman, R. (1997). *Haptic Output in Multimodal User Interfaces*, IUI 97, Orlando, Florida, pp. 105-112.
- [34] Oakley, I., Brewster, S. and Gray, P. D.(2001). *Solving Multi-Target Haptic Problems in Menu Interaction*, accepted for publication in ACM CHI'2001.
- [35] Oakley, I., McGee, M. R., Brewster, S. A. and Gray, P. D. (2000). *Putting the Feel into Look and Feel*, Proceedings of CHI'2000, The Hague, NL.
- [36] Oakley, I., Brewster, S. and Gray, P. (2001). *Can You Fell the Force? An Investigation of Haptic Collaboration in Shared Editors*, Proceedeings of Eurohaptics 2001, Birmingham, UK., pp. 54-59.
- [37] Olson, G. M., Olson, J. S., Carter, M. R. and Storrösten, M. (1992). *Small Group Design Meetings: An Analysis of Collaboration*, Human-Computer Interaction, Vol. 7, 1992, pp. 347-374.
- [38] Salem, B. and Earle, N. (2000). *Designing a Non-Verbal Language for Expressive Avatars*, Proceedings of CVE 2000, pp. 93-101.
- [39] Salisbury, J. K., Jr (1999). *Making Graphics Physically Tangible*, Communications of the ACM, Vol. 42, No. 8, August 1999, pp. 74-81.
- [40] Salisbury, J. K. and Srinivasan, M. A. (1997). *Phantom-Based Haptic Interaction with Virtual Objects*, IEEE Computer Graphics and Applications, Vol. 17, No. 5, 1997.
- [41] Salisbury, K., Brock, D., Massie, T., Swarup, N. and Zilles, C. (1995). *Haptic Rendering: Programming Touch Interaction with Virtual Objects* Proceedings of the 1995 symposium on Interactive 3D graphics, April 9 - 12, 1995, Monterey, CA USA, pp. 123-130.
- [42] Sallnäs, E.-L., Rasmus-Gröhn, K. and Sjöström, C. (2000). *Supporting Presence in Collaborative Environments by Haptic Force Feedback*, ACM Transactions on Computer-Human Interaction, Vol. 7, No. 4, December 2000, pp. 461-476.
- [43] Salvador, T., Scholtz, J. and Larson, J. (1996). *The Denver Model for Groupware Design: (Yeaaaaa Haaaaa!)*, ACM SIGCHI Bulletin, Vol. 28, No. 1, 1996, pp. 52-58.
- [44] Short, J., Williams, E. and Christie, B. (1976). *The social psychology of telecommunications*. London: Wiley.
- [45] Srinivasan, M. A. and Basdogan, C. (1997). *Haptics in Virtual Environments: Taxonomy, Research Status, and Challenges*, Computers & Graphics, Vol. 21, No. 4, 1997, pp. 393-404.

- [46] Srinivasan M. A., Basdogan C., and Ho C.-H. (1999). *Haptic Interactions in the Real and Virtual Worlds*, In *Design, Specification and Verification of Interactive Systems '99*, Springer-Verlag Wien.
- [47] Stanney, K. M., Mourant, R. R. and Kennedy, R. S. (1998). *Human Factors Issues in Virtual Environments: A Review of the Literatur*, *Presence*, Vol. 7, No. 4, August 1998, pp. 327-351.
- [48] Tan, H. Z., Srinivasan, M. A., Eberman, B. and Cheng, B. (1994). *Human Factors for the Design of Force-Reflecting Haptic Interfaces*, *Proceedings of Winter Annual Meeting of the American Society of Mechanical Engineers: Dynamic Systems and Control, DSC*, Vol. 55-1, pp. 353-359.
- [49] Taylor II, R. M. (1999). *Scientific Applications of Force Feedback: Molecular Simulation and Microscope Control*, Course notes for "Haptics: From Basic Principles to Advanced Applications", SIGGRAPH'99.
- [50] *Nationalencyklopedin*, <http://www.ne.se/>, juni 2001.
- [51] *USOR – A Collection of User Oriented Methods*, <http://www.nada.kth.se/cid/usor/>, november 2001.

Bilaga 1 – VRML-filer

```
# -----  
#   Object.wrl  
# -----  
# VRML V2.0 utf8  
DEF ObjectSHAPE Shape {  
    geometry DEF GEO Box {  
        size 0.05 0.05 0.05 }  
    appearance DEF ObjectAPP Appearance {  
        surface DEF ObjectSURF BumpmapSurface {  
            stiffness 500  
            texture ImageTexture {  
                url "bump_0.png" }  
            bumpHeight 0 }  
        material DEF ObjectMAT Material {  
            diffuseColor 1 1 1  
            specularColor 0.4 0.3 0.1 }  
        texture ImageTexture {  
            url "noTexture.png" } } }  
  
# -----  
#   Shelf.wrl  
# -----  
# VRML V2.0 utf8  
DEF ShelfSHAPE Shape {  
    geometry DEF ShelfGEO Box {  
        size 0.01 0.001 0.005 }  
    appearance DEF ShelfAPP Appearance {  
        surface DEF ShelfSURF FrictionalSurface {  
            stiffness 500  
            startingFriction 0.3  
            stoppingFriction 0.3  
            dynamicFriction 0.5 }  
        material DEF ShelfMAT Material {  
            diffuseColor 1 1 1 }  
        texture ImageTexture {  
            url "shelf_floor.png" } } }  
  
# -----  
#   Ceiling.wrl # width: 0.30, depth: 0.20  
# -----  
# VRML V2.0 utf8  
DEF CEILING Shape {  
    geometry IndexedFaceSet {  
        coord Coordinate {  
            point [ -0.15 0 -0.1  
                   0.15 0 -0.1  
                   0.15 0 0.1  
                   -0.15 0 0.1 ] }  
        coordIndex [ 0 1 2 3 ] }  
    appearance Appearance {  
        surface FrictionalSurface {}  
        material Material {  
            transparency 1  
            diffuseColor 1 1 1 } } }  
  
# -----  
#   Floor.wrl # width: 0.30, depth: 0.20  
# -----  
# VRML V2.0 utf8  
DEF FLOOR Shape {  
    geometry IndexedFaceSet {  
        coord Coordinate {  
            point [ -0.15 0 -0.1  
                   0.15 0 -0.1
```

```
                0.15 0 0.1
                -0.15 0 0.1 ] }
    coordIndex [ 3 2 1 0 ] }
    appearance Appearance {
        surface FrictionalSurface {
            dynamicFriction 0.5
            stoppingFriction 0.3
            startingFriction 0.3 }
        material Material {
            diffuseColor 0.67 0.81 0.88 }
        texture ImageTexture {
            url "floor.png" } } }

# -----
#     frontWall.wrl # width: 0.35, height: 0.2
# -----
# VRML V2.0 utf8
DEF FRONTWALL Shape {
    geometry IndexedFaceSet {
        coord Coordinate {
            point [ -0.175 -0.1 0
                  -0.175 0.1 0
                  0.175 0.1 0
                  0.175 -0.1 0 ]}
        coordIndex [ 0 1 2 3 ] }
    appearance Appearance {
        surface FrictionalSurface {}
        material Material {
            transparency 1 } } }

# -----
#     leftAvatar.wrl
# -----
#VRML V2.0 utf8
DEF POINTER_XFORM Transform { }
DEF STYLUS Shape {}
DEF TIP Shape {
    appearance Appearance {
        material Material {
            diffuseColor 0.4 0.4 0.9
            shininess 0.5 } }
    geometry DEF POINTER_TIP Sphere {
        radius 0.004 } }

# -----
#     objects.wrl
# -----
#VRML V2.0 utf8

# --- Left side boxes --- ( blue ) -----
DEF LBox1 Object {
    size 0.02 0.012 0.02
    color 0.05 0.75 1
    texture ImageTexture { url "3.png" }
    stiffness 100
    bumpTexture ImageTexture {url "bump_2.png" }
    bumpHeight 0.005
    mass 0.2
    position -0.04184 0.0365 0.03566 }
DEF LBox2 Object {
    size 0.02 0.012 0.02
    color 0.25 0.50 1
    texture ImageTexture { url "2.png" }
    stiffness 300
    bumpTexture ImageTexture {url "bump_2.png" }
    bumpHeight 0.005
    mass 0.3
    position -0.07062 0.0365 0.05232 }
DEF LBox3 Object {
```

```
        size 0.02 0.012 0.02
        color 0.35 0.05 1
        texture ImageTexture { url "1.png" }
        stiffness 100
        bumpTexture ImageTexture {url "bump_3.png" }
        bumpHeight 0.005
        mass 0.5
        position -0.09946 0.0365 0.06897 }
DEF LBox4 Object {
    size 0.026 0.014 0.026
    color 0.05 0.75 1
    texture ImageTexture { url "9.png" }
    stiffness 500
    bumpTexture ImageTexture {url "bump_6.png" }
    bumpHeight 0.005
    mass 1
    position -0.04184 0.0675 0.03566 }
DEF LBox5 Object {
    size 0.026 0.014 0.026
    color 0.25 0.50 1
    texture ImageTexture { url "8.png" }
    stiffness 100
    bumpTexture ImageTexture {url "bump_1.png" }
    bumpHeight 0.005
    mass 0.2
    position -0.07062 0.0675 0.05232 }
DEF LBox6 Object {
    size 0.026 0.014 0.026
    color 0.35 0.05 1
    texture ImageTexture { url "7.png" }
    stiffness 200
    bumpTexture ImageTexture {url "bump_5.png" }
    bumpHeight 0.005
    mass 0.5
    position -0.09946 0.0675 0.06897 }

# --- Right side boxes --- (green)-----
DEF RBox1 Object {
    size 0.02 0.012 0.02
    color 0.25 1 0.50
    texture ImageTexture { url "4.png" }
    stiffness 500
    bumpTexture ImageTexture {url "bump_6.png" }
    bumpHeight 0.005
    mass 1
    position 0.04184 0.0365 0.03566 }
DEF RBox2 Object {
    size 0.02 0.012 0.02
    color 0.35 1 0.05
    texture ImageTexture { url "5.png" }
    stiffness 100
    bumpTexture ImageTexture {url "bump_0.png" }
    bumpHeight 0.005
    mass 0.5
    position 0.07062 0.0365 0.05232 }
DEF RBox3 Object {
    size 0.02 0.012 0.02
    color 0.10 0.50 0.10
    texture ImageTexture { url "6.png" }
    stiffness 100
    bumpTexture ImageTexture {url "bump_3.png" }
    bumpHeight 0.005
    mass 0.3
    position 0.09946 0.0365 0.06897 }
DEF RBox4 Object {
    size 0.026 0.014 0.026
    color 0.25 1 0.50
    texture ImageTexture { url "10.png" }
    stiffness 600
```

```
        bumpTexture ImageTexture {url "bump_1.png" }
        bumpHeight 0.005
        mass 0.5
        position 0.04184 0.0675 0.03566 }
DEF RBox5 Object {
    size 0.026 0.014 0.026
    color 0.35 1 0.05
    texture ImageTexture { url "11.png" }
    stiffness 500
    bumpTexture ImageTexture {url "bump_4.png" }
    bumpHeight 0.005
    mass 0.2
    position 0.07062 0.0675 0.05232 }
DEF RBox6 Object {
    size 0.026 0.014 0.026
    color 0.10 0.50 0.10
    texture ImageTexture { url "12.png" }
    stiffness 100
    bumpTexture ImageTexture {url "bump_4.png" }
    bumpHeight 0.005
    mass 0.4
    position 0.09946 0.0675 0.06897 }

# -----
#     shelves.wrl
# -----
#VRML V2.0 utf8

# ----- Shelves on the left side -----
DEF LUpper Shelf {
    size 0.1 0.001 0.04
    position -0.07062 0.06 0.05232
    rotation 0 1 0 0.523 }
DEF LLower Shelf {
    size 0.1 0.001 0.04
    position -0.07062 0.03 0.05232
    rotation 0 1 0 0.523 }
DEF LTable Shelf {
    size 0.1 0.002 0.06
    position -0.06562 -0.04 0.06098
    rotation 0 1 0 0.523 }

# ----- Shelves on the right side -----
DEF RUpper Shelf {
    size 0.1 0.001 0.04
    position 0.07062 0.06 0.05232
    rotation 0 1 0 -0.523 }
DEF RLower Shelf {
    size 0.1 0.001 0.04
    position 0.07062 0.03 0.05232
    rotation 0 1 0 -0.523 }
DEF RTable Shelf {
    size 0.1 0.002 0.06
    position 0.06562 -0.04 0.06098
    rotation 0 1 0 -0.523 }

# -----
#     CVE.wrl
# -----
#VRML V2.0 utf8
Import {
    url [
        "Wall.wrl"
        "frontWall.wrl"
        "Floor.wrl"
        "Ceiling.wrl" ] }
DEF ROOM Group {
    children [
# --- Lightening ---
```

```
DirectionalLight {      # from above
    direction 0 -1 0
    intensity 0.5
    ambientIntensity 2 }
DirectionalLight {      # from right side
    direction -1 0 0
    intensity 0.3
    ambientIntensity 0.3 }
DirectionalLight {      # from left side
    direction 1 0 0
    intensity 0.3
    ambientIntensity 0.3 }
SpotLight {             # left inclined side down
    position -0.04 -0.02 0.15
    orientation 0 1 0 0.523
    intensity 2
    ambientIntensity 2
    coneHeight 0 }
SpotLight {             # left inclined side up
    position -0.04 0.06 0.15
    orientation 0 1 0 0.523
    intensity 2
    ambientIntensity 2
    coneHeight 0 }
SpotLight {             # left side
    position -0.06 0.06 0.1
    orientation 0 1 0 1.57
    intensity 2
    ambientIntensity 2
    coneHeight 0 }
SpotLight {             # right inclined side down
    position 0.04 -0.02 0.15
    orientation 0 1 0 -0.523
    intensity 2
    ambientIntensity 5
    coneHeight 0 }
SpotLight {             # right inclined side up
    position 0.04 0.06 0.15
    orientation 0 1 0 -0.523
    intensity 2
    ambientIntensity 5
    coneHeight 0 }
SpotLight {             # right side
    position 0.06 0.06 0.1
    orientation 0 1 0 -1.57
    intensity 2
    ambientIntensity 2
    coneHeight 0 }
SpotLight {             # front
    position 0 0.1 0.15
    orientation 1 0 0 -0.52
    ambientIntensity 1
    intensity 0.5
    coneHeight 0 }

# --- Fond ---
    USE WALL
# --- Front ---
    Transform {
        translation 0 0 0.15
        children USE FRONTWALL }
# --- Left inclined wall ---
    Transform {
        translation -0.09 0 0
        center 0.07 0 0
        rotation 0 1 0 0.523
        children [
            USE WALL
# --- Left side wall ---
    Transform {
```

```

                                translation -0.095 0 0
                                center 0.025 0 0
                                rotation 0 1 0 1.047
                                children USE WALL } ] }
# --- Right inclined wall ---
    Transform {
        translation 0.09 0 0
        center -0.07 0 0
        rotation 0 1 0 -0.523
        children [
            USE WALL
# --- Right side wall ---
    Transform {
        translation 0.095 0 0
        center -0.025 0 0
        rotation 0 1 0 -1.047
        children USE WALL } ] }
# --- Floor ---
    Transform {
        translation 0 -0.1 0.1
        children USE FLOOR }
# --- Ceiling ---
    Transform {
        translation 0 0.1 0.1
        children USE CEILING } ] }

# -----
# world.wrl
# -----
#VRML V2.0 utf8
Import {
    url [
        "objects.wrl"
        "shelves.wrl"
        "urn:inet:reachin.se:/bin/UserInterface" ] }
DEF DISPLAY Display {
    trackerUrls [
        "LeftAvatar.wrl" # blue [0]
        "RightAvatar.wrl" ] # green [1]
    proxyRadius 0.004
    stylus Shape {}
    children [
        DEF WORLD Transform {
            translation 0 0 -0.05
            children [
                CVE {
                    gravity 1
                    rubberStiffness 1500
                    shelves [
                        USE LUpper
                        USE LLower
                        USE LTable
                        USE RUpper
                        USE RLower
                        USE RTable ]
                    objects [
                        USE LBox1
                        USE LBox2
                        USE LBox3
                        USE LBox4
                        USE LBox5
                        USE LBox6

                        USE RBox1
                        USE RBox2
                        USE RBox3
                        USE RBox4
                        USE RBox5
                        USE RBox6 ] } ] ] } ] }
```



```
DEF PS PythonScript {
  references [ USE DISPLAY ]
  url "python:
display, = references
di      = getDeviceInfoTop()
devices = di.devices.get()
cnt     = 0

for d in devices:
  if d.name() == \"PhantomDevice\":
    cnt = cnt + 1
  tracker0 = display.trackers.get()[0]
  tracker0.proxyRadius.set( 0.006 )
  if cnt > 1:
    tracker1= display.trackers.get()[1]
    tracker1.proxyRadius.set( 0.006 )
"
}
```


Bilaga 2 – Header-filer

```
//-----  
//                                     CVE.h  
//-----  
  
# ifndef CVE_H  
# define CVE_H  
# include <Group.h>  
# include <Vrml.h>  
# include <Fields.h>  
# include <HapticsDevice.h>  
# include "Object.h"  
# include "Shelf.h"  
  
using namespace Reachin;  
  
class CVE : public Group {  
  
protected:  
    // Holder for all the objects in the world  
    struct MFObject : public TypedMFNode< Object > {  
        CVE *cve;  
    };  
    friend class MFObject;  
  
    // Holder for all the shelves in the world  
    struct MFShelf : public TypedMFNode< Shelf > {  
        CVE *cve;  
    };  
    friend class MFShelf;  
  
    // Composes together the objects, the shelves, the room and  
    // the "rubberbands" to one MF-node  
    struct WorldComposer : public EvaldFField< WorldComposer,  
                                                MFNode,  
                                                SFNode,  
                                                MFNode,  
                                                MFObject,  
                                                MFShelf    > {  
  
        void evaluate( SFNode    *room,  
                      MFNode    *visual_bands,  
                      MFObject  *objects,  
                      MFShelf   *shelves );  
    };  
  
    struct Rubberband {  
        CVE *cve;  
        HapticsDevice *haptic_device;  
  
        // holder for the endpoints of the rubberband.  
        // The reset function sets both endpoints to zero,  
        // to make the rubberband not showing  
        struct RubberbandComposer : public ComposeVec3f {  
            inline void reset() {  
                while ( inputs.size() ) inputs[0]->unroute( this );  
                value.resize( 2 );  
                value[0] = value[1] = Vec3f( 0, 0, 0 );  
                touch();  
            };  
        };  
    };  
  
    // keeps track of what to do when the button on the haptic device  
    // is pressed down or released  
    struct ButtonTracker : public Dependent< EvaldFField< ButtonTracker,  
                                                Field,
```

```

                                                                    SFBool > > {
    Rubberband *rubberband;
    Object      *hooked_object;
    mgFloat     rubber_length_glob_coord;
    Vec3f       grasp_translation;
    virtual void evaluate( SFBool *button_down );
};
friend class ButtonTracker;

// The position of the haptic device in cve coordinates.
// We require that input[0] is the global position
// of the haptic device.
struct HapticPosition : public EvaldFField< HapticPosition,
                                           SFVec3f,
                                           SFVec3f > {
    Matrix4f matrix_glob_to_cve;
    virtual void evaluate( SFVec3f *hd_pos_glob_coord ) {
        value = matrix_glob_to_cve * hd_pos_glob_coord->get();
    };
};

// The last accumulated matrix for CVE,
// saved in the collision traversal.
Matrix4f last_accumulated_forward;

Rubberband();

    auto_ptr< RubberbandComposer > rubberband_composer;
    auto_ptr< ButtonTracker      > button_tracker;
    auto_ptr< HapticPosition     > haptic_position;
    auto_ptr< SFNode             > shape;
}; // slut på Rubberband

// If we have a hooked object, the collider adds a force operator
// for the rubberband
struct Collider : public Group::Collider {
    virtual void collide( CollisionState *cs );
};
friend class Collider;

// The force operator for the rubberband
struct RubberForce : binary_function< Vec3f, Vec3f, mgFloat > {
    Time      last_time;
    mgFloat   last_weight;
    Object    *object;
    Vec3f     grasp_pos_glob_coord;
    mgFloat   object_mass;
    mgFloat   initial_rubber_length_glob_coord;
    Motion    motion;
    int       stiffness;
    Vec3f     operator() ( const Vec3f &hd_pos_glob_pos,
                          const mgFloat &weight );
};

public:

    CVE();

    static const Interface interface;

    auto_ptr< MFOBJECT > objects;
    auto_ptr< MFShelf  > shelves;
    auto_ptr< SFFloat  > gravity;
    auto_ptr< SFInt32  > rubber_stiffness;

    bool functional;

protected:
```

```
    auto_ptr< SFNode          > room;
    auto_ptr< ComposeNode    > visual_bands;
    auto_ptr< WorldComposer  > world_composer;

    vector < Rubberband*    > rubberbands;

    mgFloat wall_height;
    mgFloat wall_width;
    mgFloat inc_wall_width;

    DEFMap defmap;
    Vec3f avatarCollisionDetect( Rubberband *rubbl, Rubberband *rubbl2 );
    void objectCollisionDetect();
    void shelfCollisionDetect();
    void wallCollisionDetect();
};
#endif //CVE_H

//-----
//                               Object.h
//-----

# ifndef OBJECT_H
# define OBJECT_H

# include <Vrml.h>
# include <Shape.h>
# include <Dynamic.h>
# include "Shelf.h"

using namespace Reachin;

class Object : public Dynamic {

public:

    Object();

    static const Interface interface;

    auto_ptr< SFVec3f    > size;
    auto_ptr< SFRGB     > color;
    auto_ptr< SFRGB     > specular_color;
    auto_ptr< SFTexture  > texture;
    auto_ptr< SFTexture  > bump_texture;
    auto_ptr< SFFloat    > stiffness;
    auto_ptr< SFSurface  > surface;
    auto_ptr< SFFloat    > bump_height;
    auto_ptr< SFFloat    > gravity;

    vector < Object *    > objects_above;

    bool grasped;
    bool second_grasp;
    bool coll_above;
    bool coll_below;
    bool on_floor;

    mgFloat orig_mass;
    RGB     orig_color;

    Object *obj_above;
    Object *obj_below;
    Shelf  *on_shelf;
    Shelf  *partly_on_shelf;

    void updateObjectsAbove();
    void composeObject();
    void moveComposedObject();
};
```

```
void    moveObjects();

Object* hookObject();
Object* releaseObjectFunctional();
Object* releaseObjectSubtle();
Object* getLowestObject();
Object* getHookedObject();
mgFloat getTotMass();
Vec3f   getMomentum();
Vec3f   getTranslation();
void    setMomentum( Vec3f m );
void    setTranslation( Vec3f m );

protected:

    DEFMap defmap;
};

# endif

//-----
//                               Shelf.h
//-----

# ifndef SHELF_H
# define SHELF_H

# include <Vrml.h>
# include <Shape.h>
# include <Transform.h>

using namespace Reachin;

class Shelf : public Transform {

public:

    Shelf();

    static const Interface interface;

    auto_ptr< SFVec3f   > size;
    auto_ptr< SFRGB    > color;
    auto_ptr< SFTexture > texture;

protected:
    DEFMap defmap;
};

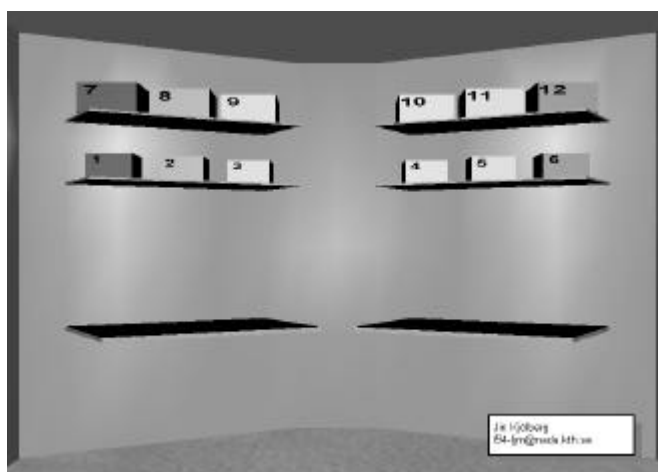
# endif
```

Bilaga 3 – Uppgiftslydelse för försökspersoner (blå avatar)

Information

Du ska nu delta i en användarutvärdering av en haptisk samarbetsapplikation. Syftet med utvärderingen är att utvärdera systemet och inte användaren. Du har rätt till att när som helst avbryta deltagandet om du av någon anledning inte önskar att fortsätta.

Du kommer att få ett antal uppgifter som du tillsammans med en samarbetspartner ska genomföra i två olika miljöer. Båda miljöerna innehåller ett rum med sex hyllor samt tolv numrerade objekt.



Din grafiska representation i miljön är en blå sfär. Din samarbetspartners grafiska representation är en grön sfär.

Det är bra om du hela tiden försöker prata högt för dig själv och med din samarbetspartner, om vad du gör och upplever i miljön. Dina reflexioner och spontana kommentarer kommer att vara till stor hjälp i utvärderingen av systemet.

Under genomförandet kommer bildskärmen att videofilmas och allt tal spelas in. Detta kommer endast att användas i forskningssyfte och inte lämnas ut till någon. Om du av någon anledning inte accepterar detta, var god säg till.

När du och din samarbetspartner känner er redo kan ni vända blad och börja genomföra uppgifterna. Ni bestämmer själva när ni är klara med en uppgift.

Var god säg till när ni är klara med Uppgift 5, så att miljön kan bytas ut inför nästa uppgift.

Uppgift 1

Försök att hitta det blå objekt som ni upplever som tyngst.
Skriv objektets nummer i rutan till höger:

Uppgift 2

Placera två objekt ovanpå varandra och ge dem till din samarbetspartner.

Ta sedan emot objekten din samarbetspartner ger dig och placera dem på valfri plats.

Om ni tappar objekten får ni börja om.

Uppgift 3

a) Försök hitta det objekt som ni upplever har mest ojämn yta.
Skriv objektets nummer i rutan till höger:

b) Försök hitta det objekt som ni upplever har jämnast yta.
Skriv objektets nummer i rutan till höger:

c) Försök hitta två objekt med lika jämna eller ojämna ytor.
Skriv objektens nummer i rutan till höger:

Uppgift 4

Placera fyra objekt ovanpå varandra, vartannat ska vara blått respektive grönt.

Lyft sedan objekten tillsammans med din samarbetspartner över till en hylla på andra sidan av rummet.

Om ni tappar objekten får ni börja om.

Uppgift 5

Föreställ er följande:

- Varje låda representerar olika sorters information.
 - Lådans siffra används för att beskriva informationens värde.
 - Ett informationsobjekt kan vara enkelt (en låda) eller sammansatt (flera lådor).
 - Ett sammansatt informationsobjekt skapas genom att placera lådor ovanpå varandra. Summan av lådornas siffror är informationsobjektets totala värde.
 - Det är viktigt att inte tappa någon information.
- a) Ta emot det informationsobjekt som din samarbetspartner ger dig och placera det på valfritt ställe.

Skapa sedan ett informationsobjekt med värde 10 och överlämna det till din samarbetspartner.

- b) Du och din samarbetspartner ska tillsammans gruppera informationen så att
- informationsobjekt på hyllorna till vänster har värde mindre än 10 och
 - informationsobjekt på hyllorna till höger har värde större än 10.

Skapa gärna nya sammansatta informationsobjekt.

Var god säg till för byte av miljö.

Uppgift 6

Var god fyll i följande uppgifter om dig själva.

Ålder: _____ år

Kön: Kvinna Man

Utbildning/Yrke: _____

Är detta första gång du använder haptiska gränssnitt? Ja Nej

Uppgift 7

Försök att hitta det gröna objekt som ni upplever som tyngst.
Skriv objektets nummer i rutan till höger:

Uppgift 8

Ta emot objekten din samarbetspartner ger dig och placera dem på valfri plats.

Placera sedan två objekt ovanpå varandra och ge dem till din samarbetspartner.

Om ni tappar objekten får ni börja om.

Uppgift 9

Placera fyra objekt ovanpå varandra, vartannat ska vara blått respektive grönt.

Lyft sedan objekten tillsammans med din samarbetspartner, över till en hylla på andra sidan av rummet.

Om ni tappar objekten får ni börja om.

Uppgift 10

Försök att åter föreställa er att objekten representerar information.

Skapa ett sammansatt informationsobjekt av tre enkla informationsobjekt och överlämna det till din samarbetspartner.

Ta sedan emot det sammansatta informationsobjekt din samarbetspartner ger dig. Dela upp informationsobjektet i enkla objekt och sortera dem i stigande värde från vänster till höger, på översta hyllan till vänster. (Lägst värde längst till vänster på hyllan.)

*Tack för din medverkan så långt.
Den praktiska delen av utvärderingen är nu färdig.*

Utvärdering: _____ SF FS

Bilaga 4 – Intervjufrågor

Har ni några spontana reaktioner eller kommentarer till det ni har gjort?

Ni testade två miljöer eller två versioner av miljön. Upplevde ni att det var någon skillnad mellan miljöerna?

Vad var skillnaderna?

Hur upptäckte ni skillnaderna?

Hur var det att arbeta i de olika miljöerna?

Var det något ni förvånades av?

Betodde objekten sig så som ni förväntade att de skulle göra?

Var det något beteende ni skulle vilja ändra på?

Kändes det naturligt att arbeta med objekten?

Kändes det realistiskt?

Var det något i någon av miljöerna som ni irriterades över?

Var det fysiskt tröttnande eller jobbigt att arbeta i miljöerna?

Var det någon skillnad mellan de olika miljöerna?

Var det något med utrustningen ni var speciellt nöjd med?

Var det något av utrustningen som ni irriterades över?

Kändes det som om ni behövde anpassa er efter utrustningen?

Vad tyckte ni om att använda stereoglasögon?

Hur tyckte ni att det gick att lösa uppgifterna?

Var det roligt?

Var det svårt eller lätt?

Var det några uppgifter som var svårare än andra?

Kände ni att ni kunde lösa uppgifterna så att ni var nöjda?

Var det skillnad i de olika miljöerna?

Kändes det som om ni var tillsammans i miljön?

Hur hade det varit att arbeta ensam?

Hade det varit bättre att vara ensam?

Kändes det tryggt att lämna ifrån sig ett objekt?

Var ni säkra på att den andra skulle ta emot objektet?

Kunde ni lita på att den andra tog emot?

Var det skillnad mellan de olika miljöerna?

Använde ni något sätt att kolla om den andra hade kontroll över objektet innan ni släppte det själv?

Påverkades detta av miljön?

När ni skulle ta emot ett objekt, kändes det som om ni hade kontroll över objektet?

Var det skillnad i de två miljöerna?

Var det obehagligt att ta emot objekten?

Upplevdes det säkrare att ge eller ta emot?

Påverkades detta av vilken miljö ni jobbade i?

Var det lättare att ge eller ta emot?

Påverkades detta av vilken miljö ni jobbade i?

Om ni tänker er en situation i framtiden där det är möjligt att kommunicera på liknande sätt på distans, till exempel i en community, chat eller liknande.

Skulle ni då vilja ge ifrån er eller ta emot ett mycket viktigt objekt? Det skulle kunna vara information eller något som kunde gå sönder eller som inte fick tappas bort.

Skulle ni vilja ta emot ett objekt?

Hade någon av miljöerna varit bättre eller sämre lämpad för en sådan situation?

