# TWLT 15

# Interactions in Virtual Worlds

PROCEEDINGS OF THE FIFTEENTH
TWENTE WORKSHOP ON LANGUAGE TECHNOLOGY

MAY 19-21, 1999
ENSCHEDE, THE NETHERLANDS

**Anton Nijholt, Olaf Donk**
**and Betsy van Dijk (eds.)**

CIP GEGEVENS KONINKLIJKE BIBLIOTHEEK, DEN HAAG

Nijholt, A., Donk, O.A., Van Dijk, E.M.A.G.

*Interactions in Virtual Worlds*
Proceedings Twente Workshop on Language Technology 15
A. Nijholt, O.A. Donk, E.M.A.G. van Dijk (eds.)
Enschede, Universiteit Twente, Faculteit Informatica

ISSN 0929–0672

trefwoorden: virtual worlds, virtual reality, agent technology, presence, talking faces, VRML, speech recognition, speech generation, human-computer-interaction, social structures, WWW



Book orders:
Ms. A. Hoogvliet
University of Twente
Dept. of Computer Science
P.O. Box 217
NL 7500 AE Enschede
tel: +31 53 4893680
fax: +31 53 4893503
Email: hoogvlie@cs.utwente.nl

Druk- en bindwerk: Reprografie U.T. Service Centrum, Enschede

# Preface

TWLT is an acronym of Twente Workshop(s) on Language Technology. These workshops on natural language theory and technology are organized by the Parlevink Reasearch Project, a language theory and technology project of the Centre of Telematics and Information Technology (CTIT) of the University of Twente, Enschede, The Netherlands. For each workshop proceedings are published containing the papers that were presented. See next page for an overview of the previous TWLT workshops.

TWLT 15 has been organized by a program committee consisting of Anton Nijholt (chair, The Netherlands), Niels Ole Bernsen (Denmark), Philip Brey (The Netherlands), James Lester (USA), Stephen Matsuba (Canada), Pierre Nugues (France) and Oliviero Stock (Italy). The workshop is endorsed by SIGDIAL and is organized in the context of the U-Wish project of the Dutch national Telematics Institute.

TWLT 15, Interactions in Virtual Worlds, is a 3-day workshop on interactions in virtual reality (VR) environments. Contributions were invited on theoretical, empirical, computational, experimental, anthropological or philosophical approaches to VR environments. In this workshop the emphasis is on VR environments that provide means for interacting with the objects in the environment, with embedded information sources and services (possibly represented as agents) or with other users and visitors of the environment.

In recent years the computer science and the cognitive ergonomics communities have discovered and are exploring the possibilities of VR in user interfaces. In order to visualize available information and to provide users access to this information, virtual environments have been built such that users can find and explore information, communicate with other users and communicate with objects and agents in the virtual worlds. In this workshop new forms of interactivity are explored. Topics are, amongst others, the role of agent technology and of speech and language in virtual environments. How can we navigate in VR using speech and language, how can we model multimodal access to such environments, how can we communicate with other humans and with artificial agents in the VR environment, etc.

A workshop is the concerted action of many people. We are grateful to the authors and the organisations they represent, for their efforts and contributions. In addition we would like to mention here the people whose efforts have been less visible during the workshop proper, but whose contribution was evidently of crucial importance. Charlotte Bijron and Alice Hoogvliet took care of the administrative tasks (registration, hotel reservations, etc.). Furthermore we are most grateful to the members of the program committee and to Rieks op den Akker, Bert Lenting, Danny Lie, Charles van der Mast, Mannes Poel, Boris van Schooten and Roel Vertegaal for their support in the refereeing process.

Anton Nijholt, Olaf Donk and Betsy van Dijk May 1999

## Previous TWLT workshops

Previous TWLT workshops were

| | |
|---|---|
| TWLT1, | *Tomita's Algorithm: Extensions and Applications*. 22 March, 1991. |
| TWLT2, | *Linguistic Engineering: Tools and Products*. 20 November, 1991. |
| TWLT3, | *Connectionism and Natural Language Processing*. 12 and 13 May 1992. |
| TWLT4, | *Pragmatics in Language Technology*. 23 September, 1992. |
| TWLT5, | *Natural Language Interfaces*. 3 and 4 June, 1993. |
| TWLT6, | *Natural Language Parsing*. 16 and 17 December, 1993. |
| TWLT7, | *Computer-Assisted Language Learning*. 16 and 17 June 1994. |
| TWLT8, | *Speech and Language Engineering*. 1 and 2 December 1994. |
| TWLT9, | *Corpus-based Approaches to Dialogue Modelling*. 9 June, 1995 |
| TWLT10, | *Algebraic Methods in Language Processing*. 6-8 December, 1995 |
| TWLT11, | *Dialogue Management in Natural Language Systems*. 19-21 June, 1996 |
| TWLT12, | *Automatic Interpretation and Generation of Verbal Humor*. 11-14 September 1996 |
| TWLT13, | *Formal Semantics and Pragmatics of Dialogue, Twendial'98*. 13-15 May 1998 |
| TWLT14, | *Language Technology in Multimedia Information Retrieval*. 7-8 December 1998 |

For the contents of the previous proceedings, please consult the last pages of this volume.

## Contents


*Let's-Improvise-Together* 1
Riccardo Antonini (University of Rome)

*Design and the Social Ontology of Virtual Worlds* 5
Philip Brey (Twente University)

*Investigating Navigation and Orientation within Virtual Worlds* 13
Dimitrios Charitos (University of Athens), Alan H. Bridges (University of Strathclyde) and Drakoulis Martakos (University of Athens)

*VR-Spider for (non) Immersive WWW navigation* 23
M. Cibelli, G. Costagliola, G. Polese and G. Tortora (University of Salerno)

*Steps toward Learning in Virtual World Cyberspace: TheU Virtual University and BOWorld* 31
Bruce Damer, Stuart Gold (The Contact Consortium), Jan A. W. de Bruin (Tilburg University) and Dirk-Jan G. de Bruin

*Virtual Worlds Applications* 43
Russell Eames (Microsoft Research, Seattle)

*Creating Virtual Worlds with a Graspable User Interface* 45
Hauke Ernst, Kai Schäfer and Willi Bruns (University of Bremen)

*Virtual Reality Interface for the World Wide Web* 59
Denis Gracanin and Kecia E. Wright (University of Southwestern Louisiana)

*The Usability of Interacting with the Virtual and the Real in COMRIS* 69
Geert de Haan (IPO, Eindhoven University of Technology)

*ECHOES: A Collaborative Virtual Training Environment* 81
G. M. P. O'Hare, A. J. Murphy, T. Delahunty and K. Sewell (University College Dublin)

*Why Bill was killed – understanding social interaction in virtual worlds* 95
Mikael Jakobsson (Umeå University)

*Natural Interaction with Pedagogical Agents in Virtual Worlds* 103
W. Lewis Johnson (Marina del Rey)

*Modelling Interaction to Inform Information Requirements in Virtual Environments* 107
Kulwinder Kaur Deol, Alistair Sutcliffe and Neil Maiden (City University London)

*Virtual Campus Project – A Framework for a 3D Multimedia Educational Environment* 115
Rainer Kuhn and Sigrún Guðjónsdóttir (University of Karlsruhe)

*Natural Language Generation in Multimodal Learning Environments: Lifelike Agents and 3D Animated Explanation Generation* 125
James C. Lester (North Carolina State University)

*Speaking Spaces: Virtual Reality, Artificial Intelligence and the Construction of Meaning* 127
Stephen N. Matsuba (The VRML Dream Company, Vancouver)

*Verbal and Written Interaction in Virtual Worlds – Some application examples* 137
Pierre Nugues (ISMRA-Caen)

*The Twente Virtual Theatre Environment: Agents and Interactions* 147
Anton Nijholt (University of Twente)

*When Worls Collide – Interactions between the Virtual and the Real* 165
Sandy Ressler, Brian Antonishek, Qiming Wang, Afzal Godil and Keith Stouffer
(National Institute of Standards and Technology)

*EISCAT Virtual Reality Training Simulation: A Study on Usability and Effectiveness* 171
Lakshmi Sastry and D. R. S. Boyd (Rutherford Appleton Laboratory)

*"Males say 'blue,' females say 'aqua,' 'sapphire,' and 'dark navy'"* 183
*The Importance of Gender in Computer-Mediated Communication*
Frank Schaap (University of Amsterdam)

*Modelling Interaction in Virtual Environments using Process Algebra* 195
Boris van Schooten, Olaf Donk and Job Zwiers (University of Twente)

*Presence: Interacting in Virtual Reality?* 213
Martijn J. Schuemie and Charles A. P. G. van der Mast (Delft University of Technology)

*Towards an Understanding 3D Virtual Reality Architectural Design System* 219
Jarke J. van Wijk, Bauke de Vries and Cornelius W. A. M. van Overveld
(Eindhoven University of Technology)

*Spoken Language Interaction with a Virtual World in the MUeSLI Multimodal 3D Retail System* 225
Peter J. Wyard and Gavin E. Churcher (BT Laboratories, Ipswich, Suffolk)

*Real Time Gesture Based 3D Graphics User Interface for CAD Modelling System* 233
J. M. Zheng, K. W. Chan and I. Gibson (University of Hong Kong)

## Sponsors and support

We gratefully acknowledge help from:

- Shell
- NWO
- Switch Automatisering
- Research Project Parlevink
- Neuro Fuzzy Centrum
- TNO-MET
- University of Twente
- Carp Technologies

## Organisation

TWLT 15 is organised by:

- Research Project Parlevink
- Centre for Telematics and Information Technology
- University of Twente

# Let's-Improvise-Together

Riccardo Antonini[1]

[1] Consorzio Roma Ricerche, Università di Roma "Tor Vergata", Via O. Raimondo, 8,
00173 Rome, Italy
Riccardo.Antonini@UniRoma2.it
Amusement Project i3 Esprit n. 25197

## Abstract

In the Amusement project [7] one of the goals is the integration between different aspects of the interaction at distance through Virtual Worlds. During the process of pursuing that goal we experienced the need of language, vision and music integration in a multiuser distributed virtual environment. Hence we implemented a game 'Let's-Improvise-Together' as testbed for our views on integration. In the following a conceptual description of the game is illustrated, the formal description can be found in [2].

## Goals of the Game

The creators of 'Let's-Improvise-Together' adhere to the idea that while there is a multitude of online games now available in cyberspace, it appears that relatively few are focused on providing a positive, friendly and productive experience for the user [6]. Producing this kind of experience is one the goals of our Amusement Project.

To this end, the creation of 'Let's Improvise Together' has been guided by dedication to the importance of three themes:

- the importance of cooperation,
- the importance of creativity, and
- the importance of emotion.

The **ability to cooperate** is a valuable skill, but we humans do not always do what is best for us, a fact evidenced by the large number of computer games which pander to the opposite end of the spectrum of human nature.

Moreover, while many games are tightly ruled-based, on the other hand improvisation is one of the most proving field of investigation on **creativity** as testified, for example, by the fact that it has been taken as "leitmotif" of the "Doors of Perception 5" Conference, among others [8].

High **emotional impact**. To be clear: the emotional impact is on the players themselves, not on the avatars, that, in our cases, have a limited, if any, autonomous emotional capability (of course, they are autonomous under other aspects [1]). Moreover, in our games the emotion is based also on the fact of being together.

## Description of the Game

The avatar arrives in a certain area where there are many sound-blocks/objects. Or he may add sound "property" to existing ones. He can add new objects at will. Each object may represents a different sound, they do not have to though. The avatar walks around and chooses which objects he likes. Makes copies of these and add sounds or change the sounds on existing ones, then with all of the sound-blocks combined make his personalized "instrument".

Now any player can make sounds on the instrument by approaching or bumping into a sound-block. The way that the avatar makes sounds on the instrument can vary.

At the end of the improvising session, the 'composition' will be saved on the instrument site, along with the personalized instrument. In this way, each user of the Amusement Center will leave behind him a unique instrumental creation, that others who visit the Center later will be able to play on and listen to. Execution of the virtual compositions, along with their relevant 3D environment are experimentally broadcasted via Italian RAI3 (freq. 11.534 GHz pol. V) on Hot Bird 1, 13 degree east, and can be received and experienced on PCs equipped with a commercial data broadcast receiver board. There are, among others, more than four thousand receiver boards in italian schools. The idea of broadcasting interactive virtual worlds has been pioneered by Toshio Iwai [4] and Lili

Cheng [4]. The implementation in the Amusement project is described in [1,3]

The fully creative experience of making a new instrument can also be obtained connecting via Internet to Active Worlds world ‘Amuse’ and ‘Amuse2’.

Animated colorful sounding objects can be assembled by the user in the Virtual Environment as a sort of sounding instrument. We refrain here deliberately from using the word ‘musical’ instrument, because the level of control we have on the sound in terms of rythm and melody, among other parameters, is very limited. It resembles instead, very closely, to the primitive instruments used by humans in some civilizations or to the experience made by children making sound out of ordinary objects. The children are very good at improvisation and very interested at it see for example [5]. The dimension of cooperation is of paramount importance in the process of building and using the virtual sounding instrument. The instrument can be built on ones own effort but preferably by a team of cooperating users. The cooperation has as an important corolary: the sharing of the same experience either synchronically or diachronically. The shared experience finds its permanence in the collective memory of the sounding instruments built. The sounding instrument can be seen also as a virtual sculpture, indeed this sculpture is a multimedial one. The objects have properties that ranges from video animation to sound to virtual physical properties like solidity. The role of the user representation in the Virtual World, called avatar, is important because it conveys, among other things, the user’s emotions. It is worth pointing out that the Avatar has no emotions on its own but it simply expresses the emotions of the user behind it. In a way it could be considered a sort of actor performing the script that the user gives it in real-time while playing.

The natural **language** is used for the cooperative task of building and experiencing the virtual instrument, but also body language and facial expression are used by our avatars. The user, when building the virtual instrument, uses some skills that, in our view, are closely related to the ones used for the expression by means of natural language. We cannot claim to have made a new language in ‘Let’s Improvise Together’, nevertheless, the users, while playing, rearanges not only the layout of the virtual objects and their properties, but also the rules of their use, creating some sort of conventions for their expression.

We tried to organize the game as an improvisation rather than a tight rule based game. The improvisation happens in two steps. First the users organize their virtual space. That means, in practice, that they buid their own virtual instrument. Hence they play with the instrument. The two steps are iterated in a potentially endless process. Since no part of the expression tools is completely determined, the user can, if not build a new language, at least personalyze the existing one.

The **vision** is used for the perception of the 3D objects in the virtual space. But vision is also used for experiencing the virtual instrument. The vision is integrated by some sense of physical feedback given by the objects, for example an object can stop the avatar from going through it and possibly make a sound when bumped into it.

The game, ‘Let’s Improvise Together,’ is mostly related to the experience of sound rather than the experience of **music**. Moreover, the sound is not important perse but as an additional dimension to the experience in Virtual Worlds. We have already said that we have deliberatedly circumscribed the acoustic experience to generic sound only. Nevertheless it is, in principle, possible to have real music pieces as properties of the object constituting the virtual instrument. We have not studied this more complex case thorouthfully up to now though.

The other important element of the integration is related to the memory of the experience left by the user into the Virtual World. The new layout is explored and experienced. The layout can be regarded as a permanent editable memory.

## References


[1] Antonini R.. Evolvable Believable Agents for Broadcasted Interactive Game Shows. In *Proceedings of AISB '99 - Edimburgh - apr 99* , 1999, http://www.darmstadt.gmd.de/mobile/aisb99/submissions.

[2] Antonini R.. Let's Improvise Together a testbed for a formalism in Language Vision and Sounds integration. In *Language, Vision & Music - Galway Ireland - aug 99* , 1999.

[3] Antonini R.. Broadcasting Digitally Generated Gameshows. In *2nd International Workshop on Presence* , 1999, http://privatewww.essex.ac.uk/~jfreem/programme.html

[4] Cheng L.. YORB. http://www.itp.tsoa.nyu.edu/~yorb/yorbNew.html.

[5] Iwai T.. UgoUgoLhuga. In *Fujii Television, Japan*, 1994, http://www.iamas.ac.jp/~iwai/ugolhu/index.html.

[6] KIM A. J.. Ritual reality. In *http://www.naima.com/resume.html* , 1998, http://www.naima.com/resume.html.

[7] Universidad Politecnica de Madrid, Consorzio Roma Ricerche, Ecole Superieure de Telecomunication Bretagne, Skydata S.p.A., Instituto Europeo de Transferencia de Tecnologia. Amusement - An International Virtual Space for Individual and Collective Presence and Interaction. In *i3 Esprit project 25197* , , http://www.i3net.org/i3projects.

[8] *DOORS OF PERCEPTION 5 - Play* , 1998, Http://www.doorsofperception.com/.

# Design and the Social Ontology of Virtual Worlds

Philip Brey
Fac. WMW / SW, University of Twente
P.O. Box 217, 7500 AE Enschede
The Netherlands
brey@wmw.utwente.nl

## ABSTRACT

This essay analyzes the ontological structure of institutional and other social entities in virtual environments. The emphasis is on institutional reality, which consists of entities (objects, events, etc.) like money, contracts and chess pieces, that are constituted in part through collective agreements. It is studied how institutional entities are constructed in virtual environments, how they relate to institutional entities in the real world, and how they are recognized by users. It is subsequently disucssed how designers of virtual environments may better account for the peculiar characteristics of institutional entities in design.

**Keywords:** Social Reality; Institutional Reality; Ontology; Virtual Environments; Language; Interpretation; Design.

## 1 INTRODUCTION

Ontology, the study of being, constitutes one of the main branches of philosophy. This paper contains an ontological investigation of virtual worlds, with particular attention paid to the construction of social and institutional reality in virtual worlds. It is argued that such an investigation is relevant for the design of virtual environments, because it can help explain how the identity of objects, actions and events in virtual environments is fixed and how users arrive at interpretations of entities they encounter in virtual environments. This, in turn, may contribute to the development of design practices that better take into account the interpretive practices and interests of users. Results presented in this essay are arrived at mainly through conceptual analysis and naturalistic observation.

The structure of the paper is as follows. In section 2, a basic ontology of the real world is proposed, which is based on an influential recent study by philosopher John Searle. Searle's ontology distinguishes between physical and social reality, and outlines basic characteristics of social reality as well as of institutional reality, which is part of social reality. This ontology provides the background for a subsequent ontological investigation of virtual environments in section 3. In section 3, a comparison is made between virtual entities (entities encountered in virtual environments) and real-world entities (entities encountered in the ordinary world), and it is argued that institutional entities, unlike other entities, can have literal existence in virtual environments. Subsequently, it is analyzed how institutional entities are constituted and recognized in virtual environments. In section 4, finally, some implications are drawn out for the design of virtual environments. Two principles for the design of institutional environments, the transparency principle and the intersubjectivity principle, are proposed and elaborated.

## 2 BASIC ONTOLOGY OF THE REAL WORLD

A basic ontology of the real world is an account of the basic types of things that exist in the world, classified according to their mode of existence. By a mode of existence, I mean the way in which something has come into reality and the manner in which it currently exists. One of the most influential ontologies of the real world has recently been presented by John Searle [1]. Searle's account is coherent and well-supported with examples, and is more complete than most other ontologies of the real world, since unlike many other ontologies it also tries to account for social reality, and not just for physical reality. Searle's organizing principle in arriving at his ontological distinctions is the relation of things to human interpretation of them. Searle argues that things may be wholly independent of human interpretation or thought, but they may also be partially constituted, to different degrees and in different ways, through human interpretation.

This organizing principle leads Searle to make a fundamental distinction between physical reality and social reality. *Social reality* is the set of all entities that are not genuinely objective but are the outcome of a process of social interpretation or construction. *Physical reality* is genuinely objective and includes entities that exist independently of our representations of them. Searle illustrates this distinction by pointing out the difference between physical and social facts. *Physical facts* includes such truths as that there are snow and ice near the summit of Mt. Everest, that apples grow on apple trees, and that there is electric lighting in many houses on the Western hemisphere. Searle is willing to admit that the *concepts* used in expressing physical facts are socially constructed; concepts like that of Mt. Everest, of a house and of the Western hemisphere constitute a particular way of representing physical reality, and this mode of representation is socially constructed. Yet, Searle denies that the *referents* of these concepts are also socially constructed. Rather, Mt. Everest, houses, trees and the Western hemisphere exist independently of our representations of them.

Searle contrasts the class of physical facts with the class of *social facts*, for which it is true that not just the concepts used in representing these facts, but also the facts themselves are socially constructed. Social facts all have in common that they are defined over human institutions, practices, and artifacts. They pose a paradox in being at the same time 'objective' (in the sense of being widely accepted and uncontroversial) and dependent on human representation. The class of social facts includes such facts as that Bill Clinton is a married man, that Beatrix is the queen of the Netherlands, that a bar of gold is worth a lot of money, that Harvard university offers a graduate degree program in physics, and that the curved object in my kitchen drawer is a corkscrew. These facts, Searle claims, seem to be objective in that there is (near-)universal agreement on them; they are accepted as uncontroversial and true in just the same way that physical facts are accepted as true.

Yet, Searle argues, it also seems to be true of these facts and the entities that play a role in them (such as marriage, relations of ownership, monetary value, universities, and friendship) are human constructs in a way that physical facts and entities are not. In particular, these facts and entities seem to be dependent on human representation or intentionality in a way that physical facts and entities are not. There is nothing intrinsic about the green paper bills that are used as money that determines their nature as money. Only when people start representing (intentionally using, accepting, believing in) such bills as money, intuitively, does it become a fact that these bills are money. On the other hand, Searle claims, it is intuitively true that facts such as the fact that hydrogen atoms include two protons or that Mount Everest is the highest mountain on earth continue to be facts even if there are no humans who represent hydrogen atoms as having two protons or Mount Everest as being the highest mountain.

How do social facts come into existence? Searle argues that, barring some exceptions, they come into existence through the *collective imposition of a function* on some object, event or action. For instance, it is now a fact that the Dutch Delta works constitute a barrier against floods, because this function has in the past been collectively imposed on them in Dutch society. Likewise, it is true that Bill Clinton is married, because by the performance of certain ceremonial acts, a certain functional relationship, that of being a husband, was imposed on him to another person, Hillary Rodham. Searle claims that the collective imposition of function is a *collective intentional act*, which is an act that is intentionally performed by a collective (e.g., Dutch society, or American society).

Searle distinguishes between two kinds of collectively imposed functions, which give rise to two different kinds of social facts. The first kind, consisting of ordinary collectively imposed functions, leads to *ordinary social facts,* which seem to apply mainly to *(material) artifacts*. Examples of such facts include the fact that devices of a certain form are screwdrivers (accounted for by the collective imposition of the function of driving screws on such devices), or the fact that the Delta works are a barrier against floods (accounted for by the collective imposition of the function of obstructing high tides on them). The second kind, called *status-functions,* leads to *institutional facts,* that constitute *institutional reality*. Such facts are normally created within the context of previously created human institutions. Examples include the fact that Bill Clinton is married, that dollar bills exist, that some people possess real estate, and that Paul McCartney is a former member of the Beatles.

Ordinary social facts and corresponding entities and properties come into existence when a function is collectively imposed on an object that is inherently able, in virtue of its physical constitution, to perform this function. Hence, the ontology of such social entities is accounted for by the fact that these entities are accepted by a society or group of having a certain function, and have the inherent feature of being physically able to perform this function. For example, an object is a screwdriver just in case *(a)* people have imposed on it the function of driving screws, and *(b)* it has the physical capacity to drive screws. Whenever these conditions hold, it becomes an uncontroversial, 'objective' fact that the object in question is a screwdriver.

There is a large class of social entities, however, that have collectively imposed functions, but that are not able to perform this function solely in virtue of their physical constitution. In such entities, the act of collective imposition of function attributes causal powers to these entities that these entities did not previously have. Functions of this sort are called *status functions*. Status functions are imposed when people collectively assign a *status* to some object. This collective assignment of status brings with it an agreement to consider or treat this entity *as if* it had inherent causal powers to perform this function. Such agreement, Searle claims, takes the form of a *constitutive rule*, which has the form "*X* counts as *Y* (in *C)*," where *X* defines the class of objects which qualify to be assigned a status, *Y* defines the status that is assigned, and *C* is any context which must be present for this status to hold. Thus, for example, undergoing the marriage ceremony (*C*) has made Bill Clinton (*X*) into a married man (*Y*). That this happened is because in American society, the constitutive rule applies that when a person undergoes a properly performed marriage ceremony, he or she turns into a married person. This constitutive rule exists because of a collective agreement (which is a collective intentional act) in American society that someone who undergoes a marriage ceremony acquires a certain status.

The social facts that are the result of the imposition of status functions are called *institutional facts*. They are called that because they are normally created within the context of human institutions, such as marriage, universities, and money. According to Searle, the great ontological difference between entities created by the assignment of status functions and entities created by the assignment of ordinary functions is that the former need not be able to physically perform the function that is imposed on them. Thus, for an entity to be money, it is necessary and sufficient that it is accepted by a collective to *be* money, whereas for an object to be a screwdriver, it must be accepted as such, *and* be physically able to drive screws. Yet, this raises an important question: How are entities able to live up to the status functions that are assigned to them if it is not required that they are physically able to perform the status function assigned to them?

Searle explains that for many entities, the imposition of status functions occurs under the expectation that the entity in question has inherent capacities that enable it to *live up to* its status function. For example, to qualify as a licensed driver, one has to take a written test and a road test, and the expectation is that individuals who pass the test have inherent capacities to drive well. The status granted to licensed driver has the consequence that they are permitted by society to operate a vehicle. Obviously, this status could also be attained without passing any tests, if a society chose to relax the conditions under which individuals are granted the status of a licensed driver. In such a society, the status of licensed driver would still bring with it the rights, privileges and duties that licensed drivers have under the system of law. Many licensed drivers in this society will, however, unlikely be able to live up to their duties on the road, as they are likely to lack the skills to drive well.

There are other status functions, however, for which the presence of certain intrinsic capacities seem to be a less important condition for their assignment to an entity. Almost any object, for example, can be made to have the status of money, if a society chose to make it so. Shells, gems, pieces of metal, pieces of paper and other items are all qualified to take the status function of money in a society. There are much less specific demands on the inherent capacities of an entity for it to be assigned the status of money, than for it to be assigned the status of, for example, a licensed driver. Searle discerns a scale from 'arbitrariness to reason' with respect to the inherent constitution of the entities that are assigned status functions. At one end, there are entities such as money, which can be of almost any form. At the other end, there are entities such as licensed drivers or surgeons or research institutions, which are required to display certain highly specific abilities to be assigned this status.

Many entities in the real world are institutional in nature. They include people (e.g., janitors, professors), physical objects (e.g., dollar bills, wedding rings, contracts, chess games), properties (e.g., being licensed, being under probation), events (weddings, parties, elections), and actions (trespassing, scoring, prohibiting). Importantly, language is also an institutional phenomenon. The marks that read "tree" can only refer to trees because it is collectively accepted that these marks have this meaning. Nonlinguistic symbols similarly derive their meaning from a collective imposition of a symbolizing function to them.

Social reality is always relative to a community or collective that engages in the imposition of functions. When there is agreement in a community on the assignment of functions, this community has a shared social reality that it collectively accepts to hold objectively. However, a community may be divided over the social reality it accepts, in which case some members accept a different social reality than others. For example, in the mid 1990s, a large group of

Northern Italians, led by Lega Norte frontman Umberto Bossi, declared Northern Italy an independent state named Padania. From then on, the existence of Padania had become a reality for Lega Norte, but not for the rest of Italians. This example is however not unusual. Every day, all of us are confronted with differences between our social ontology and those of others we encounter: people that accept different meanings for words than we do, that assign different functions to artifacts, that accept different constitutive rules for games, or that attribute different social statuses to people or events. So even though a large part of the social reality we accept is shared with others, a large part is also contested.

To conclude, Searle distinguishes between physical reality, which is independent of human interpretation, and social reality, which is usually constituted in part by the collective imposition of functions on things, actions or events. Social reality is subsequently divided into ordinary social reality and institutional reality. Ordinary social reality consists of functional entities such as screwdrivers and chairs that must be able to perform the function imposed on them. Institutional reality consists of entities such as dollar bills and weddings whose identity is derived from that fact that some status function has been collectively imposed on them according to the rule "*X* counts as *Y* (in context *C*)." It was also noted that language is institutional in nature and hence part of institutional reality, and that the function or status of objects may be controversial, so that consequently different members of society may employ different social ontologies.

# 3 THE ONTOLOGY OF VIRTUAL WORLDS

## 3.1 VIRTUAL ENTITIES

Just like the real world, virtual worlds have an ontology, meaning that entities encountered in them have a mode of existence that may be analyzed. Before such an ontological analysis can be performed, it must first be clear what entities are found in virtual worlds at all. This can be done by performing a naïve description of the kinds of entities encountered in virtual worlds by ordinary users. At first glance, virtual worlds contain many of the same kinds of entities found in the real world. They may contain spaces, trees, desks, chairs, pencils, dogs, written messages, conversations, money, words, etc. They may also contain entities that have no direct counterpart in the real world, such as cursors, menus, windows or scroll bars, but even these are not unlike some entities that may be encountered in the real world.

Entities encountered in virtual worlds may be called *virtual entities*. At first glance, the ontological status of virtual entities is puzzling. They resemble fictional objects like characters in novels or movies because they do not have physical existence: they have no mass and no identifiable location in physical space. However, virtual entities are not just fictional objects because they often have rich perceptual features and, more importantly, they are *interactive*: they can be manipulated, they respond to our actions, and may stand in causal relationships to other entities. So in our everyday ontology, virtual entities seem to have a special place: different from physical entities, but also different from fictitious or imaginary entities [2].

## 3.2 SIMULATION AND ONTOLOGICAL REPRODUCTION

Being virtual is not the same as being unreal. A remarkable fact about virtual entities is that many of them are accepted as an integral part of the real world. For example, virtual environments can contain real (electronic) money and real documents, and people can play real chess games in them, and trade real insults. An electronic document on a computer is just as real as a paper document in the physical world: it can be moved, lost and destroyed, and it can serve most of the same functions. On the other hand, virtual entities can also be recognized as *un*real, as mere simulations or representations of real-world entities. Virtual rocks and trees are not normally interpreted as real rocks and trees, but as simulations of rocks and trees. It seems, then, that there is a distinction between virtual entities that are accepted as mere simulations of real-world entities, and virtual entities that are accepted as being, for all purposes, as real as nonvirtual entities.

I will call virtual entities that do not just simulate real-world entities but that are in every way equivalent to them *ontological reproductions* of real-world entities. So virtual versions of real-world entities are either mere *simulations*, that only have resemblance to real-world entities by their perceptual and interactive features, or ontological reproductions, which have a real-world significance that extends beyond the domain of the virtual environment.

Interestingly, this distinction between simulations and ontological reproductions turns out to map onto the ontological distinctions made in the previous section. Physical reality and ordinary social reality can usually only be *simulated* in virtual environments, whereas institutional reality can in large part be *ontologically reproduced* in virtual environments. To illustrate, rocks and trees (physical objects) and screwdrivers and chairs (ordinary social objects) can only be simulated in virtual reality. On the other hand, money and private property (institutional objects) can literally exist in virtual reality. I will now proceed to explain why physical and noninstitutional social entities generally *cannot* be reproduced (and note some exceptions to this rule), after which I will explain why institutional entities *can* be ontologically reproduced.

The reason that most physical and noninstitutional social entities cannot be reproduced in virtual reality is that computers evidently are not able to reproduce their essential physical properties. A virtual seed does not possess the physical properties by which it can provide nourishment or grow into a tree. Likewise, an ordinary social object like a screwdriver is in part constituted by the physical ability to drive screws, which cannot be reproduced in virtual reality. It should be pointed out, however, that some physical and ordinary social entities *can* be ontologically reproduced on a computer. This is possible because computer systems are physical systems that have physical powers, and some of these physical powers may be conferred onto virtual entities modeled in them. Computer systems may not be able to ontologically reproduce physical entities that have mass. However, due to their physical powers, they are able to ontologically reproduce certain 'weightless' physical entities, like images, sounds, shapes, and colors. For the same reason, they can contain software emulations of artifacts like synthesizers, VCRs and stereos. These are, however, exceptions to the rule.

In contrast to most physical and ordinary social entities, institutional entities *can* be ontologically reproduced in virtual environments. This is possible because institutional entities are ontologically constituted through the assignment of a status function, of the form "*X* counts as *Y* (in context *C*"). Now, in principle, any status function can be assigned to anything, if only there is the collective will to do it. For example, it is possible in principle to collectively grant telephones the right to marry, which means there can be married telephones. Therefore, if an institutional entity can exist in the real world, it can also exist in a virtual environment. In practice, of course, status functions are only assigned to entities that have certain features that make it sensible to assign the status function to them. As it turns out, many virtual entities lend themselves well for the meaningful assignment of status functions to them. The consequence is that a large part of institutional reality is currently being reproduced in virtual environments, where real institutional activities are taking place like buying, selling, voting, owning, chatting, playing chess, gambling, stealing, trespassing, taking a test, and joining a club, and requisite objects are found like contracts, money, letters, and chess pieces.

## 3.3 The Construction and Recognition of Virtual Institutional Entities

What has not yet been explained is how institutional entities in virtual environments initially arrive at their status, nor how users are able to recognize them as having this status. This is what I will proceed to explain now. First, the *recognition* of institutional entities in virtual environments by users depends on two things: Users must accept the proper constitutive rule ("*X* counts as *Y* (in *C*)") for the entity, and they must recognize the entity as satisfying that rule (i.e., recognize it as being an *X*). For example, a convention may be operative in a virtual environment that yellow rooms are women-only chat rooms. For a user in this environment to recognize a virtual entity as a women-only chat room, he or she must first recognize the virtual entity as a yellow room, and must also recognize that yellow rooms (*X*) count as women-only chat rooms (*Y*) in the context that virtual world (*C*).

In both the real world and in virtual worlds, the status of an entity is often difficult to recognize by only studying the entity and nothing else. For example, it is not always easy to see that an encountered piece of land constitutes private property, or that an man one encounters on the street is married. To remedy this situation, the status of institutional entities is often clarified by what Searle (1995) has called *status indicators*. Status indicators are markers that are intentionally attached to an entity so as to facilitate recognition of its status. So private property may be marked with a fence or with 'keep out' signs, and a married man may use a wedding ring to show that he is married. Often, status markers are linguistic: they are texts or official documents that accompany some entity so as to indicate or certify its status. However, there are also many nonlinguistic status indicators, such as wedding rings and uniforms, and many similar objects and behaviors that have a symbolical, iconic or indexical meaning. The general context in which an entity is encountered may also help to indicate its status. Moreover, some institutional entities, such as dollar bills and English words, do not normally require separate status indicators, because they have intrinsic features that indicate a permanent status and that are recognized by any competent user (although for Chinese users, American money and English texts may be accompanied by a status indicator like an inscription in Chinese identifying them as such).

The interpretation of institutional entities in virtual environments as opposed to their interpretation in ordinary reality also poses a special problem. As claimed earlier, institutional entities in virtual environments may either simulate or ontologically reproduce possible real-world equivalents. This means that users who interpret a virtual institutional entity (e.g., a virtual check or a virtual document) must also decide on the *reality status* of the entity: is it an ontological reproduction that has real-world significance, or is it a mere simulation, that only has meaning within the context of the virtual environment? In the absence of status indicators that also indicate the reality status of the object, this may sometimes be difficult to decide. After all, the way in which virtual entities are hooked up with the real world is often hidden from the users' view. For instance, it cannot easily observe just by looking at it whether pressing a virtual payment button will only result in a payment in the context of a simulation, or whether money will actually be transferred from one's bank account to the other end of the world.

In multi-user virtual environments, this kind of ontological uncertainty often reveals itself in social interactions. This may be due in part because the simulated character of virtual environments may make users feel that their interactions with others may also contain elements of make-believe. As a result, users may sometimes have very different ideas about the reality status of their interactions. Sherry Turkle [3] has observed this phenomenon in MUDs. A regular topic of discussion among MUD users is the status of violence and sexual assault in MUDs. If MUDding is to be understood as playing a game, then perhaps violence and sexual assault are permissible, because they can be introduced as elements of normal play. However, if MUDding is to be understood as built up out of real social interactions, then perhaps violence and sexual assault in MUDs should be understood as really happening, and should be treated as such. Which version of events is correct depends solely on the status functions that MUDders are willing to agree on. And they may not want to choose. Interestingly, users of virtual environments sometimes appear as if they want to keep the dividing line between reality and role-playing fuzzy, so as to have the benefits of real-life social interactions while always having the fall-back option of claiming that it is all make-believe. More often, however, uncertainty about the reality status of institutional entities poses a problem for users of virtual environments that they need to overcome.

Let us now turn to the *construction* of institutional entities in virtual environments. This is the process by which entities in virtual environments acquire a status function and hence aquire their institutional identity. Remember that status functions result from the collective agreement that an entity (*X*) falls under a constitutive rule of the form "*X* counts as *Y* (in *C*)". Now, the initial proposal to apply a certain constitutive rule to an entity need not involve all members of the community that accepts the rule, or even a majority of them. Often, new status functions are proposed and introduced by small minorities, sometimes only by one individual, after which the rest of the community follows suit.

Two main motivations may induce members of the community to accept a proposed status function. First, members may recognize the proposing individual or group to have *special authority* to impose certain kinds of new status functions. For example, when the Bank of England issues a new pound note, British subjects will generally accept the bill as being a pound note, because they accept the authority of the Bank of England in issuing money. Second, people may just hold the status function to be *useful*, and therefore adopt it. For example, members in the linguistic community may come to accept new words like 'nimby' or 'yuppie' simply because someone has proposed them, and they are found to be useful expressions. Often, a combination of both kinds of motivations is operative.

In virtual environments, likewise, a status function of a virtual entity is fixed either because some recognized authority (such as a producer, provider, system operator, or certifying agency) is believed to endorse this status, or because this status has been proposed in a nonauthoritative way and members of the community of users have come to accept it as useful. For example, a virtual room may become a women-only chat room either because a provider has labeled it that way from the beginning and is granted this authority by its customers, or because this status has gradually emerged and come to be accepted within the collective of users. Users also frequently reject impositions of status functions on virtual entities by authorities and often come to assign their own status functions.

## 4 Some Implications for the Design of Virtual Environments

A large part of the reality modeled in virtual environments is institutional in nature. This includes things like money, contracts, documents, gambling machines, and private property, and activities like buying, owning, trespassing, playing chess, and joining a club. In the preceding discussion, I have explained what institutional reality is, and how it is constituted and interpreted in the real world and in virtual environments, but I have not yet drawn implications for the design of virtual environments. In what follows, I will propose two rather intuitive normative criteria or principles for the design of institutional reality in virtual environments, and then suggest what they may mean in practice.

(1) *The transparency principle*: In as far as entities in virtual environments have an institutional status that has been fixed in advance, this status should be clearly marked.

This principle should be important in design because if users are unclear about the status of institutional entities in virtual environments, they are not able to decide and act in an informed way, and may either fail to act, or act without being able to oversee the consequences of their actions and possibly make mistakes. It is therefore imperative that the institutional status of virtual entities is clearly marked. As discussed in the preceding section, the institutional status of an entity can sometimes be recognized by observing intrinsic features of the entity, but is more often inferred on the basis of external status indicators. Consequently, institutional entities in virtual environments should be designed either to have clear intrinsic features that mark their status, or be accompanied by unambiguous status indicators.

The institutional status of virtual entities may be unclear more frequently than that of real-world entities, because virtual environments often contain entities with novel, unfamiliar status functions that cannot quickly be inferred from contextual markers. For example, virtual environments often contain buttons, and pushing a virtual button can mean anything from making a payment to sending a message to canceling a transaction. To learn what a certain button means is to learn a new status function, which has the form "pressing this button (*X*) means ….(*Y*) (in this virtual environment (*C*))". This status function may be learned by linguistic status indicators, such as a an explicit statement of the constitutive rule next to the button, or a less explicit reference, such as the inscription of the word "payment" on the button. Nonlinguistic indicators constitute another possibility. For example, the button may be designed in the shape of a coin, or may be surrounded by dollar signs. Placed within the proper context, these status indicators may be sufficiently clear to warrant the correct inference.

The correct interpretation of statuses can often be aided by designing virtual institutional entities to be *perceptually and functionally similar* to their real-life equivalents. For example, a virtual mall that is designed to look like a real mall may make the process of shopping more transparent than a virtual store with a more abstract interface. Iconic or metaphorical representations of objects and events may also aid the understanding of the consequences of actions. For example, the copying of a file in Windows 95, as a result of giving the computer an instruction, is accompanied by an animation that shows pages being moved from one folder to another, thus helping the user visualize what is happening. Similarly, an electronic payment may be visualized by representations of cash flowing from the user's bank account to the account of the recipient. In this way, users are clearer about the status of objects, actions and events, and their consequences.

Another way in which proper interpretation of status functions may be aided is by *ritualization* [4]. By ritualization, I mean the process by which human practices are standardized so as to follow relatively well-defined scenarios that involve various acts and objects. Marriage is a good example of a ritualized practice, since it is surrounded by an elaborate decorum involving standardized and often highly symbolical actions and objects. Yet even more mundane practices like contract signing usually follow a scenario in which the parties involved perform various formal actions that help them be fully aware on the nature and significance of the event they are participating in.

The value of ritualization is that it aids the interpretation of institutional entities by implicating them in an integrated web of objects and practices that are symbolically related to each other. To illustrate, a marriage ceremony that only involves the joint pressing of a yes/no button does not imbue a marriage with the significance and sense of responsibility that may result from a full-blown marriage ceremony. This suggests that new status functions in virtual environments (e.g., virtual contract signing, entering a private room) may be more successfully implemented if they involve a genuine decorum, involving various standardized acts and symbolisms, that make users aware of the significance of actions and events, and their own role in them.

(2) *The intersubjectivity principle*: In the design of virtual environments, it should be accounted for that institutional reality is not objective in an absolute sense, but is intersubjective, in that it is continuously dependent on collective agreement, may be contested, and may change over time.

Virtual environments may be designed with a lot of institutional reality already pre-inscribed in them, or with very little pre-inscribed institutional reality. Designers may pre-inscribe institutional reality in virtual environments by instructing users through linguistic and nonlinguistic means that entities have certain status functions, notably by marking these entities with status indicators. However, it may not always be appropriate for designers to do this. This depends on what interests the virtual environment is to serve and who has the proper authority to decide this. If the virtual environment is owned by some provider, and users are nothing more than guests, then it may be appropriate to ask users to conform to the institutional structure of the environment pre-inscribed by the provider. The provider, as owner, should be granted significant authority in deciding the status of things he owns. However, if a virtual environment were in fact the collective property of its users, then it would be more appropriate if status functions would represent the collective will of its users. Then it may be more appropriate for designers to pre-inscribe very little institutional reality or at least to make it easy for users to reconfigure institutional features, for example by allowing users to freely change status indicators.

Designers should be aware that the language and (metaphorical) images they use to represent things and events in virtual environments are means by which social realities are created. The way in which things and events are labeled or graphically depicted in virtual environments often endows them with social meanings that they do not have in and of themselves. This point is especially important in considering the design of virtual assistants and other kinds of intelligent agents that communicate with users through language. Such agents function as autonomous units with which users interact in a way that strongly resembles social interactions with other human beings. In ordinary social interaction, individuals adapt to each other's version of social reality, as expressed through language, and sometimes agree on new status functions for entities. This may also occur in the interaction with intelligent agents. Users have to adapt to the system of linguistic categories and the institutional knowledge used by the intelligent agent, just like the agent has to adapt to the user. Also, both intelligent agents and users may propose to assign status functions to virtual entities (e.g., classify an incoming message as "junk mail"), which the other party may or may not may accept. As intelligent agents are increasingly becoming an integral part of virtual environments, it is important to consider the way in which they help structure social reality, and make sure that they do not unilaterally impose particular versions of social reality on users.

## REFERENCES


[1] J. Searle. *The Construction of Social Reality*. Cambridge, MA: MIT Press, 1995.

[2] P. Brey. Space-Shaping Technologies and the Geographical Disembedding of Place. In A. Light and J. Smith, editors, *Philosophy & Geography vol. III: Philosophies of Place*. New York and London: Rowman Littlefield, 1998.

[3] S. Turkle. *Life on the Screen. Identity in the Age of the Internet*. Camberidge, MA: MIT Press, 1995.

[4] B. Pfaffenberger. Technological Dramas. In *Science, Technology and Human Values*, 17, pages 282-312, 1992.

# Investigating Navigation and Orientation within Virtual Worlds


Dimitrios Charitos
Department of Informatics
National and Kapodistrian University of Athens
Greece
virtual@di.uoa.gr

Alan H. Bridges
Department of Architecture and Building Science
University of Stathclyde
Glasgow, Scotland
a.h.bridges@strath.ac.uk

Drakoulis Martakos
Department of Informatics
National and Kapodistrian University of Athens
Greece
Martakos@di.uoa.gr



## ABSTRACT

This paper presents an experiment which investigated the impact of the way that a place was positioned in relation to a path in a virtual environment, on the orientation of subjects within this place. The application of any rotation on a place in relation to a path was found to decrease the easiness of orientation within the place. The impact of pitch and roll rotations on a place was compared. Finally, the way that subjects positioned their viewpoints when they moved within a place was investigated.



**Keywords:** Virtual environment design, HCI in virtual worlds, virtual reality.


## 1. INTRODUCTION - BACKGROUND TO THE EXPERIMENT

This paper documents part of a research project, which has attempted to develop an architectural way of thinking about designing space in a virtual environment (VE). In specific, it presents one of several experiments which investigated several design issues, pertaining to the way that spatial elements may be composed for the purpose of aiding human wayfinding behaviour in a VE.

These experiments were conducted on the basis of a hypothesis which was the result of the theoretical part of the project. This hypothesis was based on the assumption that by understanding how we think of and how we remember spaces we may begin to comprehend the spatial structure that we try to infer onto an environment when we experience it for the first time. If the form of spatial elements and the overall structure of space in a real environment was designed according to this understanding, then this environment would be legible, would facilitate the generation of cognitive maps and consequently wayfinding and would eventually be pleasant to exist in. Accordingly, the project attempted to:

- identify a taxonomy of the space-establishing and spatial elements that a real environment (RE) consists of,
- understand the characteristics of these elements, their spatial significance for humans andtheir possible structure within an environment.
- investigate the relevance of the taxonomy and structure of real space for space in VEs, by trying to identify the intrinsic characteristics of VEs and the differences between REs and VEs.Consequently, the project proposed:
- a taxonomy of the possible constituent elements of space in a VE,
- certain hypotheses about their characteristics and spatial significance for a navigating subject and
- certain hypotheses about the way that they could be structured within a VE,

on the basis of the assumption that a human perceives and remembers space in a VE in the same manner and via the same processes as with space in an RE.

According to these hypotheses, space in a VE may comprise the following elements:

- places
- paths
- intersections and
- domains.

These spatial elements are established, out of the void, by appropriately arranging the following space-establishing elements:

- landmarks
- boundary objects
- thresholds and
- signs.

This hypothesis, which has also been presented and explained in [2] and [1] is seen as a framework for supporting the spatial design of the particular type of VEs, which represent very complex material world information or information that has no physical representation, in an abstract manner; it does not refer to simulations, where the aim is to realistically imitate real objects and events. This framework may inform the design of form in order to formulate space and convey spatial meaning in a VE, for the purpose of implicitly enhancing the spatial awareness of its participant and ultimately aiding his orientation and wayfinding within this VE. As in the case of urban environment design, a VE may be designed so that the conveyed sense of space allows users to anticipate forthcoming events and directs them towards spaces significant for the goal of a particular task. It can also been suggested [3] that a legible environment which facilitates the formation of cognitive maps, is preferred by subjects for navigating and performing several activities in. In this sense, the use of this framework may aid the design of legible VEs, which are preferable for navigation and which make the execution of several tasks more pleasant, as well.

## 2. PLANNING PHASE

During the design of several pilot VEs for implementing the above mentioned hypothesis, it became evident that if a place was positioned in certain ways in relation to a path, a subject who moved into or out of the place would become somehow disorientated. This phenomenon indicated an issue, which was crucial for the composition of domains by appropriately arranging places and paths in the three-dimensional space of a VE. This issue has been the object of an experiment, documented in this paper.

This experiment investigated the impact of the way that a place was positioned in relation to a horizontal path, on the sense of orientation experienced by subjects who

1. entered the place,
2. performed a task in the place,
3. exited the place.

Additionally, the way that subjects positioned their viewpoint when they tried to orientate and perform a task in the place, dependent upon how this place was positioned in relation to the path, was also investigated. The ultimate aim of this investigation was to identify how do places have to be positioned in relation to a path in three-dimensional space, for facilitating the "sense of orientation" of a subject who navigates into the place and performs a certain activity within this place.

Providing a "sense of orientation" is a significant parameter that has to be taken into account when designing space in VEs. As is the case in real space, it is assumed that in a VE a subject has to feel orientated within a space in order to "belong" there: "*It is ... important that our environment has a spatial structure which facilitates orientation*" [4, p.423-4]. A subject has a "sense of orientation" within an environment when he is able to maintain a direction while moving, or to point to a direction, independently of his location in space and independently of cues originating from the environment [5]. Spatial orientation and wayfinding could both be defined as *"a person's ability to mentally determine his position within a representation of the environment made possible by cognitive maps."* [5, p.35].

## 3. DESIGN PHASE

Since there was no precedent for predicting the behaviour of subjects in such a situation, it was not possible to develop a hypothesis about the subjects' response. The independent variable which was taken into account in this experiment was the way that each place was positioned in relation to the path; this variable was described by the axis and angle of rotation of the place in relation to the path. This qualitative factor was set at specific levels of interest and its effect

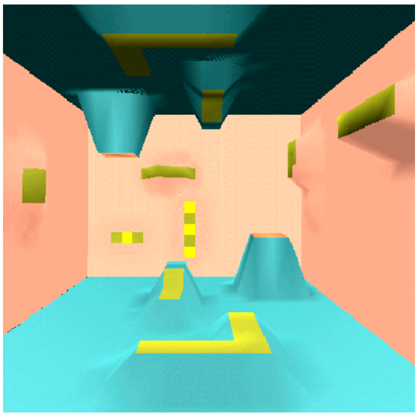


**Figure 1:** Interior view of a place

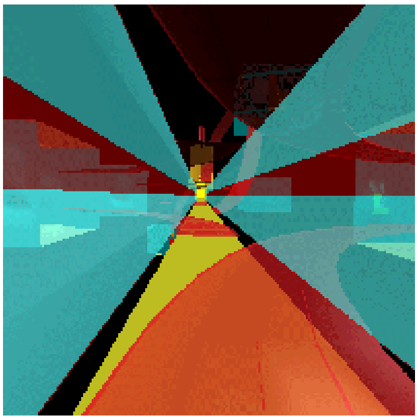


**Figure 2:** View along a path

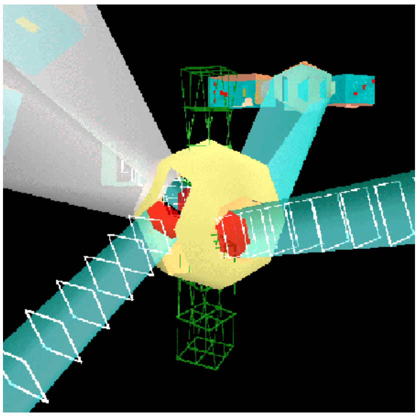


**Figure 3:** An intersection of paths

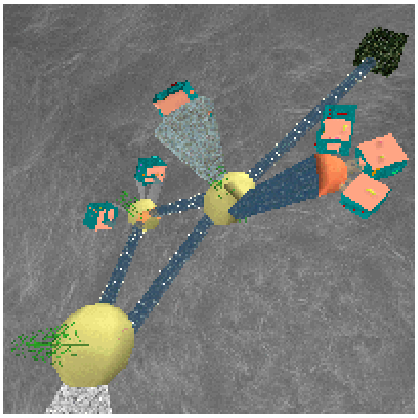


**Figure 4:** A domain comprising places, paths and intersections.

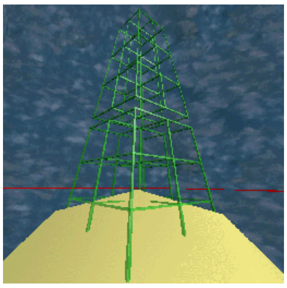


**Figure 5:** A landmark object on top of a sphere.

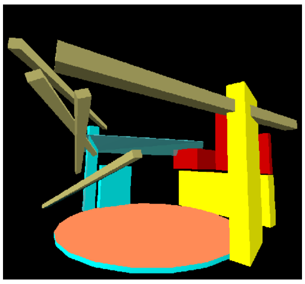


**Figure 6:** A space defined vaguely by its boundaries.

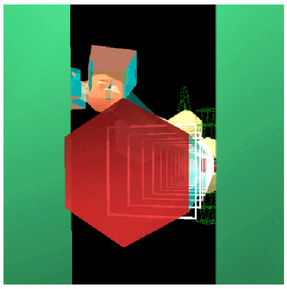


**Figure 7:** A threshold indicating the entrance to a path.

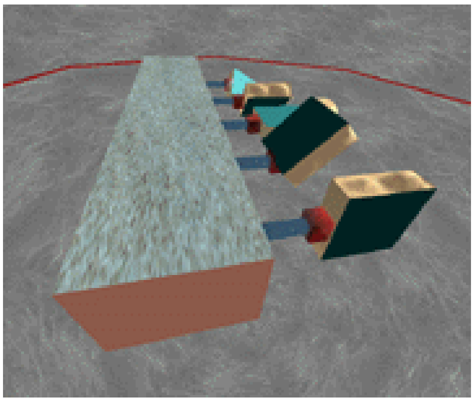


Figure 8: Side view of the experimental domain.

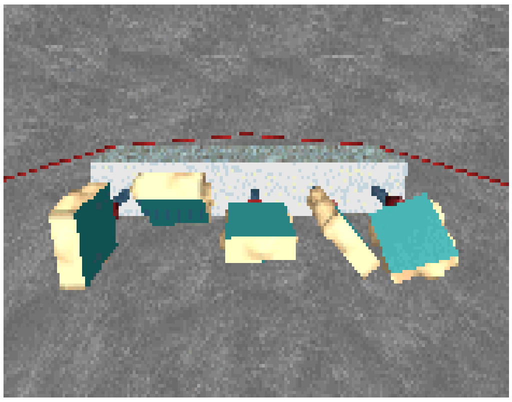


Figure 9: Front view of the experimental domain, with place (1) on the far left and place (5) on the far right

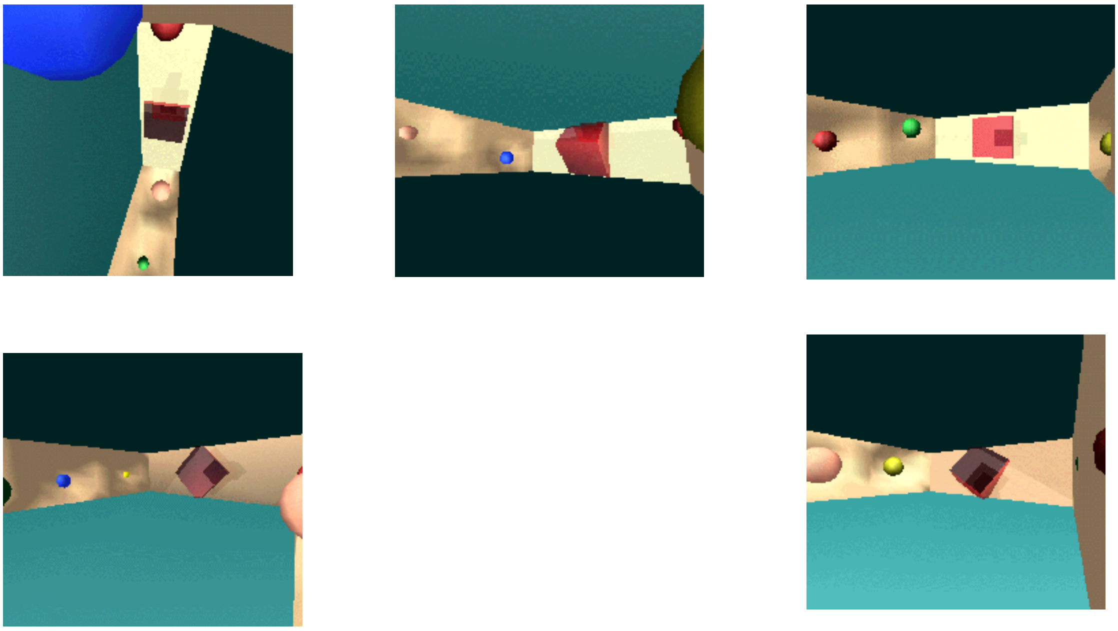


Figure 10: View of the interior of each place from the same corner towards the threshold at the entrance; top left is place (1), top in the middle is place (2), top right is place (3), bottom left is place (4) and bottom right is place (5).

on the dependent variables was studied. Other factors which may have affected the behaviour of subjects in each place were the physical characteristics of each place and path and the way that each path was spatially arranged; these factors were rigidly controlled so that all paths and places were identical and all paths were positioned in a similar way. Finally, the order with which subjects experienced one place after the other was randomised.Therefore, a VE, which consisted of

- a central hall on the one surface of which 5 entrances to 5 paths were positioned;
- 5 identical, horizontal paths that lead to
- 5 identical places, differently positioned in relation to each path,

was designed. All places were orthogonal parallelepiped in form, their length being equal to their width and their height being a third of the their length. The bigger two boundaries of the place were not intended to be seen as a 'floor' and a 'ceiling' respectively. Instead, the intention behind investigating the way that subjects positioned their viewpoint inside each place was to identify whether subjects did indeed tend to see these bigger surfaces as floor and ceiling or not. Each of the 5 places was rotated in a different way in relation to the path. Five numbers, which were positioned at the entrance of each path from within the central hall, signified the way towards each of the places.

The way that these places were positioned can be described if we consider an (x,y,z) co-ordinate system as a reference, where z is the main axis of the horizontal path, y is the vertical main axis of the numbers in the paths' entrances and x is the axis parallel to the longer direction of the central hall, then:

1. Place (1) is rolled $90^0$ around the z axis
2. Place (2) is pitched $45^0$ around the x axis
3. Place (3) is not rotated at all
4. Place (4) is rolled at a $-60^0$ angle to the z axis
5. Place (5) is rolled at $30^0$ to the z axis and tilted at $-45^0$ to the x axis.

The impact of the way that these places were positioned on the criteria:

- how easily do subjects orientate while performing an activity,
- the way that subjects position their viewpoint in relation to a place

was studied. The results for the first criterion would identify which ways of positioning a place in relation to a path were preferable and which were problematic, for a subject who navigates within a domain, comprising places and paths.

Subjects were requested to enter the place, perform a simple task in there and then exit the place. This task aimed at motivating the subjects to enter the place, move around and perform a certain activity there and was not supposed to have an impact on the response of subjects in terms of how easily they orientated. Five spheres were positioned inside certain concavities on the surfaces of each place, at similar positions. Each sphere differed in colour and size from the other spheres. In each place, the task was to identify which was the colour of either:

- the largest or
- the smallest sphere in each place.

The different order of presentation was expected to randomise any unpredictable effects that the task may have had on the responses.

After performing the task and exiting the place, subjects were asked how easy it had been for them to orientate while:

- entering the place,
- moving within the place in order to perform the task and
- exiting the place.

Their response was measured on a scale of 0 to 100. Time of task execution was also recorded and was used as the second dependent variable for this experiment. The two dependent variables studied, were the 'easiness of orientation' and the 'time of task execution'.

A third dependent variable corresponded to the way that subjects positioned their viewpoints, when they tried to orientate within each place in order to do the task, as observed by the researcher. As a result of a pilot experimental study, involving three subjects, four different possible positions of the viewpoint within a place were identified. If we consider the (x,y) co-ordinate system of the subject's viewpoint, where x is the horizontal and y is the vertical axis, then the four options were:

1. the y axis of the viewpoint was vertical to the two biggest boundaries of the place that one could refer to as a 'floor' and 'ceiling';
2. the y axis was parallel to this 'floor' or 'ceiling';

3. the viewpoint was positioned randomly;
4. subjects tried to keep a global vertical orientation, following external cues visible to them.

Subjects were generally constrained into navigating within the limits of the experimental domain, so the surroundings of this domain only existed as a background to the experienced interior setting. These surroundings consisted of a spherical environmental map and a horizon. Although the experimental domain was largely opaque and therefore these elements were rarely seen, they still aimed at aiding the subject in terms of orientation to a global context.

The task, which was executed in each of the places, was not supposed to have an impact on the response of subjects in terms of how easily they orientated. The only aim of the task was to motivate the subjects to enter the place, move around and perform a certain activity there, so that their spatial behaviour, while doing something in this place, was investigated. Therefore it was not necessary to assign a different task in each place in order to avoid learning effects on the dependent variables. Accordingly, the tasks in each place were similar. The different order of presentation was expected to randomise any unpredictable effects that the task may have had on the responses.

31 subjects (11 female and 20 male) took part in this experiment.

## 4. ANALYSIS PHASE

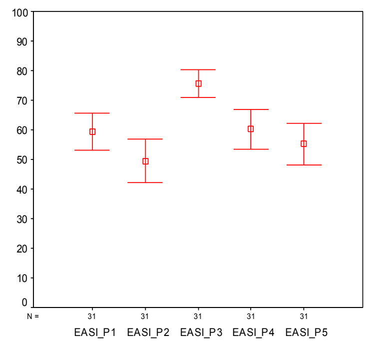
Since all subjects experienced all five places, a repeated measures general linear model ANOVA procedure was used for the analysis of the results for the two variables of 'easiness of orientation' and 'time'. The within-subjects factor was named 'place' and had 5 levels, each of them corresponding to each of the 5 places of the experimental domain. The measures recorded at each level corresponded to the two dependent variables - 'easiness of orientation' and 'time' respectively.

With respect to the third dependent variable, tables and bar charts of frequencies, for the four different values of the response, were investigated in order to study the spatial behaviour of subjects in each place. Any relationships among behaviour in different places were established on the basis of these tables and charts.

### 4.1 THE VARIABLE OF EASINESS OF ORIENTATION

For the first dependent variable, the analysis of the results aims at rejecting the null hypothesis that: "in all five places, subjects found it equally easy to orientate within the place in order to do the task". The results for the variable of 'easiness of orientation' are illustrated by the error bar graph presented below. These results provide evidence for the rejection of the null hypothesis for the variable of the 'easiness' of orientation. Accordingly, it can be concluded that:

- The control place (3) - which was not rotated at all - was clearly the easiest to orientate in, compared to the other four places.
- Place (2) - the floor of which was tilted $45^0$ around the x axis of the reference co-ordinate system - was clearly worst than places (1), (3) and (4) but it is not clear whether it was worst than place (5) as well, in terms of easiness of orientation.
- All other contrasts were not significant enough to be considered; this implies that it is difficult to differentiate between the easiness of orientation in places (1), (4) and (5) and also between place (2) and place (5).



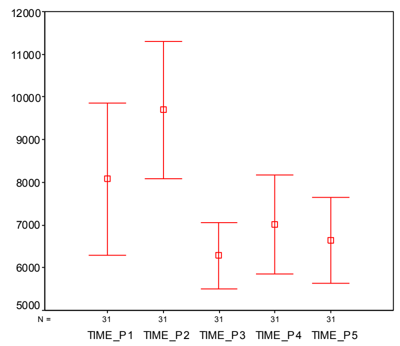




Chart 1: Error bar graphs for the variable of 'easiness' of orientation (left) and for the variable of 'time'

.

## 4.2 THE VARIABLE OF TIME

For the second dependent variable, the analysis aims at rejecting the null hypothesis that: "subjects spent an equal amount of time in all five places". The results for the variable of 'time' are illustrated by the error bar graph presented below. These results provide evidence for rejecting the null hypothesis for the variable of 'time'. Accordingly it may be concluded that:

- Subjects spend significantly more time in place (2) than in places (3), (4) or (5) and slightly more time in place (2) than in place (1).
- Subjects spend significantly less time in place (3) than in place (1) and (2) and slightly less time in place (3) than in place (4).
- Subjects spent slightly more time in place (1) than in place (5).

Due to the form and size of places in this experiment, which was not likely to induce exploratory behaviour, subjects did not spend any time in each place doing much more than merely performing the task. This fact was also confirmed by observation of the subjects' behaviour in these places. Accordingly, the variable of time has been taken into account in this experiment. As a result, it can be argued that the results for 'time' support the analysis of the 'easiness' results, in that:

- Place (3) is the easiest to do the task in and it took subjects significantly less time to do the task in place (3) than in places (1) and (2); a slight difference was also detected between place (3) and place (4) but no difference was detected between place (3) and place (5).
- Place (2) is considered the most difficult to orientate and do the task in and accordingly it took subjects significantly more time to do the task in there than in the other places.
- No significant differences were detected for either easiness or time measurements among places (1), (4) and (5).

## 4.3 THE VARIABLE OF VIEWPOINT ORIENTATION WITHIN A PLACE

The second objective of this experiment was to investigate the impact of the way that a place is positioned in relation to a path on the way that a subject is positioning his viewpoint, while moving within this place. Since the response values for this dependent variable were provided by nominal measurements, frequency tables and bar graphs were studied for the purpose of identifying possible patterns among responses in the five places. Since there were no precedents for predicting the outcome of this part of the experiment, no specific hypothesis about the behaviour of subjects in each place was defined. The aim of analysing the results is to find out whether subjects thought that they needed to position themselves in a particular way in order to orientate and feel comfortable enough to perform the task in each place. It is therefore concluded that in all five places, subjects largely preferred to position their viewpoint perpendicularly to the two planes defined by the two larger dimensions, which could be seen as the 'floor' and the 'ceiling' of the place.

It is understood that the way that subjects had orientated their viewpoint in relation to the threshold as they entered each place may have affected the way that subjects perceived the place as being orientated in relation to the path; this variable could not be controlled. The possibility of such an effect had been detected during the pilot experimental study. It was therefore decided that subjects would be asked to maintain the y axis of their viewpoint parallel to a global vertical orientation, as they entered each place, but it was stressed to them that they were free to move anyway they felt after entering the place. The word 'floor' was not used by the researcher during this prompt, in order to avoid influencing subjects into considering one of the path's surfaces as a floor. It is understood that the way that subjects were prompted may have affected their overall response in a manner that has not been predicted, but it was nevertheless considered the best possible way to make subjects enter all places in a similar way.

As was concluded from the analysis, the majority of subjects in all five places wanted to position their viewpoint vertically to the surface that they perceived as a 'floor'. For the purpose of explaining the spatial behaviour of subjects, with reference to this viewpoint, it is essential to consider a subjective co-ordinate system for a subject where

- x as the horizontal axis of the subject's viewpoint
- y as the vertical axis of the viewpoint and
- z as the axis perpendicular to the viewpoint and extending away from the subject.[1]

Using this co-ordinate system as reference, it can be suggested that humans feel more comfortable to yaw their viewpoint - rotate around the y axis - rather than pitch their viewpoint - rotate around the x axis - when they are looking around in a space and exploring it. It is natural to pitch our head up or down at rather small angles, but it would seem unnatural to tilt our whole body $360^0$, as would be the case for a subject moving his viewpoint parallel to the floor in place (1).

Accordingly, a possible explanation is that: a human in an enclosed space, tends to position his viewpoint so that the largest area of this space is parallel to the (x,z) plane of his subjective co-ordinate system and tends to translate and rotate this viewpoint along this (x,z) plane for the purpose of exploring the space; consequently, the smallest dimension of this space is perceived as height. This phenomenon may be attributed to the fact that humans are more used to being in places where walls are rarely much bigger than floors or ceilings. Such places are more realistic and are similar to the majority of spaces that we experience, in our everyday life. In an urban context we may experience much higher walls, but this is usually the case when we are in a space which, unlike a place, is not completely enclosing us. These suggestions are supported by at least 9 subjects' observations. It is understood that such suggestions are valid only for a place of form and proportions such as the places used in this experiment; for places of different form and volumetric proportions with objects placed within them in a different manner, further experiments should be conducted.

Some subjects felt that it was significant for them to establish a relation with certain surfaces in the place, which they understood as floor, ceiling and walls. One of them attributed this need to the fact that a floor and a ceiling made her feel safe. Four of these subjects also reported that the colours of the surfaces affected the way that they orientated within the place. It seems that the difference in colour between the floor/ceiling and the walls of the place influenced the identification of these surfaces within each place. However, since the colour of both bigger surfaces - which could be seen as floor and ceiling - was the same, subjects had to detect other cues in order to differentiate between the two, if they felt they needed to do so.

## 5. CONCLUSIONS

This experiment has investigated the impact of the way that a horizontal path is positioned in relation to a place and has concluded that:

- The application of a rotation on the place in relation to the path decreases the easiness with which a subject orientates in this place.
- The application of a significant pitch rotation on the place in relation to the path makes orientating in this place difficult and confusing.
- Subjects seemed to have less trouble orientating in a place, to which a significant roll rotation or a less significant pitch rotation has been applied.
- With respect to orthogonal parallelepiped places, subjects tend to interpret the biggest boundaries of the place as 'floor' and 'ceiling' and accordingly position their viewpoints vertically to these boundaries in order to achieve a state of orientation and perform a task in this place. However, a smaller majority of subjects is likely act like that, when the rotation applied to the place is significant. Subjects generally preferred yawing rather then

[1] This convention was dictated by the way that the co-ordinate axes of the user viewpoint are considered when authoring a simulation with WorldUp ® by Sense8, which was the simulation tool used for developing this experiment.

pitching their viewpoint within a place, in order to look around in this place.

## REFERENCES


[1] The Architectural Design of Information Spaces in Virtual Environments. In Ascott, R.,ed. *Proceedings of the 1st International 'Consciousness Reframed' Conference*. Newport: CaiiA, University of Wales College, 1997.

[2] Designing Space in Virtual Environments for Aiding Wayfinding Behaviour. In Bowden, R., ed. *Proceedings of the 4th UK VR-SIG Conference*. Brunel University, 1997.

[3] The image of the city, MIT Press, Cambridge, Massachusets, 1960.

[4] “The Phenomenon of Place” (first published in 1976) in Nesbitt, K. (ed.), *Theorizing a New Agenda for Architecture: An Anthology of Architectural Theory,* Princeton Architectural Press, New York, 1996.

[5] Wayfinding in Architecture, Van Nostrand Reinhold, New York, 1992

# VR-Spider for (non) Immersive WWW Navigation

M. Cibelli, G. Costagliola, G. Polese, G. Tortora
University of Salerno
Dipartimento di Matematica e Informatica
Via S. Allende 84081, Baronissi (SA)- Italy
maucib@pantelleria.diaedu.unisa.it
{gencos| giupol| jentor}@udsab.dia.unisa.it

## ABSTRACT

In this paper we present a novel architecture that maps the two-dimensional Internet informational space into 3D worlds according to predefined or inferred metaphors and structures. The main component of this architecture is a pool of specialized web miners, called VR-Spider, that visit sites trying to formulate assertions about the domain of the site according to learning and inferring mechanisms and by means of fuzzy domain modeling.

**Keywords:** 3D WWW, Virtual Reality, VRML, Web Miners

## 1 INTRODUCTION

Immersive systems reproduce human and natural actions in a virtual environment and allow naive users to interact with complex systems easily. With the rapid expansion of the WWW the application of Virtual Reality technology to the design of advanced web-navigation systems becomes a natural step.

With the terms "virtual reality" we define computer-generated environments. More and more scientists are developing new technologies and new solutions every day. Virtual reality will become the standard interaction between humans and computers and probably between humans itself.

The WWW is used by an enormous number of people often not experts. For this reason many specialized tools, more and more advanced, help users during the navigation and the interaction with the Web. In this panorama, advanced user interfaces and search/navigate mechanisms are well suitable.

Today, we distingue two different typologies of Virtual Reality (VR for short) systems [12]:

- Immersive systems, utilizing mounted head-sets, devices for body tracking, tactile response peripherals, sound and smell simulation and even complex fully immersive spherical systems[7][11][18][17][15][16];

- Non-Immersive systems, in which is the computer monitor the only viewport into the virtual world [3][4][14][22][23];

Non-immersive environments are easy to realize because they do not require ad-hoc peripherals. However a monitor does not express a 3D world naturally. On the other hand, actions performed with immersive peripherals reproduce natural functions of human art, architecture and symbolic development [10], but require dedicated peripherals.

The web is a collection of semi-structured information [6] and every day is visited by a large number of specialized agents that categorize, filter, manipulate, index, and map web information. Other agents learn how to interact with web services autonomously. These agents are often identified as web miners [5].

Many studies have proved that metaphor visualizations help users during the interaction with the WWW [21]. An example is the well-known desktop metaphor [1].

In this paper we present a novel architecture that map the "2d"-internet (e.g. the multimedia web representation of the information is typically viewed in a 2D form paradigm) in 3D worlds according to predefined or inferred metaphors and structures. The main component of this architecture is a set of web miners that formalize the so-called VR-Spider. These web miners visit sites trying to formulate assertions and conceptualization of the sites and trying to derive a feasible 3D visualization by means of learning and inferring mechanism. Completely new concepts and/or low probable assertions are resolved with the help of a VR-manager that expand the knowledge of the VR-Spider incrementally.

The paper is organized as follows. In section 2 we introduce related works, section 3 shows the main components of the VR-Spider architecture, section 4 presents a prototype and section 5 concludes and presents our further researches.

## 2 Related Works

Virtual Reality and WWW are concepts strongly related and the number of Virtual Web Browsers developed prove this.

The Virgilio[14] project reduces cognitive load providing a non-immersive virtual reality environment to browse a multimedia database. However Virgilio works with high-structured information and is not well suitable for the chaotic WWW.

An example of a more web oriented VR viewer is the *vrmlViewer* that loads a VRML file, interprets and displays the scene described by the file [2]. *Web3D* [4] display real-time animations of virtual objects providing an easy editor that allows the definition of 3D objects. Web3D is capable of embedding graphical objects within text documents, and of linking these documents by means of hypertexts. Hyperlinks can be other virtual worlds or HTML documents. In the last case Web3D invokes a standard WWW browser. However it needs a previous specification of the virtual worlds.

A different approach for the use of 3D on the Internet is given by *Visual Expresso* [23] that is more context oriented than the previous 3D tools. Visual Expresso examines the structure of an Internet site, paying particular attention to hyperlinks, and builds an intermediate data structure to provide a 3D representation of the site. A user can navigate the site by moving in a 3D world. One problem with Visual Expresso is that the virtual world reproduces only the hyperlink structures rather than the content.

The *i3D* scene viewer project [3][8][9] for annotated 3D environments combines full-screen rendering using CrystalEyes LCD shutter glasses and a space-ball 3D input device, providing a desktop-VR-user-interface. By using i3D, in the Virtual Sardinia Project [22] users can explore a 3D model of the island, built from digital terrain model data, textured with satellite images. The 3D objects representing hyperlinks are positioned on the surface.

The main difference between our system and the current VR-Web-Browsers is the definition of the virtual metaphors to be exploited for representing a site. VR-Spider is content oriented. The main goal of the system is to reproduce metaphors best representing the document content of the site by maintaining the site structure.

## 3 The Virtual Reality Spider Architecture

The Virtual Reality Spider is a pool of web miners that produces a virtual representation of the WWW. The main goal of the VR-Spider is to traverse the Internet collecting enough information to infer virtual world metaphors for each visited site. A first high level architecture is shown in figure 1.

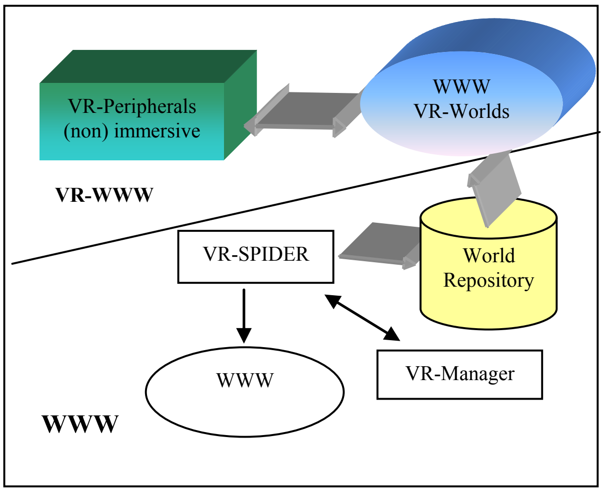



Figure 1: An high-level representation of the VR-Spider architecture

The VR-Peripherals could be immersive devices or simply a non-immersive interface. The WWW is navigated following the WWW VR-World map that is essentially the meta-www structure. Each world represents a site and is coded in the world repository. The discovery of new worlds (with inferring mechanism or automatic categorization procedures) will update both the map and the world repository. The VR-Spider is mainly composed of the *structure-miner*, the *conceptual-miner* and the *builder-miner*.

The mapping of the WWW in a 3D environment needs knowledge base. This information can be extracted with a trainable system providing a training set of domain-specific documents and/or hand-coded examples. Starting with this knowledge the VR-Spider will apply a categorization and a conceptualization of the target site.

The VR-Spider generates faults in case it is not able to derive any domain. A VR-Manager feedback could help during the definition of a new domain improving the system knowledge. The mapping process can be summarized in three different phases each assigned to a single agent.

- Visit the site and produce an internal schema of it (*structure-miner*).
- Define a feasible virtual world and metaphors. (*conceptual-miner*)
- Generate the corresponding WWW VR-Worlds map (*builder-miner*)

In the following we presents the three miner architectures.

## 3.1 THE STRUCTURE MINER

The structure process is the first step of the entire task. This is done by the *structure-miner* (see figure 2).

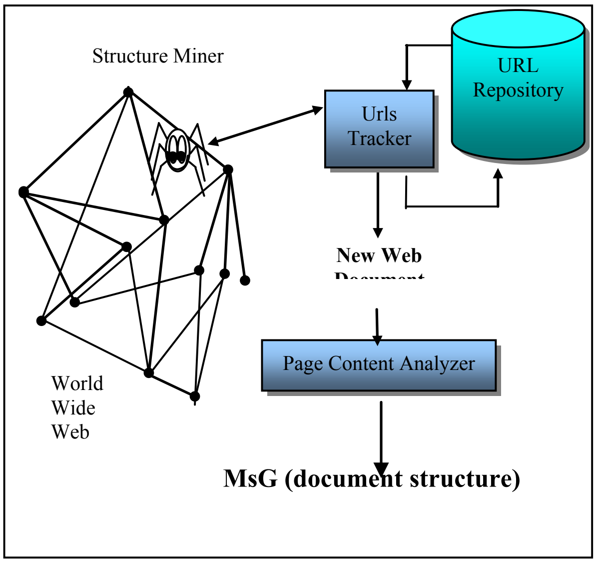



Figure 2: The structure process

The structure miner keeps track of new internal URLs by testing the URL tracker that stores the miner visit history. The Page Content Analyzer successively analyzes each local URL.

The Page Content Analyzer groups different media types while parsing the document source (e.g. parsing html tags), and extracts additional information by processing html layout cues and paths. This cluster allows a successive analysis in order to derive the metaphor set to adopt. The output of this process is the so-called *Meta-Site-Graph*

A meta-site graph MsG=(V,E) is a structural view of a site where each node contains the list of the conceptual components that form a page (e.g. sound objects, text portions et.). Given u, v $\in$ V, there is an edge between a node u and a node v if and only if there is a hyper link between the page u and the page v (a hyper-link between two URLs).

An example of a MsG is depicted in figure 3. The corresponding site is composed of four documents. The homepage contains two text portions and two multimedia objects (e.g. a sound and a movie).

This page contains two hyperlinks to nested pages that contain other media objects.

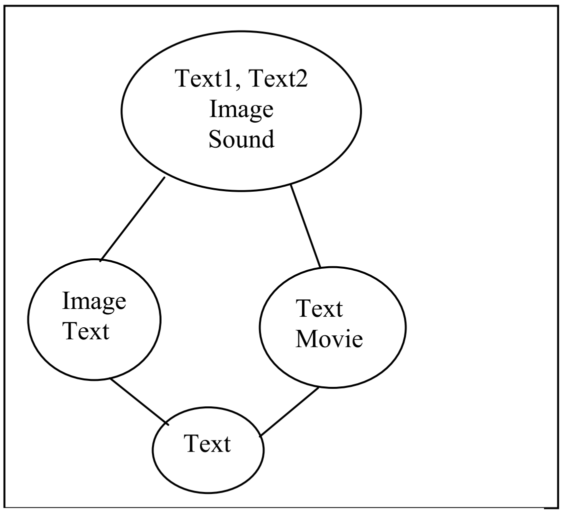



Figure 3: An example of a meta-Site Graph

Once we have organized the site with a MSG, the next step is the conceptualization of the information stored in the MsG.

## 3.2 THE CONCEPTUAL-MINER

The conceptualization process is a necessary step to derive the virtual world representing a site. A feasible site mapping must be provided as well as a set of metaphors to represent the information within the documents. The conceptual-miner architecture is shown in figure 4.

The derivation of the domain best representing a site is done mainly analyzing textual information. The conceptual-miner process can be summarized in two main steps

- Categorization of a site
- Conceptualization of the documents content

The categorization process mainly focuses on text analysis algorithm and is based on *multilevel categorization* [13] that combines text categorization and clustering methods.

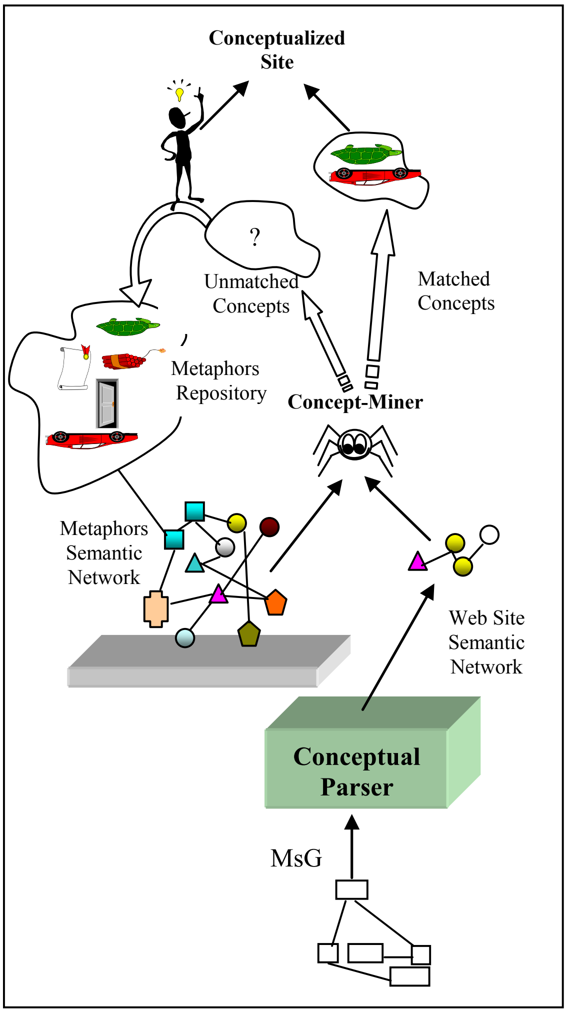



Figure 4: The conceptual-miner architecture

Web pages are very complex and various. Furthermore on the World Wide Web new topic and not defined categories can be found frequently. It seems to be a hard task to predict all categories. This problem is avoided in multilevel categorization because unknown topics are classified in "negative example" categories. These categories are used to predict new coming categories. The web page representation is based on the Term Frequency and Inverted Document Frequency widely devised in [19].

Negative categories can be mapped with predefined metaphors and virtual worlds. For example, in the current prototype the system adopts a building metaphor if any inferred domain metaphor is available. The default associations can be refined at a later time by human assistance.

However document categorization do not capture spatial and temporal relations. For this reason we have integrated the categorization step with a conceptualization of the site by using CD forms [20].

An actor, a conceptual category, an object, plus other information specific to the particular category characterize a CD form. Each conceptual category is associated to a collection of verbs, which are used to detect the occurrence of the conceptual category within textual sentences. As an example, the verbs *give*, *take*, *receive*, *sell*, etc, all mean a transfer of ownership, and hence will be all modeled by the same conceptual category ATRANS. Thus, the sentence “Paul gave Alice a ring” can be conceptualized with the ATRANS category, where Paul is the Actor, ring is the object, and the direction is from Paul to Alice. These categories are directly mapped onto semantic-networks, where network nodes represent the actor and the object, and the category by links.

Furthermore we have extended the conceptual categories presented in [20] with the following categories:

- Spatial

This category includes spatial relationships between text blocks, or between text blocks and image icons, etc. As an example, in the web site of a university we might find a vertical list of structures, including Faculties, Departments, etc. The conceptual parser might include this information in the spatial category. Examples of spatial relationships include Vertical, Horizontal, Overlay, Diagonal, etc.

- Temporal

To capture temporal relationships in web documents we analyze temporal transaction statements. For example, in the university web site mentioned above, we capture all the dates related to courses and academic years to construct temporal relationships between these web elements. The Conceptual Parser will associate these elements to the temporal category. Temporal relationships include co_start, co_end, meet, before, after etc.

The conceptual miner visits the conceptual representation of the web in order to associate parts of it to parts of the semantic network representing aspects of metaphors. The matching process could present a certain number of failures. In this case, an expert could adapt a metaphor or modify its conceptual representation to reduce failures.

The main steps of the conceptual miner algorithm is the following:

- The Virtual World to adopt

The entire site is mapped with the metaphor according to the category the document belongs to The world name is the homepage document name.

- How Hyperlinks are mapped

For each hyperlink, try to associate a metaphor analyzing the hyperlink name. If there are no sufficient information, then put default metaphor in the virtual space to represents the link, otherwise connect the document to the derived metaphor.

- How document content is mapped

Put the multimedia information in each predefined place inside the metaphor adopted. That is, each metaphor contains virtual spot where text, image suond etc. are placed. For example, in a concert ticket room metaphor the textual information is placed in a poster attached to the wall. In the same virtual room the sounds came out from a radio. Note that when a metaphor is designed, physical and logical constraints must be specified in order to prevent illogical representations.

## 3.3 THE BUILDING MINER

The aim of the *building-miner* (see figure 5) is to generate VRML sources according to the derived world and metaphors.

The building-miner interacts with the domain repository and the web documents producing new html sources reproducing the virtual world where users navigate and place the concept inside this world. The generation process is a link between the concepts and the metaphors VRML sources.

Once the process is completed, users can navigate the target web site in a virtual scenario.

A virtual navigation represents an isomorphism between the underlying Internet and the high level vision. This will augment the user's ability providing a lower cognitive load and a low number of mismatched concepts as well as low errors.

Figure 5: The building miner architecture

# 4 AN EXAMPLE OF VIRTUAL WORLD GENERATION

We have developed a java prototype of VR-Spider. The goal of this first prototype is a virtual world representation of music related information.

To show how the VR-Spider prototype works, we trace its main steps during the automatic mapping of a site related to music concerts at the Amalfi coast site (see http://www.rcs.amalficoast.it). Afterwards, the VR-Spider has been launched on other music-related sites, without providing additional information.

The concepts provided in the first training phase have been reused in order to limit the human interaction. The target site homepage is shown in figure 6.

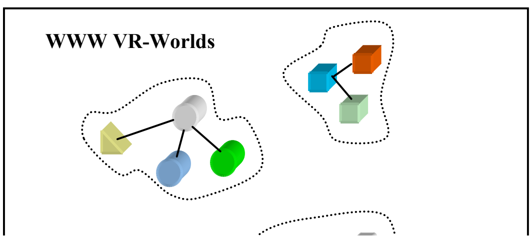



Figure 6: Homepage of the target site

Our system has classified the target site in the category of concerts. Thus, according to the knowledge provided to the VR-Spider, the system adopts an open virtual space metaphor to represent the site. In figure 7, part of this virtual world is depicted. (note that the picture in the background representing a villa is added manually)

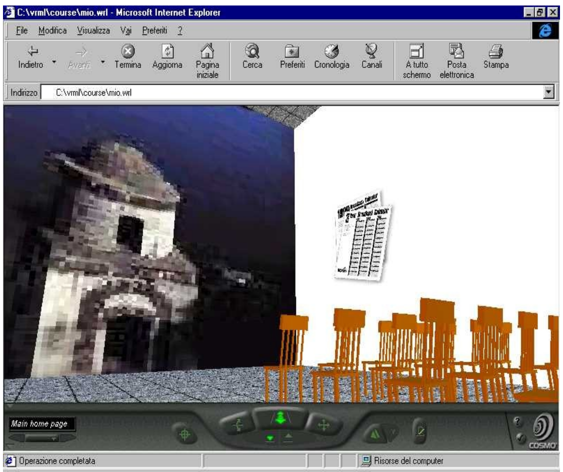



Figure 7: The new Virtual Homepage of the target site

An example of derived metaphor is the ticket office that has been placed inside the virtual world, because in the target site there is nested link to a page containing ticket information.

Another example of derivation is the calendar metaphor used to represent the tour date link (see link "January-December". A portion of semantic network is shown in figure 8.

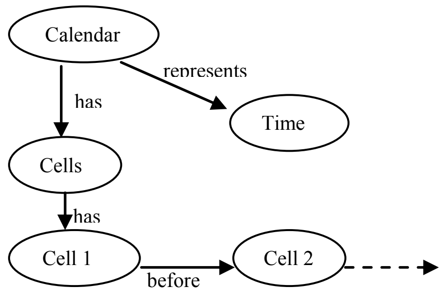



Figure 8: A fragment of semantic calendar network

Note that this metaphor has been chosen by the system because the sub-document contains no nested sub-levels. That is, it is possible to use this metaphor if the web document to be mapped on the calendar does not have nested links to other places.

The node related to "live concerts" also contains temporal information. In fact, the semantic network reveals a sequence of dates that recall temporal concepts similar to the ones for the calendar semantic net. In this case, due to the presence of nested links, the conceptual miner decided to adopt consecutive doors rather than utilize a calendar.

The system has also generates faults. For example the link "Villa Rufolo" referred to the place of the concerts was not correctly recognized.

The human interaction is needed, in order to provide additional knowledge about "Villa Rufolo" for which it has been adopted the building metaphor.

## 5 Conclusions and Further Research

In this paper we have presented the VR-Spider system, a novel architecture that allows a virtual navigation of the WWW. The VR-Spider integrates web mining and virtual reality technology in order to provide a user friendlier view of the Internet.

We aim to extend our system in different directions. First, we need to refine our current prototype. In particular we aim to introduce a user-profiling module that takes in account the user mental model during the navigation. We will further investigate on the use of immersive peripherals to interact with the derived virtual worlds.

The integration of XML with the VR-Spider will be investigated. For instance, we believe that the meta-tag mechanism can simplify the generation of metaphors.

Also, we ought to consider the interaction between avatars and the generated virtual worlds.

## REFERENCES


[1] Apple Computer Inc. Human Interface Guidelines "The Aplle Desktop Interface", *Addison-Wesley Publishing:Reasing*, MA,1987

[2] Arulnambi Kaliappan, "Browsing Virtual 3D scenes on the Web" *Technical Report at the University of Maryland Baltimore County (UMBC)*, in Spring 1995. http://www. cs.umbc.edu/gavl/vrmlviewer/arul/proj.html

[3] J. Balaguer and E. Gobbetti "I3D: A high speed 3D web browser", In john R. Vacca, editor, *VRML: Bringing Virtual Reality to the Internet* AP Professional, Boston, MA, USA, 1996

[4] S. Chaudhry "Web 3D- The Virtual Reality Browser" B.tech *project at Indian Institute of Technology,* Delhi http://csgrad.cs.vt.edu/~agol/ web3D.html

[5] M. Cibelli, G. Costagliola "Automatic Generation of Web Mining Environments", in proceeding of *SPIE's 13th Annual International Symposium Vol. 3639 Data Mining and Knowledge Discovery: Theory, Tools, and Technology*, Orlando, Florida USA, April 5-6, 1999, pp.215 -226

[6] Oren Etzioni, The World Wide Web: quagmire or gold mine? *CACM '96 survey on Web Mining,* November 1996.

[7] Fully Immersive Spherical Projection System, http://www.ndirect.co.uk/~vr-ystems/text/ sphere1.htm

[8] E. Gobbetti and R. Scateni "Interactive tools for navigation in 3D spaces " *In CRS4 Bulletin 1995, pp.80-84. CRS4, Centre for Advanced Studies, Research, and Development in Sardinia*, Cagliari, Italy, 1996.

[9] E. Gobbetti and R. Turner "Exploring annotated 3D environments on the world-wide web" In jim MayField and Charles Nicholas editors, *Intelligent Hypertext: Advanced techiniques for the World Wide Web*, volume 1326 of *Lecture Notes in Computer Science*, pp. 31-46, Springer-Verland Inc., New York, NY, USA, 1997

[10] G.H. Hovagimyan "Notes on Immersion" http://www.uni-mainz.de/~hergueta/ INTERNETdigest/march97.html,1997

[11] A. Johnson, F. Fotouhi "The SANDBOX: a Virtual Reality Interface to Scientific Databases" *Seventh International Working Conference on Scientific and Statistical Database Management, September 28-30, 1994, Charlottesville, Virginia, Proceedings. IEEE Computer Society Press 1994,* USA, ISBN 0-8186-6610-2"

[12] R. Jones, "A virtual Reality Interface for Structural Data", *Tectonic Studies Conference,* December 1995

[13] Jiyun LEE, Dongwook SHIN, "Multilevel Automatic Categorization for Web Pages", in Proc. Of *INET'98* 21-24 July, Geneva, Switzerland 1998.

[14] A. Massari, L. Saladini, M. Hemmsje and F. Sisinni "Virgilio: A Non-Immersive VR system to Browse Multimedia Database" in Proc. Of the *IEEE International Conference on Multimedia Computing and Systems 1997*, Ottawa, Canada, June2-6, 1997, Los. Alamitos: IEEE Computer Society Press-,1997, pp. 573-580.

[15] Mine, Mark "ISAAC: A Virtual Environment Tool for the Interactive Construction of Virtual Worlds"*UNC-Chapel HillComputer Science Technical Report TR95-020*. ftp://ftp.cs.unc.edu/ ub.technical-reports/95-020.ps.Z

[16] Mine. Mark "Working in a Virtual World: Interaction Techniques Used in the Chapel Hill Immersive Modeling Program" *UNC Chapel Hill Computer Science Technical Report TR96-029*

[17] Poupyrev I.,Billinghurst M., G Weghorst S. and Ichikawa T. "The Go-Go Interaction Techniques: Non-linear Mapping for Direct Manipulation in VR." In *Proceedings of ACM Symposium on User Interface Software and Technology 1997 (IUST 96),* pp. 79 - 80

[18] Poupyrev I., Weghorst S., Billinghurst M. and Ichikawa T. "A FrameWork and Testbed for Studying Manipulation Techniques" In *Proceedings of ACM Symposium on User Interface Software and Technology 1997 (VRST),* pp. 21 - 28

[19] G. Salton, and C. Burkley "Term-Weighting Approached in Automatic Text Retrieval" *Information Processing and Management, 24(5),pp.513-523*,1988

[20] R.C. Schank, *Conceptual Dependency: A Theory of Natural Language Understanding*, Cognitive Psychology, 3(4), 552-631, 1972.

[21] E. D. Smilowitz, "Do Metaphors Make Web Browsers Easier to Use" http://www.addesigus.com

[22] Virtual Sardinia Project http://www.crs4.it/PRJ/VIRTSARD

[23] Visual Expresso Project http://www.cs.may.ie/~dod/dcleary/index.html

# Steps toward Learning in Virtual World Cyberspace: TheU Virtual University and BOWorld

Bruce Damer, Stuart Gold
Contact Consortium
P.O. Box 66866
Scotts Valley
CA 95067-6866, USA
bdamer@ccon.org
www.ccon.org

Jan de Bruin, (assisted by Dirk-Jan de Bruin)
Tilburg University
P.O. Box 90153
5000 LE Tilburg, The Netherlands
J.A.W.deBruin@Kub.nl
www.virtualworlds.org

## ABSTRACT

This paper describes two experiments in building Virtual Worlds for educational use on the Internet. The main part of the paper details TheU Virtual University, an experiment conducted by Contact Consortium, which works from outside existing educational organizations. It deals with the collaborative construction of 3D multi-user online virtual campus spaces. The lessons learned may guide future efforts in that direction. In addition, we summarize a modest experiment, the BOWorld, carried out at a Dutch university. Both experiments underline the importance of using the concept of virtual (imaginary) organizations or systems in relation to the change and transformation of educational systems.

**Keywords:** interactive programs, Virtual Worlds, virtual communities, virtual organizations, imaginary systems.

## 1 INTRODUCTION

Since 1995, the Internet has played host to a dozen different emergent technologies supporting two- and three-dimensional multi-user graphical Virtual Worlds. These spaces have been shared by and bear the marks of tens of thousands of users who enter them daily, each donning a digital embodiment known as an 'avatar'. Avatar is an ancient Sanskrit term meaning 'a god's embodiment on the Earth'. Chip Morningstar first applied the term to online personification in the Habitat project in 1985 [8].

As we shall elucidate, the use of multi-user graphical Virtual World platforms can be an important aspect of transforming educational systems into what policy and organization sciences call virtual or imaginary organizations and systems [11]. The imaginary organization is a perspective which shows us how organizations, which can utilize imagination, information technology, alliances, and other types of networks, are able to organize and sustain a boundary transcending activity. [11: p.14]. With only a small core of employees, the imaginary organization can have, because of its vision, the employment of all kinds of networks and by being trusted by many involved individuals, an impact far greater than you might expect from its formal size. From a sociological point of view, you could say that the imaginary organization is based on integrative forces other than the traditional organization. Trust, synergy, and information technology are high on the list of those forces. Although information and communication technology (ICT) is not a prerequisite, normally ICT will facilitate the rise and growth of imaginary organizations.

An example of an imaginary organization is Contact Consortium (CCON), established in 1995. It acts as the focal point of a virtual organization, which serves as a catalyst within online Virtual Worlds, stimulating their evolution, and experimenting with practical uses for these shared spaces. Contact Consortium is especially focused on creating social/creative Virtual Worlds in the area of learning, rather than designing gaming or entertainment spaces. These worlds have typically relied on the inhabitants themselves to build up the community structure, economies, and visual virtual real estate, rather than presenting a more limiting content prescribed by a small set of designers. Contact Consortium is also an innovative force outside the formal educational systems, which seeks to transform education by using Virtual Worlds. This paper details the first phase of an experiment by describing the Virtual World project 'TheU Virtual University'.

Organizations that provide platforms such as The Palace and Active Worlds (AW) [6], promote the educational use of Virtual Worlds with low prices and special facilities, such as the offer to use a virtual space at no cost in a try-out period. Because of such incentives, and the relatively user-friendly technology, the proliferation of Virtual Worlds is a real possibility at all levels within existing educational organizations. The use of a Virtual World doesn't need to be part of the central ICT policy of the educational organization. In this paper, we document a small experiment at the level of departments within the university, namely, BOWorld, in the field of Policy and Organization Sciences at Tilburg University in The Netherlands.

This paper has three parts. The first part, sections 2 and 3, describe, in a rather general way, the roots and technical underpinnings of the online graphical multi-user Virtual Worlds. The second part, sections 4 and 5, document the two experiments. Section 4 details the experiment with 'TheU Virtual University'. It highlights two types of interaction processes: the cooperative activity of designing the architecture of the Virtual University and the way the activity of judging them is organized and hosted on the Internet. Section 5 summarizes the experiment with BOWorld. Here we focus upon the technical, as well as the social problems we encountered while searching for useful educational functions for the BOWorld. The perspective of the virtual or imaginary organization ultimately proved to be quite useful. The third part, section 6, concludes with some thoughts on the future of Virtual Worlds in cyberspace.

## 2 ROOTS OF THE MEDIUM

Virtual community finds its technological roots in the earliest text-based multi-user games, such as Space War, a popular application within the early development of Unix. Continuing this trend was the development of UseNET, LISTSERVs, MUDs, MOOs, IRC and conferencing systems like the WELL in the 1970s and '80s [16], and the World Wide Web and its many progeny in the '90s.

The merging of text-based chat channels with a visual interface in which users were represented as 'avatars' occurred first in Habitat in the mid-1980s [2] and reached an important watermark with the launch of the 3D Internet-based Worlds Chat in the spring of 1995.

Although one would suppose that the rise of 'inhabited' 2D and 3D visual spaces in cyberspace would have been heavily influenced by the prior example of Virtual Reality (VR) systems and be closely connected with the development of the Virtual Reality Modeling Language (VRML), this was not the case. Inhabited Virtual Worlds and their communities drew primarily from their roots in MUDs and text-based real-time chat systems, and they utilized the power of existing 3D rendering engines developed for gaming applications, such as Doom and Quake. Next, online Virtual Worlds do not take advantage of the full immersion and special devices of VR systems but instead concentrate on running effectively on a large range of consumer computing platforms at modem speeds.

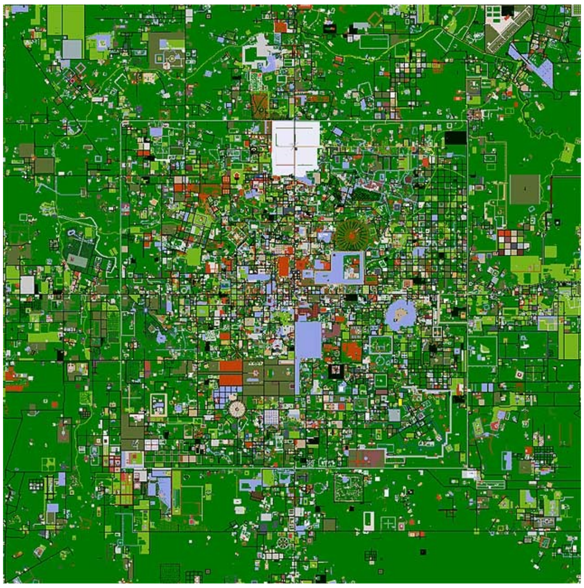


Figure 1: Generated 'satellite' view of the AlphaWorld cityscape, February 1998

## 3 TECHNICAL UNDERPINNINGS: HOW DO YOU BUILD A WORLD?

The technology involved in serving up an inhabited Virtual World experience is extensive and impressive: from robust client-server architectures, to streaming 3D object models, to tricks dealing with latency, to citizen authorization and crowd control, and finally to databases managing, and mirroring hundreds of millions of objects and thousands of users across networks at modem dial-up speeds. These virtual spaces represent one of the great architectural achievements of computing.

We invite you to view the image of one particular cityscape, referenced in figure 1. This 'satellite view' of the AlphaWorld cityscape is actually an artful processing of the database of 3D content that was placed down by some of 200,000 users in the first 32 months of operation beginning in the summer of 1995. Currently over 50 million objects occupy AlphaWorld, which can be visited by users with ordinary consumer computers 'walking at ground level', streaming in a visible periphery of 3D content over modem connections.

The literature surrounding Virtual World architectures, community development [4, 5, 6, 15] and avatar design [22] is comprehensive and growing, so we will not treat these topics further here.

# 4 TheU Virtual University Architecture Competition

## 4.1 Predecessor

The grandfather organization of the Contact Consortium is CONTACT: Cultures of the Imagination, a group of Anthropologists, space scientists, artists and writers, established in 1982.

CONTACT created projects in honor of Gregory Bateson, projects which involved learning through contact exercises with simulated cultures and their worlds. A team led by Northern Arizona University anthropology professor Reed Riner created one such Bateson project, SolSys Sim.

In SolSys, separate campus teams of students and virtual consultants designed and documented solar system human civilization projected one hundred years hence and then went live in a MUD text-based simulation of contact between these 'colonies' [17]. SolSys was highly successful as a learning environment, garnering international recognition and graduating scores of dedicated alumni over its eight years of operation.

## 4.2 Birth of an Experimental Virtual World:TheU

Starting with students of the University of Toronto's Marshall McLuhan Program in Florence (Italy) in May of 1996, a group of Contact Consortium set out to build on the legacy of SolSys by creating a learning space appropriate to a 3D online inhabited Virtual World. A dedicated Active World server, TheU, was donated to the Contact Consortium for special experimentation in education.

The goal of TheU is to serve as a test bed combining traditional campus-based universities and the growing number of distance learning projects. Distance learning offers many advantages to students in remote areas and students attending part-time courses. However, it lacks the social interaction and sense of community, which can be achieved by sharing environmental spaces and experiences. This technology may trigger the emergence of completely innovative teaching methods.

Figure 2: San Francisco State University students meeting in TheU University Development Center

Given its unique combination of social interaction, visual human embodiment and user definable virtual environments TheU sought to go beyond current networked learning experiments. Playing it safe for our first experiment, we opted to stay with familiar metaphors of a university campus with its instructional spaces, tutorial help centers, social commons, and library reference zones.

In order to utilize the power of the Active Worlds builder community, we decided to host a virtual architecture competition thereby generating a range of approaches to using a Virtual World in pedagogy.

Fortunately for the organization, Stuart Gold, a British architect and database expert with a long involvement in online systems, took the lead in this effort, operating the event as a professional juried competition.

## 4.3 Competition to Build the Institute of Virtual Education

### TheU

TheU was an experiment in the use of shared virtual spaces in distance education. The first step in developing that mission was thought to be the creation of a shared space which would support discussions on the application of Virtual Worlds technology in the field of education. We hoped that the Institute's role will be to:

- Stimulate and nurture debate on the development of virtual education.

- Provide a research facility and forum for the development of Virtual Worlds technology.
- Provide information and instruction on the use of virtual environments.
- Act as a governing body for TheU, developing and implementing policy decisions.

The first faculty was comprised of the panel of judges for the design competition. The panel was made up of educators, architects, artists/designers, and technologists including: Murray Turoff from the New Jersey Institute of Technology, Marcos Novak from UCLA, Derrick Woodham from the University of Cincinnati, Gerhard Schmitt from the ETH in Zurich, John Tiffin of Florida State, Bruce Damer, Stuart Gold, and others.

## Aims and Objectives of the Competition

Since the inception of Alpha World, its citizens have produced many stunning and innovative virtual spaces and structures. It therefore seemed appropriate to ask Alpha World citizens to take part in a competition, hosted and sponsored by SRT Enterprises and the Contact Consortium, to design the Institute of virtual education. The goal of the participants was to create a virtual environment which could best meet the following criteria on which it would be judged:

- Suitability and usefulness as a space to house and facilitate the functions of the institute.
- Innovative application of Virtual Worlds technology and other emergent technologies such as web casting.
- Use of space and objects to create a stimulating, aesthetically pleasing and compelling environment.

Participants were given a large degree of freedom in their interpretation of the requirements. As the driving technology is so fluid and is constantly evolving, we felt that the participants should evolve their ideas over the period of the competition and have access to a consultative panel of experts, who would also be the judges, via a listserv available to all the entrants.

## Competition Procedures

There were 34 teams; each was given an Active Worlds personal server donated by Circle of Fire and hosted by SRT Enterprises. Each of the servers was configured to allow any Active World user to enter but only one can build in the world. The duration of the competition was around six weeks.

Due to the open-ended nature of the requirements and the emphasis on ideology and innovation, the participants were encouraged to develop one or more Web pages, linked to their schemes, which served to describe their designs and the concepts behind them. Participants were encouraged to outline how their designs related to the learning and educational process.


Figure 3: Competition participants, judges and bystanders meet in the pavilion

A competition pavilion was constructed in TheU (figure 3) and served as a communal area for all the participants. In it were teleports through which visitors and competition judges could enter each of the worlds.

## The Competition: judging and prizes

After six weeks of intense building in all 34 competition worlds, six entries were short-listed for serious consideration. On March 20, 1998, the final walk through of the finalist worlds with their builders, the judges and a large group (40 to 50) of observers occurred (see figure 4). This event lasted almost four hours. With any in-world event, careful planning must occur to keep it from degenerating into chaos. The process that was conceived for this experiment should apply to any similarly conceived pedagogical Virtual World project. The following steps were taken to make this coherent, and obtain value for the participants:

1. The event had a clear structure, advertised from the beginning: meeting in the Pavilion, a tour of the six finalist worlds, with the judges, narrated by the builders of the worlds. Strict control of the pacing, moving on between each world every 20 minutes. A final gathering at the pavilion, voting and announcement of the winners by placing banners in the pavilion for all to see.
2. Assignment of gatekeepers and guides: we had trained world users at the pavilion to guide

newcomers and folks who crashed their software, back into the live tour. They used private telegram to stay in constant contact with the event leaders.

3. A pair of event leaders, in charge of herding of attendees, crowd control, discipline of event crashers, communications with gatekeepers and generally keeping the conversation interesting and on track.
4. Documentors were logging all text chat and taking a continuous series of screen shots of the walk through, which can all be seen at http://www.ccon.org/theu/album-background.html.
5. Voting of judges was by secret ballot, sent by telegram and email with follow-up after the event.

Figure 4: Competition judge, Professor Murray Turoff, standing outside the Learning Towers of Henrik, in the winning world Aurac

The winning world Aurac was built by Craig and Penny Twining of Active Arts Design in the UK working with Henrik, a chemical engineer and 3D model constructor from Norway. Figure 4 above shows the 'learning towers of Henrik' being viewed by judge Murray Turoff from the New Jersey Institute of Technology. These towers were built by Henrik to share with others the techniques of constructing objects in the Active Worlds environment.

Innovations in the winning world included amphitheaters with changing slide shows linked to websites, clear guidance paths and transport mechanisms allowing users to easily tour the world, and the tying in of triggered video and audio resources into the virtual spaces. All of these innovations have since been used to good effect in learning spaces in Active Worlds.

## What was learned

As TheU competition was seen as a following in the footsteps of the original Sherwood Town experiment, we felt we had achieved some major goals including: greater consistency of the event, clear, published goals, well defined roles and concrete outputs (the winning worlds and their documentation). Did we construct a viable Institute of Virtual Education? Apart from the great aesthetic value of the winning world, Aurac, and its merit for use in demonstrations, it did not serve our needs for the Institute.

The richness of TheU competition environments yielded their own problems: overbuilding created slow frame rate performance and the danger of a 'museum effect', in that the environments became static demonstration areas only. Three valuable principles emerging from this experience were:

1. Large spaces can cause users to get lost and provide a scarcity of immediate stimulating objects and other affordances to draw them on to the next activity. There needs to be a rich variety of objects close at hand, suggesting that smaller plazas function better than sweeping vistas requiring lengthy navigation.
2. A great deal of wasted objects and real estate can result if one is trying to be faithful to the real world setting. Do not design worlds that seek to model real world places, unless those places are particularly suited to support interaction in a virtual environment. Navigation and habitation of virtual spaces is so different than the same activities in a physical setting. Potemkin villages, theme parks, town squares or shopping malls, designed for denser crowding with plenty to see and do are some of the very few real world models worth emulating in virtual space.
3. Design worlds that are constantly changing and changeable. In fact, the 'ground zero' (default entry area) of TheU has become a major meeting area, filled with the changing detritus of signage, teleports and web links from prior events while the 'museum areas' are static and preserved only for narrative tours.

## Come visit TheU

Find TheU Virtual University on the CCON's Homepage at: http://www.ccon.org/theu/index.html. TheU is a frequently used, changing world that will continue to evolve.

Visit TheU on the Internet by downloading and installing the Active Worlds Browser http://www.activeworlds.com, starting the program while connected to the Internet, and then selecting

TheU from the listing of worlds on the left hand side of the interface.

Join the projects of the Contact Consortium by signing up on the many special interest groups lists at: http://www.ccon.org.

# 5 VIRTUAL WORLDS IN BUREAUCRATIC ORGANIZATIONS

## 5.1 BUREAUCRATIC ORGANIZATIONS

In section 4, the birth - and first phase of the development - of a Virtual University was described as the initiative of the virtual - or network - organization Contact Consortium. A complete statistical analysis of the characteristics of the participants, such as ages, level of computer expertise and professional background, is not available. From the interviews held with some of the participants, the following was clear: they were really enthusiastic and competent users of the Internet. If you read, for instance, the interview with a sixteen-year-old participant who calls himself a veteran Internet-user and programmer, it is evident that it was not the rank and file computer user, who entered the competition.

One can think of the Virtual University as an alternative, as Turoff has done, for the 'brick and mortar colleges'. He compared these alternatives in articles such as 'Costs for the Development of a Virtual University' (1982, 1996). In section 5, we look at the problems and promises of introducing and using Virtual Worlds in existing, primarily bureaucratic organizations. In such organizations there is a central ICT policy, a standard for the use of computer programs, and cost-sensitive replacement policies for computers, which results in technical specifications that are not state of the art. But even more important, real universities are not inhabited by the computer enthusiasts of TheU Virtual University Competition. The approach of using Virtual Worlds in bureaucratic organizations is also a realistic one and worthy of study.

In recent years, we have seen in The Netherlands, at all levels of the existing educational system, modest attempts to experiment with Virtual Worlds in relation to education in normal educational organizations. At the primary school level, De Wereldschool, an organization that provides long distance education for Dutch citizens who work abroad and whose children are unable to go to school, is right now doing a feasibility study to use a Virtual World as the primary interface between the educational center in the Netherlands and the dispersed pupils. At the secondary school level, a Montessori school in The Hague has taken the lead by using an Active Worlds' based Virtual World as a learning environment. At the university level, the Department of Policy and Organizations Sciences of Tilburg University experimented with BOWorld, also using Circle of Fire's Active Worlds as a platform.

## 5.2 THE BOWORLD

### Central ICT policy and the departments

Like all universities in the Netherlands, Tilburg University spends a lot of time and money on ICT. For the various departments, educational ICT is, however, something that is initiated at and comes down from the central level. It is, for instance, not unusual that departments don't even take the trouble to update their own web pages. The lack of interest of departments is one of the problems of implementing ICT policies in universities.

One incentive for the departments of Dutch universities to pay more attention to ICT in the educational process comes from the Dutch quality assessment system. The system, which has been in operation since the eighties, requires that every curriculum is reviewed ('the visitation') every five or six years. The use of ICT in their courses is one of the topics on which a curriculum is evaluated.

In 1997, the Committee on Education of BOW used the external impetus of the upcoming inspection of the curriculum of BOW at Tilburg University to found an educational committee (ELO) to promote ICT within its curriculum. Policy and Organization Sciences is a part of the faculty of Social Sciences. Normally, this type of department is not ICT oriented. Without the active involvement of staff and students the new activities would certainly fail. Therefore, ELO spent a lot of time on researching the needs and demands of the staff and students, to prevent resistance to the rise of ICT in the educational process.

## The first phase of the BOWorld project

The objective for the first phase was to build a cost-effective Virtual World to demonstrate the range of possibilities and to mobilize involvement for the further development of BOWorld.

The fact that a Virtual World could be bought at low costs – a mere 300 dollars for a one-year-subscription - and quickly, and therefore cheaply, could be constructed by using pre-fabricated models, was a surprise.

In the first phase, the BOWorld wasn't supposed to be a well-organized whole, but more like a bazaar, with a wide-range of possibilities. For instance, the fact that it was possible, by using a Frames Approach, to combine web sites and Virtual Worlds, was, for demonstration purposes, a valuable 'show piece'.

The third part of the objective was the most difficult one. An obvious approach to promoting the involvement of students - and staff – was to let them take an active part in building the world. Here we encountered our first major disappointment. Lack of time and low levels of ICT expertise prevented student participation in the project. A lot of our students, and also staff members, are only marginally interested in and experienced with ICT. One person, Dirk-Jan de Bruin, built the BOWorld.

In the first phase, we also encountered some technical problems. Because of the 'firewall', communication was only possible within the university. In 1998, this problem was solved when the Circle of Fire released Active Worlds 2.0, which used a UDP-protocol instead of a TCP- protocol.

Various other problems had to do with the fact that bureaucratic organizations have a lot of standard procedures and strict norms. The only place where a lot of students have access to the Internet is the central library of Tilburg University. Because of the central ICT policy, students could not install the AW browser at the library computers. Therefore, a large-scale experiment to demonstrate the interaction capacities of the 32-simultaneous-user-world could not be organized.

## The BOWorld and courses

One of the core activities of each department is their contribution to the working processes of the educational system: teaching courses. The first thing was to look for courses where simulation of interaction like you can achieve in a Virtual World could be employed.

BOW considered gaming and simulation as exercises for policy and organizational processes [9]. For precisely that purpose INSEAD, a school of Business Management, used 'scenes' built in the public Alpha World [1]. The most obvious point of entry for the application of BOWorld was the growing number of courses where gaming and simulation played a role.

The ELO committee discovered quickly that BOW, because of its small staff and many students, lacked the time to follow the example of INSEAD. Traditional methods for gaming and simulation, such as role-playing using written instructions and cardboard playing fields, are still in use in BOW.

Another point of entry was to look for an educational technology that can be combined with – or integrated within – the use of a multi-user graphical Virtual World environment. In BOW's courses, the Electronic Meeting Room (EMR) is sometimes used. This facilitates decision-making by using a local network of computers and groupware. Because everybody is in the room at the same time, participants can also interact face-to-face. Currently, however, BOW is not concerned with integrating the EMR-concept with a graphical Virtual World, or with the Internet as such.

In the literature, the case for learning through the Internet, with or without the use of a Virtual World, is always linked to two concepts: the concept of dispersion and that of asynchronicity. Both concepts are normally used in relation to distance learning. A lot of BOW students were, in fact, part-time students and some preferred distance education, because of the travel time to attend the courses. This did not mean, however, that BOW felt a need to invest in 'asynchronous learning networks' (ALN) [12]. In the end, neither the argument of dispersion of the students, nor the need for asynchronous learning, favored the use of a Virtual World.

The final conclusion after the first phase was that BOWorld could not yet play a role within the core educational processes of BOW. Are there other social functions related to the educational system which are appropriate for BOWorld?

## Metaphors and Virtual Worlds

When we think of Virtual Worlds, we must not take the notion of 'reality' too serious. In an earlier section, we claimed that various metaphors can be used to direct your thinking while designing a Virtual World.

The fact that the design process is, for instance, denoted as an architectural process, will influence your thinking by the metaphors it suggests, namely, the metaphors of space and place. Metaphors of space/place (room, campus, world) are, of course, only one kind of metaphor we live by [13].

If we use metaphors in the design process, we can go backward and forward between metaphors of structure – physical (space/place) or social – and metaphors of function. The metaphor ‘Virtual Classroom’ combines a space metaphor (room) with a metaphor of function (teaching courses). ‘Virtual University’ also combines a space metaphor (campus) with educational functions, although the relation is not as specific and on a different level than the metaphor ‘Virtual Classroom’. Some browsers prefer the metaphor of a ‘room’ (chat room), for instance WorldsAway and The Palace, while others, like Active Worlds, are not so committed to that metaphor.

In Virtual Worlds, normally a person (avatar) has a fixed aura around him or her. This is an awareness mechanism which allows persons at certain ‘distances’ to see and talk to each other. Awareness is a prerequisite for interaction. Sociologists also find it important to have a seclusion mechanism. In cases where the room-metaphor is used, this mechanism is more or less built in the basic design. In a fundamental ‘open society’ like Active Worlds, social seclusion must be achieved in other ways. In short, if you focus on a certain type of educational function, it will steer your architectural design and eventually the type of Virtual World browser in which a certain metaphor is incorporated.

At the course level, the room metaphor is very suitable. The metaphor ‘Virtual University’, designed as ‘an open society’, can fulfill internal and external functions. ‘Virtual reality’ is, as the sociologist Durkheim would call it, a ‘reality ‘sui generis’’ – a reality of its own. This applies, of course, also to the ‘Virtual University’.

We can consider the ‘Virtual University’ as a manifestation from an underlying social reality, namely the ‘virtual organization’. In our introduction, we linked in that vein the virtual organization Contact Consortium to TheU Virtual University.

It is now time to expand the argument and to link it to BOWorld.: if a world can not be used in the internal, primary processes, look for external and boundary-transcending functions and interaction processes. Our question is, can BOWorld be used as an instrument of public relations and as a cohesive force in the information processes directed at future students?

## A potential second phase for BOWorld?

The virtual organization builds a boundary-transcending involvement based on vision (ideology), trust, and loyalty as cohesive forces, in addition to the classical utilitarian motives.

An attempt to build and maintain such cohesive networks with persons outside the university focuses on two categories. The first is the category of the alumni. Universities have always invested in that category. The other is the potential student. In the Netherlands, it was, till the eighties, ‘not done’ to influence the choices of potential students. Nowadays, a lot of marketing techniques are being used. Besides advertisements and informative meetings at the university, universities also employ techniques to establish a relationship before students enroll. Sometimes, potential students may use the subsidized sport facilities at the university a year before they are officially enrolled.

Figure 5: The Virtual Cafeteria of BOWorld with live video broadcasts on big screen televisions

ICT is also an instrument in the Public Relations battle. In addition to incidental activities, there is always the possibility for potential students to access the university’s website. As an instrument of public relations, the problem of the website is the lack of interactivity. To make a more exciting impression, Tilburg University experimented with a kind of ‘Virtual University’ on a CD-ROM. But why not use a Virtual World?

Even a modest Virtual World like BOWorld has a lot of interesting features. If we use metaphors while thinking about the university, ‘ivory towers’ comes easily to mind. In contrast to that metaphor, a Virtual University - or in the case of BOWorld a part of it - is a very open, friendly, and inoffensive place. The avatars, looking like figures from a cartoon, lower the level of entrance to the Virtual University, the manifestation of the 'real world' university, even further. This is totally different from that other famous metaphor of a Virtual World, Plato’s ‘psychic prison’ [14: p.199]. From inside the Virtual World, websites and other means of communication can be accessed. So the world of fun, chat and exploration can be combined with serious information.

On the basis of the above considerations, we planned the second phase for 1998. BOWorld would become an additional instrument in the BOW toolkit to attract potential students and to create a kind of relationship on an ongoing base with potential students. Our intention was to do the additional building in collaboration with a few selected secondary schools. We planned to monitor the process intensively. We totally agree with Schroeder [18] that empirical research of Virtual Worlds is necessary, but also difficult.

Figure 6: Screenshot of the test bed we built for the second phase of BOWorld

However, in 1998, the faculty was involved in a very large-scale reorganization and further activities connected with BOWorld were temporarily suspended.

### What was learned

The lessons learned complement each other in the two experiments. The case study of the TheU Virtual University taught us some lessons about rather long sequences of interaction within a Virtual World, such as building a world and hosting an event, and some design rules for their architecture. The BOWorld case study shifted attention to the social acceptance of a Virtual World within and outside the university, highlighting the social problems and promises in relation to the dissemination and utilization of Virtual Worlds.

From the second case study, we learned the following:

1. The low cost of Virtual Worlds, employing browsers like Active Worlds etc., permits you to design your Virtual World for a specific public. If you do 'consumer research', the special interests of the target group can guide the design.
2. If you take your target group seriously, you must keep in mind their resistance to change and the relatively simple ICT environment with which they are familiar. Choose a simple platform for your world, which will work within the existing ICT environment. The argument that a certain Virtual World will only perform well within a changed ICT environment does not hold up in the dissemination and utilization game.
3. If you introduce a Virtual World, a safe procedure is to go from the periphery of the educational process, for instance by using it as a social interface between the university and potential students, to the core processes, the teaching of courses. Potential students belong to a social category with a lot less resistance to Virtual Worlds than the average teacher. Also, the tolerance for shortcomings in the design and operation of Virtual Worlds is greater if we start with the more peripheral educational processes.
4. The constructivist approach makes us aware of the ‘generative metaphors’, which are an intrinsic part of the design process. The fact that we are now moving from text-based Virtual Worlds to worlds with 3D-graphics will only make the metaphorical component in the design process more important. The metaphors of place (space, room, and campus), seen as metaphors of structure, should be, however, connected with those of function.

## 6 THE FUTURE OF THE MEDIUM

Inhabited Visual Virtual Worlds are a medium in search of an application, but also a phenomenon worth of scrutiny. And, as always, maybe that empirical research, by tracing intended but also unintended consequences, can be of assistance in guiding the development, in so far as a technological process ever can be guided. The history of technology assessment confronts us with a well-known paradox: if an important technology is still in its infancy, we can steer it, but we don’t know much about it; if we know a lot about the technology, it is too late to direct its development [3].

At the July 1998 Avatar conference in Banff, Alberta, Canada, consensus emerged that it was too early in the medium to know how it would ultimately be used. It was felt that it was good that a “killer app” had not been identified and that “avatar cyberspace” had time to continue to evolve for its own sake and not to serve possibly inappropriate applications. Comparisons were made with the birth of important technological media of the past century. The telephone was first thought of as a way to distribute music, early film was first cast as a facsimile of theater, and the radio was considered as a method for the delivery of lectures and person-to-person two-way communication between communities.

Virtual Worlds are currently split into multi-million dollar efforts in multi-player gaming and smaller social creative spaces supported as research efforts or by die-hard builders of home-brewed digital spaces.

Millions of computer users have only recently acclimatized to using classic windows and icons desktop metaphors. Those users are now comfortable in dealing in a cyberspace made up of lists of text, and documents in the form of the Web. Another generation, brought up on Doom, Quake, Nintendo 64, and other environments that stress navigation through very complex, 3D spaces full of behaviors, may be more apt to demand a cyberspace that is built around the metaphor of a place, not just an interface.

Will that generation bring us more into Virtual Worlds for play, learning, work, and just being? What will cyberspace look like in ten years, like Gibson's Matrix or Stevenson's Metaverse [10, 19] or will the document-based web and streaming video and audio spaces be the dominant paradigm?

But in the end, the most important question is: How can – and will – our institutions adapt themselves to the technological prime mover? In this paper we took a rather optimistic stand. Virtual Worlds are not Weber's 'iron cage', nor Plato's 'psychic prison', but a rather liberating form of technology. In relation to institutional change, we connected the development of virtual - or imaginary - organizations with that of Virtual Worlds.

## REFERENCES


[1] Angehrn, A., and Nabeth, Th., Leveraging Emerging Technologies in Management Education: Research and Experiences, in *European Management Journal,* p. 275 – 285, 1997.

[2] Benedikt, M. (ed, *Cyberspace: First Step*, Cambridge: MIT Press.), 1991.

[3] Collingridge, D., *The Social Control of Technology,* New York: St. Martin's Press., 1980.

[4] Damer, B., Kekenes, C., Hoffman, T., Inhabited Digital Spaces, published *in ACM CHI '96 Companion*, page 9, 1995.

[5] Damer, B., Inhabited Virtual Worlds, in *ACM interactions*, page 27, sept-oct 1996.

[6] Damer, B., *Avatars!: Exploring and Building Virtual Worlds on the Internet*, Berkeley: Peachpit Press, 1998.

[7] England, D., and Gray, Ph., Temporal Aspects of Interaction in Shared Virtual Worlds, in *Interacting with Computers*, p. 87 – 105, 1998.

[8] Farmer, R., The Electric Communities Habitat White Papers, on the web at: http://www.communities.com/company/papers/index.html, 1991

[9] Geurts, J., Joldersma, C., Roelofs, E. (eds, *Gaming/Simulation for Policy Development and Organizational Change*, Tilburg: Tilburg University Press.), 1998.

[10] Gibson, W., *Neuromancer*, New York NY: Ace Books, 1984.

[11] Hedberg, B., Dahlgren, G., Hansson, J., and Olve, N-G., *Virtual Organizations and Beyond: Discover Imaginay Systems*, Chichester: John Wiley & Sons, 1997.

[12] Hiltz, R., Impacts of college-level courses via Asynchronous Learning Networks: Focus on Students, on the web at: http://www.njit.edu/Virtual_Classroom/Papers/Async_Learn/Philly.html, 1995.

[13] Lakoff, G., and Johnson, M., *Metaphors we live by*, Chicago: University of Chicago Press, 1980.

[14] Morgan, G, *Images of Organization*, Beverly Hills: Sage Publications., 1986.

[15] Powers, M., *How to Program Virtual Communities, Attract New Web Visitors and Get Them to Stay*, New York: Ziff-Davis Press, 1997.

[16] Rheingold, H., *The Virtual Community: Homesteading on the Electronic Frontier*, New York NY: Addison-Wesley, 1993.

[17] Riner, R., NAU SolSys Sim, on the web at: http://www.nau.edu/anthro/solsys/, 1994

[18] Schroeder, R., Heather, N., and Lee, R.M, The sacred and the Virtual: Religion in Multi-User Virtual Reality, on the web at: http://jcmc.huji.ac.il/vol4/issue2/schroeder.html, 1998

[19] Stephenson, N., *Snow Crash*, New York NY: Bantam Spectra, 1992

[20] Turoff, M., Costs for the Development of a Virtual University, on the web at: http://eies.njit.edu/~turoff/Papers/cbdevu.html, 1996.

[21] Turkle, S., *Life on the Screen: Identity in the Age of the Internet*, New York: Simon and Schuster, 1995

[22] Wilcox, S., *Creating 3D Avatars*, New York: Wiley, 1998.

## Biographical Information

Bruce Damer and Stuart Gold are members of the Contact Consortium. Damer co-founded the organization in 1995 and Gold headed up TheU Virtual University projects since 1996. Dr. Jan de Bruin is a policy scientist at Tilburg University. He has been chair of the Committee on Education of BOW for 15 years. Dirk-Jan de Bruin is a member of Contact Consortium and founder of Multi-Users Virtual Worlds Co

nsortium.

# Virtual Worlds Applications

Russell Eames
Microsoft Research
Seattle, Usa

## ABSTRACT

The focus of the Virtual Worlds Group in Microsoft Research is social interaction. This talk will cover recent projects by the Virtual Worlds Group that demonstrate how people will use the computer to connect one another in new and compelling ways.

One project that will be demonstrated is the Virtual Worlds Platform. This is a research platform for developing technology that facilitates the creation of multi-user, distributed, applications on the Internet. It is being designed as a general-purpose system. The Virtual Worlds platform uses Microsoft's ActiveX and DirectX technologies and allows great flexibility in the design of user-interfaces and the support of multimedia.

The Virtual Worlds Platform provides a synchronous, persistent, distributed object system in which users can easily program and prototype shared environments. Changes that are made by one user in an environment can immediately be perceived by other users of the environment. The platform supports multiple media types including text, graphics, voice and video, and dynamic end-user object creation with behaviors. A variety of shared environments have been built ranging from immersive 3D environments where people can see and "directly" interact with other people, to 2D environments that have synchronous communication within a virtual classroom.

Recognizing that virtual worlds applications will include a variety of experiences, we have architected our system in such a way that the core technology infrastructure is separate from the interface, allowing for very diverse and customizable interfaces. These interfaces can be tuned to specific uses of the technology. Additionally, the core technology is flexible and extensible, enabling developers to create enhanced functionality.

# Creating Virtual Worlds with a Graspable User Interface

Hauke Ernst,Kai Schäfer, Willi Bruns
Research Center Work and Technology (artec)
University of Bremen
Enrique Schmidt Str. (SFG)
28334 Bremen, Germany
{ernst | schaefer | bruns}@artec.uni-bremen.de

## ABSTRACT

We introduce a system to interact with computers using real objects as interface. The graspable model on the table has a congruent counterpart in the computer which is updated continuously. The use of concrete objects offers advantages for interaction and cooperation. The underlying virtual model provides capabilities like remote visualization, simulation and documentation which play an important role for processes in engineering and education. System implementation, interaction concepts, projects as well as the impact on distant cooperation are introduced in this article.

**Keywords:** real reality, graspable user interfaces, image recognition, data gloves, multimodal access, cooperation

## 1 INTRODUCTION

This document gives a description of selected aspects which become interesting when computer representations and its functionality are automatically allocated to physical models. This is of relevance to our research projects where we use physical models as an intuitive interface for computer based modeling. We introduce two of our applications that use this concept in two different scopes and explain afterwards the technologic basis of our systems including an overview over the system architecture. Moreover, we will compare real world interactions with language processing. Additionally, we will identify open questions for the application of the concept in distant cooperative environments and propose mechanisms for solutions.

By bridging the gap between the real and the virtual world we are in a position to overcome some of the difficulties which obstruct the computer based modeling. This is generally characterized by its interface technology like monitor, keyboard and mouse, in Virtual Reality applications Head Mounted Displays, Data Gloves and other n-dimensional input devices. These techniques in general obstruct communication, don't allow direct grasping with appropriate sensual feedback and deliverer a more or less incomplete representation of the model to the user. The application of concrete components gives the user the feeling and the experience of the real world. The use of 3D visualization and schematic representations creates links between the concrete and abstract concepts required for learning. The application of computer technology means that for instance visual, schematic and acoustic information can be made available to the user. Once we have generated a virtual world we can attach properties to the objects which allow us to simulate their functionality within the virtual model or to export them to off-the-shelf simulators. In order to achieve a bridge between these different media a new interface is required which is able to translate the construction of a concrete world into an abstract representation. This combines a multimodal access to virtual environments with a normal communication between people in the real world.

We use two different technologies to synchronize the position of the real objects with the positions of the virtual ones. The first one applies image recognition to detect and locate the real world objects using simple, fast and robust technologies. The second one uses magnetic trackers attached to the users hand. Monitoring the actions of the hands allows the synchronization of the position and also the movements of real and virtual objects. This approach allows the realization of modeling and demonstration processes for creating and programming virtual worlds. We summarize these concepts under the name *Real Reality*.

## 2 APPLICATIONS

### 2.1 APPLICATION IN VOCATIONAL TRAINING OF PNEUMATICS

After a couple of national projects which provide the theoretical background and the basic technology for our *Real Reality* concept we are now in the second year of the more product-oriented European (ESPRIT and

Leonardo) project BREVIE: Bridging Reality and Virtuality with a Graspable User Interface.

The BREVIE project aims at designing, developing and evaluating a new kind of teaching and learning environment in the field of vocational training in pneumatics. The main characteristic of this environment is to provide transitions and links between two worlds which are currently separated:
the physical pneumatic circuit which can be grasped and touched with our hands, and the world of abstract models which can be viewed on paper or on a computer screen. The links are achieved by a "Graspable User Interface" which enables students to work and learn in both of these worlds (figure 1).

In the vocational training for pneumatics several kinds of learning material for teaching can be found. For the project consortium, the most compelling is the very popular pneumatic construction kit of our partner Festo Didactic which can be used for building functional pneumatic circuits. These circuits work with compressed air and allow a very close representation of real production environments.

The project makes use of these construction kits and develops technical links - through the Graspable User Interface - to computer-based learning media.

Figure. 1: Teaching Situation with the BREVIE ModellingDesk

the possibility to interact with the model and to manipulate it:

- The physical model (figure 1)
- The virtual model in 3D (figure 3)
- A symbolic/logical view in a simulation software for pneumatics (figure 3)

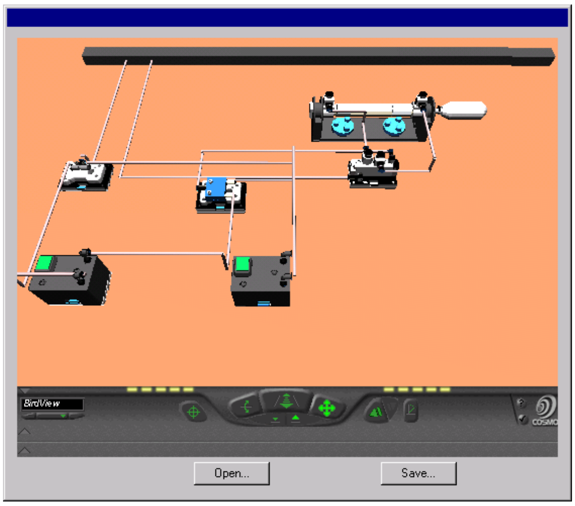



Figure 2: 3D-View of the Virtual Model within the BREVIE Learning Environment

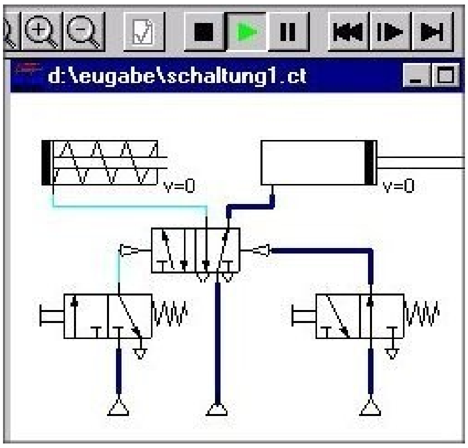



Figure 3: Working with the Model in a Simulator for Pneumatics (FluidSim)

In the BREVIE project we use two low cost video cameras and image recognition algorithms to synchronize the physical model with a virtual model which is the basis for the processing on the computer. This automatic link between the real model and corresponding computer representations provides smooth transitions between multiple views on the same model which supports the development of rich mental models. In BREVIE we have integrated the following views, each with

The BREVIE Learning Environment also provides access to multimedial learning material which introduces the functional properties of the pneumatic parts (figure 4). The user can intuitively call up this information with the help of a pointing stick which enables him to select components within the physical model.

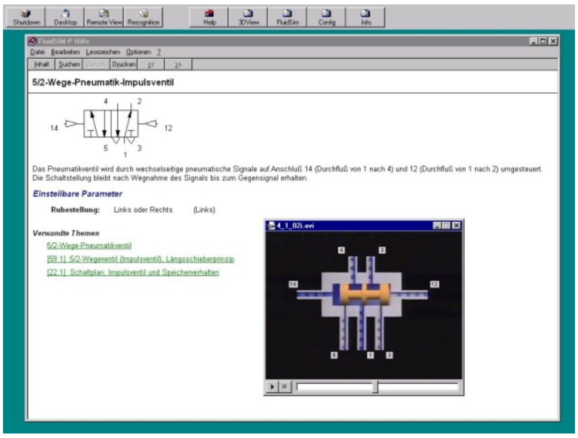



Figure 4: Presentation of Learning Material in the BREVIE Learning Environment

## 2.2 APPLICATION IN FACTORY LAYOUT

In our latest completed project RUGAMS we applied the *Real Reality* Technology to the design of factory layouts and to the programming of plants like conveyor systems and robots (figure 5).

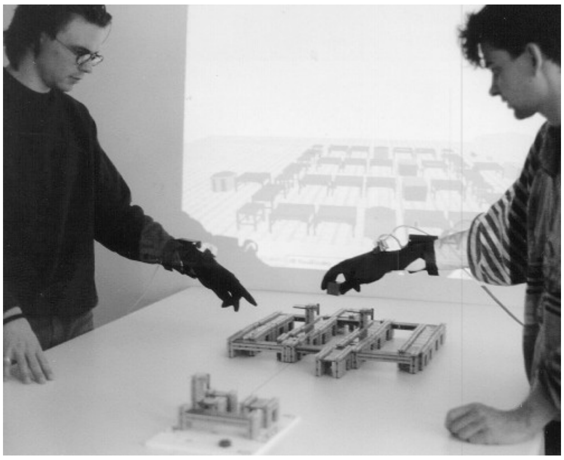


Figure 5: Co-operative Modelling and Simulation of an Industrial Plant

In this context we configured the modeling environment for interdisciplinary teams to plan layout and material flow in complex factory scenarios. Therefore, we constructed an appropriate set of factory equipment like conveyors, plants, buffers and junctions presented in an object box (in the foreground). Corresponding virtual objects were been defined to represent geometry and behaviour of the elements. The group plans and discusses the layout in the modeling session. The computer in the background keeps track of the modeling process, by generating a virtual model of the plant. This enables a simulation applying the behaviour stored in the virtual components. A video beamer projects the activity of this digital factory model in the background to provide a feedback to the planning group. In later versions the simulated dynamics are projected directly onto the objects in the scene using Augmented Reality Technologies (figure 6). This helps to provide a better context to the model.

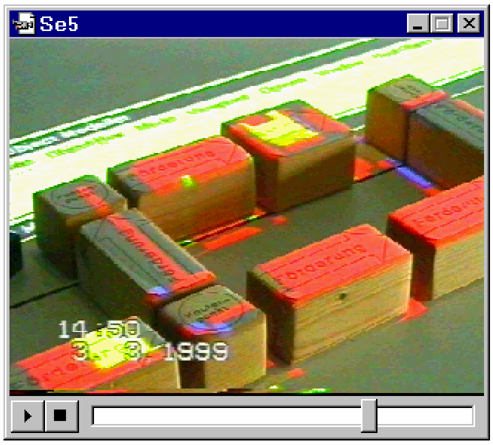



Figure 6: Augmentation of a concrete Model to visualize the Dynamics of the Simulation

Later in this article we will describe in what way the predefined behaviour of the objects can be influenced when using programming by demonstration techniques. These demonstration techniques allow the specification of the material flow which optimizes the system performance. This environment supports a substantial part of the planning tasks within a group of decision-makers in just one session. This reduces the time to market significantly which is one of the industries' major goals.

# 3 TECHNOLOGY

## 3.1 MODEL SYNCHRONIZATION

### 3.1.1 Grasp Recognition

Bruns 1993 et al [1] laid the foundation for a new class of user interfaces in shop floor and handicraft working. The main characteristic of the "*Real Reality* User Interface", as they called it, is the application of the user's hand as a manipulator of physical objects in a real environment. Appropriate interface devices like data gloves and tracking systems capture the user's hand movements and finger flexions. Gesture recognition algorithms analyze the raw interface data and recognize gestures, grasps or user commands in real time [2]. Working with physical objects while being linked to a computer has a certain analogy to the well known Drag & Drop principle of GUIs. When the object is grasped all following data of the Drag-Phase is recorded. This phase terminates when the user puts the object to another location and releases it (Drop). Now, the physical object has a new position and due to this the virtual computer model of the physical environment has been updated. The system will trigger a grasp event, if a grasp gesture together with a collision between index fingertip and the boundary box of a virtual object is detected. Stopping a grasp gesture triggers a release event. Giving the user an acoustic feedback in the moment of grasping and releasing, the graphic output on a display becomes obsolete. So the user can work independently of the encumbering aura of the monitor, the keyboard and the mouse.

The term *Real Reality* emphasizes the difference to the term *Virtual Reality.* The user immerses in the Virtual Reality and is surrounded by the interface. *Real Reality* means to remain in the real world and to experience it. All human senses are stimulated and communication within groups is encouraged. The interface becomes a passive observer and is ideally not noticed by its users. We achieve this by linking physical objects to their virtual counterparts. As our physical objects always have a virtual counterpart they are called „*Twin Objects*“. In this way the description of actions effecting two model representations becomes much easier.

## Preparing Real Reality Modeling Sessions

The *Twin Objects* are one of the basic elements of the *Real Reality* concept. For both kinds of object representations a number of instances must be available. This means to create a virtual representation consisting of the object's geometry and attributes describing the dynamic behavior. The geometric description contains the object's size (length, width, height) and its surface shape. On the other hand, the physical objects may be constructed by using technical construction kits, wood or other materials. The object representations may vary in shape, size and level of detail.

After putting on the data glove, its bending sensors must be calibrated by opening and closing the hand. Then the user places the hand on the tabletop which serves as the model ground, and the zero position is determined by the *Real Reality* system software. Now, the user's hand position is defined relative to the origin of the model's frame of reference.

In the initial state, the objects are located in an object box which has a predefined position on the tabletop (figure 7). Thus, for each object in the box the position can be computed.

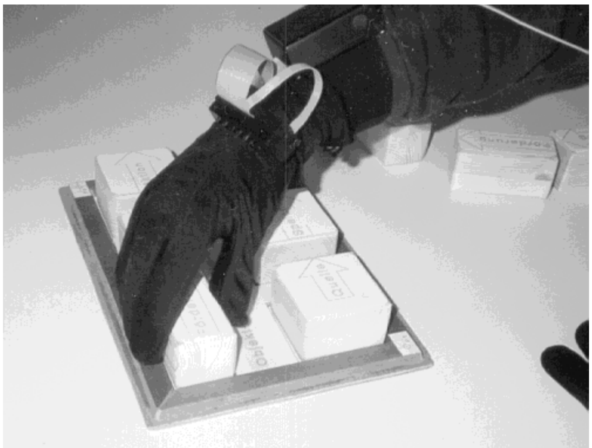


Figure 7: Object Box

## Using the Real Reality Modeling System

After the preparation of the modeling elements and the data glove, the *Real Reality* software is initialized and ready for use. A model is created gradually by taking *Twin Objects* out of the object box and putting them on the model ground. As two models are handled synchronously, the *Real Reality* system provides two views on the same model. With the help of 3D visualization software, the virtual representation is displayed and animated on a monitor screen, projected directly onto the model table or observed remotely via a network. Although the visual feedback is not necessary for persons who model with the physical objects, it is used for replaying actions recorded during a modeling session.

Figure 8 shows the virtual representation of a hand reaching for a *Twin Object* contained in the object box, in this case a conveyor. It is taken out of the box and placed in a location near another conveyor which was inserted into the model during a preceding step (figure 9). Some other forms of interaction are provided. By pointing at a specific object the user gets access to information about it, while creating a model. The information is shown on the display. For the future, voice output by the computer are planned.

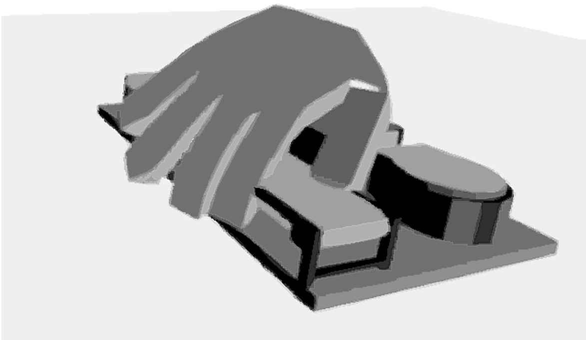


Figure 8: The Virtual Hand grasping a Twin Object

The virtual model is represented on a scene graph in VRML97 Format. This is a dynamic representation of the real scene. Real-time analysis permits the filtering out of relevant events like collision, connnectings and separations, pile up elements and characteristic object motions. Saving the scene graph in a VRML97 File permits subsequently analysis, and serves as a documentation of the modeling process which can be recalled and graphically animated later.

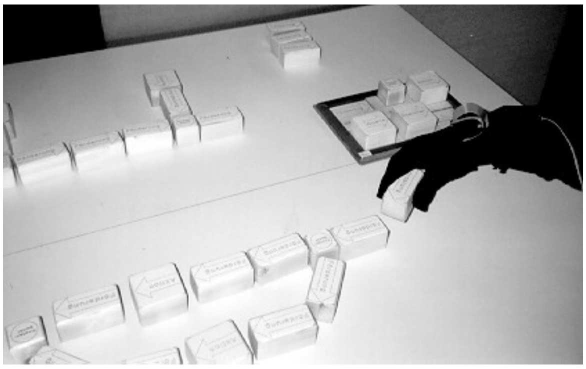


Figure 9: Building a Model with Twin Objects

### 3.1.2 Image Recognition

In our latest research project (BREVIE) we decided to use image recognition methods to recognize actions in the real world because of financial restrictions. The system is designed for the application in schools which cannot afford data gloves equipped with tracking systems.

Therefore we developed a new software module which is capable to analyze images that are delivered by two low-cost CCD cameras (s. figure 10) and to recognize the components which are placed on the base plate by the user. The software is also able to recognize the pneumatic tubes which connect the components.

Figure 10: CCD Camera

The output of the recognition process is also a VRML97 definition of the model which can be processed with 3D viewers and be converted to formats which off-the-shelf simulators can import.

For the first version we use color-barcodes which are mounted on the pneumatic components (s. figure 11) and the tubes. the barcodes turned out to be unfavorable for the commercialization of the system, we currently investigate in possibilities to exchange the unwieldy barcodes and look after techniques based on neural networks instead.

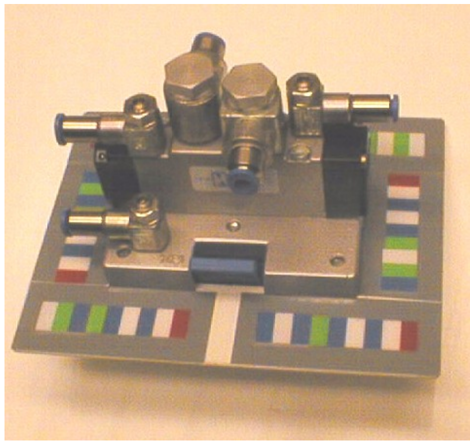


Figure 11: Pneumatic Component tagged with Barcodes

## 3.2 ARCHITECTURE

In our *Real Reality* systems we integrate several hard- and software components into coherent applications. Since every application field requires a special combination of components we have introduced a highly modular system architecture which permits us to connect easily existing or new components. The design follows a client-server approach with a *Real Object Manager (ROMAN)* as the central server unit (s. 3.2.2). Other software units are connected to ROMAN via tcp/ip based Socket protocols. The following diagram outlines the spectrum of system components which are currently part of our applications or in our research

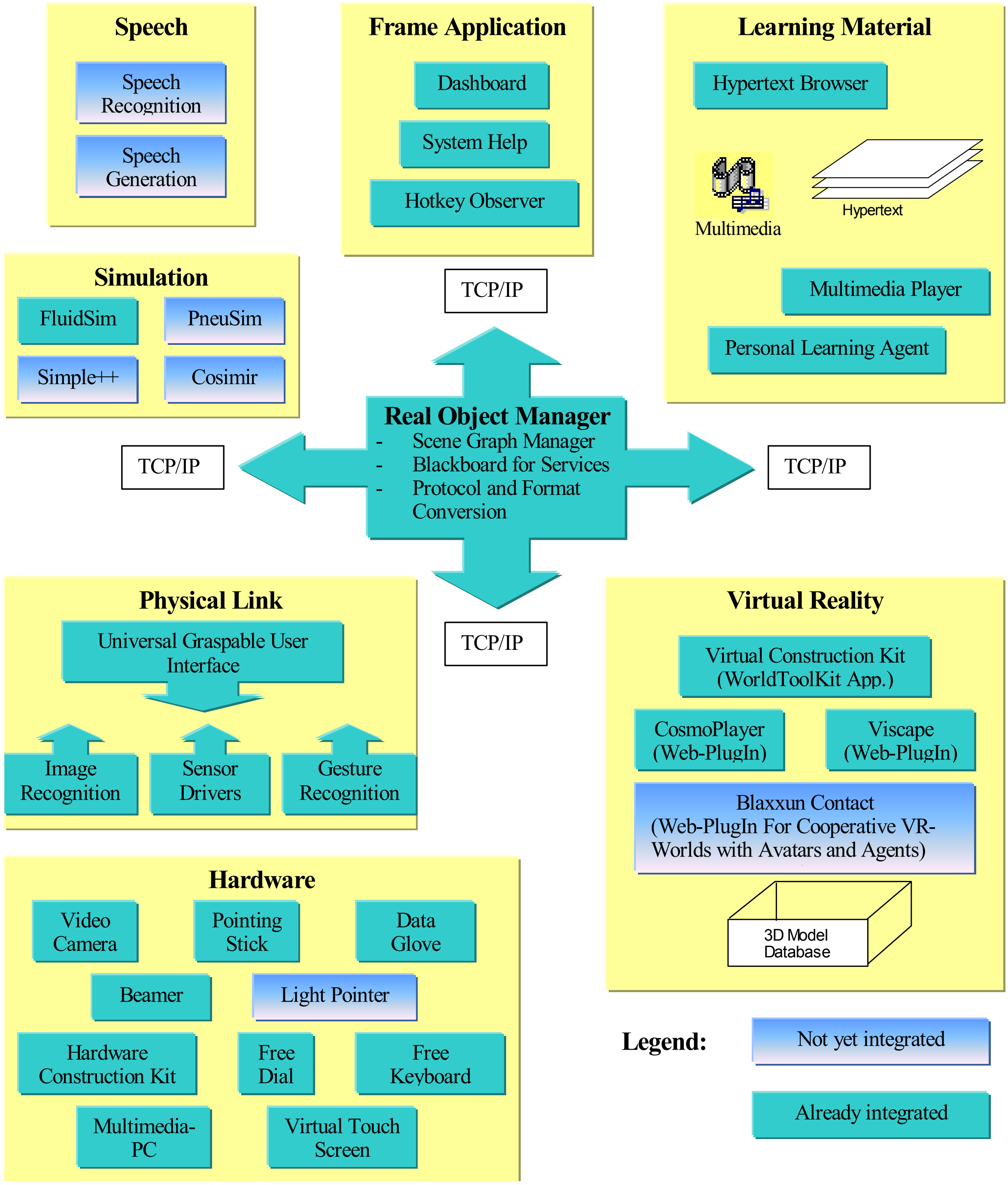



Figure 12: Overview of the Real Reality Technology

focus including

- Interfaces to several simulators for pneumatics, electronic, robotics and mechatronics
- Interfaces to Virtual Reality systems
- Integrated learning material
- Interfaces to speech recognizing/generating packages
- A *Universal Graspable User Interface* as an abstraction layer with a plug-in interface for software plug-in modules which implement the link from the physical to the virtual world
- A couple of hardware devices for the interaction between the user and the physical model.

### 3.2.1 Special Hardware Devices

Some of the devices which are shown in the diagram-box "Hardware" may need additional explanations:
The *Grasp Tracker* is a new device which can replace the data glove. It is a ring worn at the index finger tip (figure 17). The tracking system is mounted on top of it, so that its relative position to the grasped object is fixed. The underside is covered with a pressure-sensitive foil for triggering grasp events. Grasp and pointing gestures can be distinguished by inclination. If more different gestures are desired, the *Grasp Tracker* can be worn together with a data glove (figure 17).

*Free Keyboard* and *Free Dial* are devices for data manipulation within the *Real Reality* environment. Although it is a pure fake device with no functionality and cable connection, it is possible to enter data with the sensorized hand. At the *Free Keyboard* keystrokes are detected through collisions with the virtual keys. The *Free Dial* works like the Shuttle Jog or Jog Dial known from video recorders and other technical devices. An actual value is attached to the actual dial position. Turning the dial permits a variation of the value between lower and upper limit. A sensorization of the dial is not necessary because all changes occur relative to the arbitrary starting position.

*Virtual Touch Screen* permits the interaction with a projected Windows Desktop without a sensorized surface. The mouse position is calculated from the relative position of the tracking sensor to the projection area. Click events are raised through the touch sensor at the *Grasp Tracker*.

### 3.2.2 Real Object Manager (ROMAN)

**ROMAN as an Object Manager**

ROMAN is the server unit in our client-server architecture. ROMAN coordinates the access to the virtual model data, thus ensuring data consistency. The other software components are connected to ROMAN as clients via a Socket link and do access or modify the model through this component by using protocols with a defined set of commands. The model is internally represented as a hierarchical scene graph which consists of the nodes and attributes.

The network connectivity is realized with tcp/ip stream Sockets. Concerning the network protocol ROMAN itself implements only a basic transport layer for text transmission. Higher protocols are added as plug-in modules.

**ROMAN as a Service Coordinator**

Based on the Client/Server techniques it is possible to build up a network of specialized software agents which offer services to each other (and in the end to the user). ROMAN treats the agents as clients and manages the services for each agent. In this context ROMAN is responsible for taking care of the message routing between the agents.

ROMAN manages services by means of a dynamic routing table for events. Clients can access this table via the network protocol. They can create events as well as subscribe or unsubscribe to them. On the other hand clients can raise events: ROMAN checks which clients have subscribed to the event and routes the event to these clients. Events can transport arbitrary textual information.

ROMAN as a pure manager of events and services takes a back seat from the users point of view. Since the ROMAN's graphical user interface provides only monitoring information which is merely interesting for developers ROMAN will become invisible if installed as a release version.

**Plug-ins**

Aiming at a maximum generality ROMAN offers a plug-in interface for network protocols, format parsers and generators etc. This technique results in a strict modularity of the software and an optimal extensibility concerning further investigation and the integration of other products.

**Network protocols as Plug-ins**

One important application of the plug-in-interface is the implementation of network protocols as plug-ins. This mechanism enables ROMAN to coordinate clients with different protocols or different protocol versions. Existing client applications with existing protocols can, in this way, be integrated with minimal effort: it is only necessary to develop an interface plug-in which implements the protocol to fit the application to the ROMAN.

**Parsers as Plug-ins**

The support of different exchange formats for the definition of virtual models can be realized as ROMAN plug-ins, too. Such parser-plug-ins assist ROMAN in reading serialized model descriptions into the internal scene graph and vice versa.

As ROMAN contains a scene graph representation which is independent from concrete formats it can use the plug-ins to convert freely between the formats being provided by the plug-ins.

Other ROMAN plug-ins like the protocol-plug-ins are also able to make use of parser plug-ins to easily fulfil the clients' format wishes.

In our projects we use mainly the VRML97 format.

### 3.2.3 Frame Application (Dashboard)

The Dashboard is the frame application of our *Real Reality* applications. It can be used in two different modes:

- The **configuration mode** allows a developer or a system administrator to configure what will happen in the execution mode. These configurations can be saved as so called start-scripts which may be installed as links on the Windows desktop or the Windows startmenu. The start-scripts comprise the definition of all programs involved as well as their start-parameters and dependencies. Furthermore function buttons and buttons to switch between different views can be specified. The configuration mode of the Dashboard provides a sound way to customize the system to each user's needs.
- In the **execution mode** all programs which where configured in a start-script are started and controlled. Moreover, it provides a common user interface to access the system's functionality. The Dashboard is visible as a button-bar on the upper edge of the screen which is always in the foreground.


Figure 13: Dashboard in Execution Mode

The Dashboard application can be connected to a ROMAN server and therefore sends events to the overall system if the user switches between different views or presses function buttons. Furthermore, the Dashboard accepts messages from ROMAN concerning the window management.

## 4 PROGRAMMING BY DEMONSTRATION

As described before one of the major aspects of simulation models is dynamic. One of the major advantages when using the grasp detection technology is the capability to recognize dynamics the users perform with model objects. Interpreting this demonstration as programming-language enables the user to program system behaviour for simulations and plant control. To make the computer redo respectively to derive programs from what the user has previously demonstrated, is an old issue in human-computer interaction. This research is focussed on the conventional WIMP (Windows Icons, Menus and Pointers) interface styles [3]. The 3D interface provided by the *Real Reality* concept offers new opportunities for specifying dynamic model behaviourby demonstration. The investigation of this potential is one of our main goals of research. In this section we discuss an approach to this issue.

### 4.1 PROGRAMMING MATERIAL FLOW IN CONVEYOR SYSTEMS BY DEMONSTRATION

Currently, we are working on the application of the *Real Reality* concept for event based simulation systems for material flow control and plant layout. A typical scenario in this area of application is a conveyor system supplying several machines in a factory with raw materials or partially finished parts for further processing. It is the task of a control system to ensure an optimal utilization of the production capacities. Especially the order of machine utilization is an important factor for the productivity of such systems. In practice, this logistical problem is hard to solve, even for small plants. Therefore, simulation technology, particularly event-based simulators are a widely used tool.

In order to support this kind of simulation task a construction kit consisting of conveyors, machines and workpieces has been built (figure 5 and 9). Rectangular solids represent conveyors, the quadric solids are the machines, where specific operations are performed. The arrow on each element indicates the direction of the material flow. If a single element has two exits, which means that there are two possible directions available to continue, this constellation will be called a branch. On the other hand, if a single element disposes of two inputs, the material may flow together again. Such elements may cause conflicts if material is delivered from more than one direction at the same time.

At a branch a decision is necessary to determine which direction has to be followed. In our examples, blank workpieces are delivered and put in a circuitry where they are buffered and move around, until a series of machine operations is performed with them. Afterwards the workpieces leave the circuitry via a branch. This simple system allows the investigation of important aspects regarding the flow of workpieces through the plant. The main issue discussed here is the question how to derive rules for branching decisions from the input recorded with the *Real Reality* Modeler. Furthermore, these rules must be represented in a formal notation for the utilization in subsequent simulation experiments as well as for the transformation into control programs.


Figure 14: Demonstrating a Branching Decision depending on Object Properties

Already our first program version presented at the Hannover-Messe ´96 was able to generate rules depending on workpiece attributes coded by color. Of course, a more sophisticated control mechanism is needed. Figure 14 shows a situation in which the following specific rule has been demonstrated: *„move all light containers straight and branch dark ones out“*. This rule is extracted, transferred to the simulator and the participants can evaluate their program behaviour immediately in this system configuration.

In a different situation, the user may demand from the system to make a decision depending on the current state of the plant. Each resource (machine or conveyor) of the plant is either free or occupied. These two states determine whether a resource is available for processing or not. In the model this is indicated by placing workpieces, represented by colored cubes, on the resources. In a branching decision just a few resources are relevant. The context of a decision rule must be specified by placing tags on these resources (figure 15). This situation shows that the state of the two machines determines the decision of branching which is indicated by their tags (see the small discs). One of the machines is occupied whereas the other one is free for processing. The user moves the light-colored workpiece towards this machine. From this action the following rule can be derived: *„if machine A is occupied and machine B is free then route pallets to B“*. From now on, the simulator will apply this rule each time the specified condition occurs.

These activities of demonstration can be processed as a programming language. This allows the recognition of user intentions and the generation of formal specifications serving as programs for simulation or plant control. The major advantages compared to textual as well as to mainly graphical programming languages are:

- Easy to learn
- Context of the program and the plant model is kept permanently


Figure 15: A Branching Decision depending on the Plant’s State

- Immediate feedback through simulation is possible
- Simultaneous and cooperative work of participants is being featured

Machine understanding of demonstrated rules is the topic of the following passage.

## 4.2 A Stage Model of Demonstration Analysis

A model consisting of seven stages has been developed to model the process of plant layouts and material flow programming from the first conceptual meeting to the deliverable specification and control programs. From stage to stage an abstraction of the raw input data to formal programs takes place. Figure 16 illustrates and names the stages. Above the square process stages the characteristic input information provided by the system is shown. This information partially depends on the application context and is therefore named context knowledge. Below the process stages the feedback provided to the users is represented. This feedback either supports the user interactions at the model directly or offers relevant information about the actual model. This computer generated information helps to refine the model and optimize the planning results.

## 4.3 Comparing Real World Interactions with Language Processing

The tracker-based modeling generates data streams which can be processed analogous to natural speech. This gains to make use of advanced technologies for speech recognition.

The data streams of the glove and the tracking system are treated in seven successive processes (see figure 16) to define VRML97 scenarios for simulation and control programs. These steps are related to those in language processing. While glove and tracking system deliver vector streams directly, those have to be

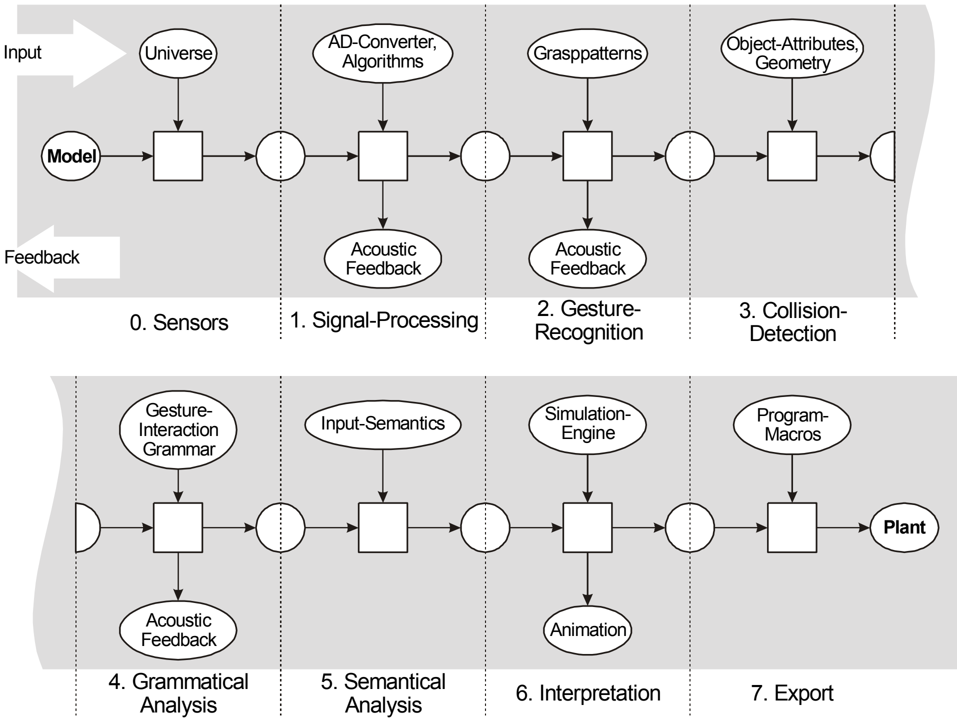



Figure 16: Processing User Inputs on a concrete Model in 7 Stages

extracted first from data streams recorded by microphones for language. It is difficult to describe voice patterns analytically. This is the reason why statistical methods which can be trained are more common [4]. Subsequent analysis of language makes use of knowledge which can be specified like lexical expressions kept in lists or the permitted grammar specified in some formal manner [5]. Depending on the application, extracting the semantical meaning out of the spoken language is a difficult task not solved satisfactorily yet. One of the main reasons for this problem is that knowledge about the relevant context is required and new information must be integrated into the database which means processing and learning in real-time. The system architecture may vary in different approaches to solve typical language recognition problems like

- individual characteristics of speakers and microphones,
- improper and fuzzy pronunciation and
- vague or indistinct meaning of the context and
- relevance of intonation.

Anyway, in the left column Tab. 1 tries to identify typical stages for language recognition.

We consider interpreting of taught behaviour as programming language for industrial plants to be a much easier task than this. Our reasons are as follows:

- Tracking system and data glove deliver feature vectors directly.
- We deal with a formal programming language defined by the target application.
- Relations between objects like collisions are easier to determine than between words and meanings.
- The recognition system doesn't have to learn.

Nevertheless, we may profit from developments made in language recognition. This idea led to the development of the seven stages in figure 16. The right column of table 1 compares gesture interaction processes directly with language recognition. We see the different input data types converge with ongoing processing, and they are equivalent from step 4. on. We found this paradigm very helpful and also stable in proceeding

| | Language processing | Gesture interaction |
|---|---|---|
| 0. | microphones | data glove and tracking sensor |
| 1. | signal processing & feature extraction | signal processing |
| 2. | phoneme detection | grasp recognition |
| 3. | lexical analysis | collision detection |
| 4. | grammatical analysis | interaction grammar |
| 5. | semantical analysis | semantical analysis |
| 6. | interpretation of information | interpretation of information |
| 7. | control of applications | control of applications |

Table 1: Process Steps of Language Processing compared with Gesture Interaction

development of our applications. Consequent idea is to enlarge the possibilities of interactions with real objects to a more expressive language for programming and communicating with computers and partners connected via the web.

# 5 DISTANCE COOPERATION ASPECTS

As already mentioned in chapter 2. (Applications) the modeling within a *Real Reality* environment is especially suited for working cooperatively with the physical model. Several users can stand around the modeling desk, and we only have to make sure that the sensors can recognize the actions of all participants and that everybody has a good view on what happens on the monitor.

Thinking one step further and imagining the use of *Real Reality* in applications for cooperative modeling with participants which are locally separated but connected via the web we will come to a new class of conceptual and technical questions – but also to a couple of new application fields. These fields will be discovered in coming research projects.

## 5.1 SYNCHRONIZATION OF PHYSICAL MODELS

Unlike the possibilities purely local *Real Reality* applications offer, the user of the modeling environment shares now his model with other, distant users now, i.e. remote participants cannot only view but actually manipulate it..

Remote users can either work with their own physical models or just with purely virtual models on normal computer workstations. For the other participants this should, however, make no difference, since modeling in the real and in the virtual world are, in principle, interchangeable forms of modeling.

A fundamental problem in the application of *Real Reality* and spatially separated, cooperative environments exists in so far as, that the system must show the user clearly, what has been changed at the model by other participants.

One possibility would be, that e.g. a robot automatically executes remote manipulations. The employment of a projection for the augmentation of the virtual model on the modeling desk through a beamer is certainly easier. A user of the physical model would then have to re-build the change of the remote user manually.

A further, completely different approach would be to employ multiple virtual models – one for each physical model. In the 3D view this could be shown as a corresponding number of virtual tables with working plates which are arranged next to each other. The user could navigate between the tables in the virtual world, look at what the others have build up and manipulate it within the virtual world. This technique would be very suitable for the support of the creation and discussion of alternative solutions.

We will further investigate in the question which of the two techniques fulfill the needs for cooperative *Real Reality* system best and think about dependencies on application fields.

## 5.2 VIRTUAL PRESENCE

In comparison with purely local *Real Reality* environments an important conceptual change occurs when modeling takes place cooperatively and spatially separate: the reference model moves from the physical world into the virtual world. The user of the physical model doesn't define anymore the model through his operations alone, since the virtual model can become newer than his physical one through the manipulation of the remote participants.

Therefore, the user must pay more attention to what happens in the virtual now, he must observe the other users' actions and make sure that his own actions will become visible to the others. Despite spatial separation he needs the feeling to be really located in a common, virtual modeling environment. Thus the user requires the perception of virtual presence, which procures him the sense of participation and involvement.

The feeling of presence appears as consequence of different sensory impressions. Important elements are, among others ([15]):

- Increase of the sensory information.
- Control over the relationship of sensors to their environment (through navigation, grasping, rotation of the virtual head for changing the viewpoint and the acoustic perception).
- Ability to manipulate the environment.

In the development of cooperative applications we can meet these points by extensively projecting the common virtual 3D-model onto the wall behind the modeling desk (s figure 5). A network connection to each remote participant is taking care of the immediate indication in the virtual model of all actions recognized by the sensors. With the help of an appropriate input device the user can navigate in the virtual model.

The own virtual presence and the one of the other participants should also contain a visual manifestation within the virtual world suitable for giving the users a good impression of the others' actions in realtime. Such virtual representatives are also called avatars. Usually, avatars are presented as human-like figures. When selecting the combination of sensors for cooperative *Real Reality* applications we must pay attention to the possibility of extrapolating meaningful avatar movements from the sensor's input.

At the edge of this passage we want to mention, that such movable 3D-figures are also very likely to provide functionality within virtual worlds. In this case the figures wouldn't be the representatives for human participants, but serve as metaphors for functions. They would be controlled by software modules and could play certain roles within the virtual world, e.g. offer access to help information. A concept for the behaviour definition of such automated characters is proposed in [7].

### 5.3 CONFERENCING

The users must be able to call the others' attention to their changes, must be able to explain, what they intended to do and must, on the other hand, have the possibility to ask the others, what they thought concerning their latest actions or what they plan to do next. Therefore, the system must offer mechanisms for getting into contact with each other. At this point the integration of conference- and chat-functionality is obvious (s. [8]).

Interacting with the participant's avatar would be an appropriate metaphor for using conferencing in virtual worlds. This could include selecting the avatar as an intuitive way to initiate a one-to-one talk and the mapping of the live video stream onto the avatar's head.

### 5.4 REMOTE POINTING AND SELECTING

Conversations regarding the model can also be supported with an appropriate visualization of actions like pointing and selecting by using the avatar figure. Pointing, for instance, could be indicated as movements of the corresponding avatars' arm and fingers; a selected object could be emphasized with a colored frame (s. [8]). In the physical model the marking of components could also be realized by employing a projector or a beam.

## 6 CONCLUSION

We agree with attempts to make engineering more visual in order to allow people to understand the process of technical planing. A rapidly growing number of projects with related topics show the evidence of that [9].

Using graspable models goes beyond that. People are in a position to easily interact in the model world and to participate in the planing process [10] [11] [12]. Bringing in their ideas leads to more planing quality and acceptance in the results [13].

By the way, people are learning very well in operating with objects because haptics are a major episode for learning to understand the environment [14]. The major drawback for making use of Virtual Reality lays in the modeling process. Designing virtual models is an expert domain and very cost intensive. This is currently leading to a development of giving away this job to polish or indian programmers occasionally (Panel "VR-The Future Engineering Workplace: A European Perspective" on Virtual Environments ´98 Stuttgart). Also for simulation studies, having an evident positive effect on planing processes, a major problem area is seen in a too pure support of the modeling process [15]. While the separate profit of concrete or virtual models is quite well proved we can assume an even better benefit from coupled models.

To go beyond theory and expectations we needed a prototype to experiment with coupled real and virtual worlds. This has been realized with the prototype of the *Real Reality* environment described in this article. During the design process it turned out that gesture based interaction requires complex software structures. We solved this problem by structuring this task into the 7 stages described above. Another major problem area is the required hardware. Data gloves cost either as much as a small car (Cyber Glove) or are not comfortable to wear and deliver inaccurate values (Nintendo Power Glove, 5$^{th}$ Dimension). In consequence, we are working on a pressure sensor based interaction device (see figure 17) [16].

The principle of magnetic tracking (Polhemus) will do a satisfying job if we accept modeling with non metallic objects. A new Tracking System based on opto-acoustic/inertial principles (Intersense) looks promising to solve this problem.

In practical use we have experienced a good handiness and acceptance of the system especially with non-expert groups modeling technical situations. We intend to certify this observation in an evaluation in vocational schools which is part of the project BREVIE.

An augmentation of the model provided by a video beamer projecting computer generated information directly onto the model, couples information from real and virtual model in an ideal way.

Motivated by the results of the past, we will concentrate our future work at artec on improving the *Real Reality* interaction environment and keep you informed.

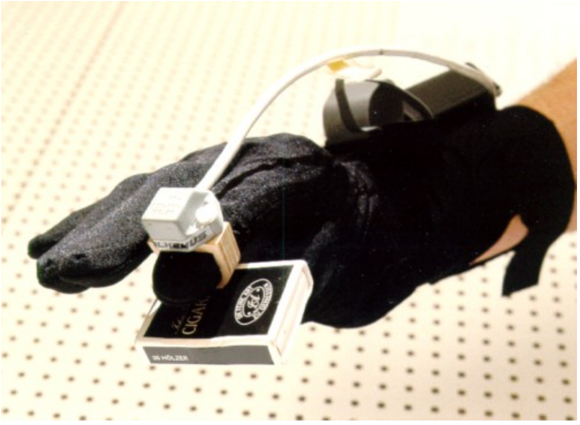



Figure 17: Grasp Tracker

# 7 References


[1] F. W. Bruns. Zur Rückgewinnung von Sinnlichkeit - Eine neue Form des Umgangs mit Rechnern. In Technische Rundschau Heft 29/30, pp. 14-18, 1993.

[2] V. Brauer. Feature-basierte Erkennung dynamischer Gesten mit einem Datenhandschuh. Diplomarbeit, Universität Bremen, 1994.

[3] E. Cypher (Ed.). Watch What I Do - Programming by Demonstration. MIT Press, Cambridge, Massachusetts, 1994.

[4] R. Rabiner, A. Lawrence. Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. In: Proceedings of the IEEE, Vol. 77, No. 2, February 1989.

[5] H. Niemann, G. Sagerer, F. Kummert, M. Mast. Repräsentation und Nutzung von Wissen in einem Spracherkennungs- und Dialogsystem. In: Künstliche Intelligenz: Themenheft Mustererkennung, Band 3/95, pp. 35-41, 1995.

[6] H. Ernst. Automatisierte Charaktere in Virtuellen Welten. Diplomarbeit, Universität Bremen, 1997.

[7] W. Barfield, D. Zeltzer, T. Sheridan, A. M. Slater. Presence and Performance Within Virtual Worlds. In: W. Barfield, T. A. Furness. Virtual Environments and Advanced Interface Design. Oxford University Press, New York/Oxford, 1995, pp. 473-513.

[8] K. Schmudlach. Computerunterstütztes Konferieren in einer gegenständlichen Modellierumgebung. Diplomarbeit, Universität Bremen, 1998.

[9] Industrie Management – Visual Engineering 1/97, GITO-Verlag, 1997.

[10] H. Ishii, B. Ullmer. Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms. In: Proceedings of CHI 1997.

[11] M. Rauterberg, T.Mauch, R. Stebler. What is a Promising Candidate for the Next Generation Interface Technology? In: Proc. of the 5th Conf. on Interfaces To Real & Virtual Worlds, Montpellier, 1996.

[12] E. Arias, H. Eden, G. Fischer. Enhancing Communication, Facilitating Shared Understanding, and Creating Better Artifacts by Integrating Physical and Computational Media for Design. Proceedings of the DIS (Designing Interactive Systems), Amsterdam, 1997, pp. 1 ff.

[13] Scheel, Hacker, Henning. Fabrikorganisation neu beGreifen. TÜV Rheinland, Köln, 1994, pp. 155 ff.

[14] D. Müller. Simulation und Erfahrung. Ein Beitrag zur Konzeption und Gestaltung rechnergestützter Simulatoren für die technische Bildung. Dissertation, Universität Bremen, 1998, pp. 97 ff.

[15] G. Reinhart, K. Feldmann. Simulation – Schlüsseltechnololgie der Zukunft? Stand und Perspektiven. Herbert Utz Verlag Wissenschaft, München, 1997, pp 35-37.

[16] K. Schäfer, W. Bruns. artec-paper 67. Rechnergestützte Übergänge zwischen gegenständlichen und abstrakten Modellen produktionstechnischer Systeme. Dritter Zwischenbericht zum DFG Forschungsprojekt RUGAMS. artec, Universität Bremen, 1999.


**More Information under:**
http://www.artec.uni-bremen.de/field1/rugams/
http://www.brevie.uni-bremen.de/


Contact:
Hauke Ernst | Kai Schäfer
{ernst | schaefer} @artec.uni-bremen.de
Phon

e: ++49 / 421 / 218- {9089 | 4833}

# Virtual Reality Interface for the World Wide Web

Denis Gracanin, Kecia E. Wright
University of Southwestern Louisiana
P.O. Box 44821
Lafayette, LA 70504-4821, USA
{dg|kew}@acim.usl.edu

## ABSTRACT

The World Wide Web (WWW) has become a standard in human communications and information retrieval. Since its development in 1989, it has matured from linking two-dimensional text-based information sources to encompassing a range of multimedia data representations. More recently, the WWW has also come to incorporate interactive virtual environments with the development of the Virtual Reality Modeling Language (VRML). A three-dimensional virtual reality (VR) based interface can be applied to the WWW to better represent complex relations and interdependencies among various pieces of information. The challenge is twofold. First, it is necessary to represent the WWW in a VR space in a standardized manner. Second, an intuitive user interface is crucial in such an environment. The representation of the WWW requires a framework for the HTML to VRML conversion. At this point, only a limited subset of the HTML tags is supported. Since there are many ways to map a two-dimensional document to a three-dimensional virtual environment, a set of rules has been created to standardize this procedure. The user interface provides a minimal functionality set needed for browsing. The use of specialized hardware (glove, tracking) provides for better interaction and manipulation. In addition, a three-dimensional hierarchical menu system is used to select information and to control the environment. The ongoing work is concentrated on the completion of the HTML to VRML conversion and the improvement of the user interface. In addition, new developments regarding the X3D (formerly VRML-NG) and the Extensible Markup Language (XML) are considered.

**Keywords:** Virtual reality, World Wide Web, user interface, Virtual Reality Modeling Language, Java

## 1 INTRODUCTION

As computers have become better and more efficient with each new generation, so have their user interfaces. Although each new paradigm provided the user with easier, more intuitive access to data, the limitations of each interface eventually became apparent. Table 1 shows the evolution of user-interfaces with each generation of computers and the barriers it created for the user.

The current interface referred to as "WIMP" (Windows, Menus, and Pointer) has made it possible for a large segment of the population, without extensive knowledge of the inner workings of a computer, to use personal computers. However as the capabilities of processors and input/output devices grow along with the increasing amount of information sources, the limitations of the WIMP interface becomes obvious [11].

| Generation | Description | Limitation |
|---|---|---|
| First | Plugboards, dedicated setup | Front panel |
| Second | Punched card batch | Countertop |
| Third | Teletype timesharing | Terminal |
| Fourth | Menu Systems | Menu hierarchy |
| Fifth | Graphical controls, windows | Screen |
| Sixth | Cyberspace | ??? |

Table 1: User-Interface generations

More and more applications are incorporating 3D graphical elements that require better ways of interaction. For instance, designing 3D objects, such as in CAD programs, can be a difficult task using two-dimensional approaches. Furthermore, the physical limitations of the monitor separate the user from the information creating an unnatural approach to information access.

Virtual reality (VR) seems to be the next logical step in the evolution of human computer interaction [14]. Immersing the user in the information space allows for more natural semantic interactions between the user and information. In addition, the presentation of information in a spatial environment can help users better understand its context and content [1].

One application that has clearly outgrown the limitations of the current desktop user interface is browsing the World Wide Web (WWW) and viewing Hypertext Markup Language (HTML) documents. Although, the traditional 2D WWW browser allows users to access a variety of information sources over the Internet, the 2D interface limits the user from forming a clear understanding of the interdependencies of these sources. A three-dimensional VR based interface can be applied to the WWW to better represent complex relations and interdependencies among various pieces of information.

The first issue to address when integrating the WWW components into a virtual environment (VE) space is how to represent the various features of the WWW in a spatial environment. The representation must take advantage of the added dimension of the 3D space, allowing the user to not only easily visualize the information but also how they are associated to each other. Representing an HTML document within a VRML environment offers many challenges, since the document may contain many embedded sources.

The second issue to address is providing an intuitive and natural user interface. The first step is to define a set of functions/commands for browsing the WWW and then utilize available input/output devices to implement them. A sensor embedded glove and a tracking system are considered.

This paper describes a virtual reality based WWW browser implemented using the Virtual Reality Modeling Language (VRML) and Java technology. In addition to providing the essential tools for navigating the WWW, the browser presents HTML elements in a three-dimensional environment.

In the following section an examination of previous work done in the area of information visualization as well as 3D WWW browser developed in the past is presented. Section three provides a description of the browser architecture and implementation techniques used. Section four presents a small example. Section five discusses future research directions and concludes the paper.

# 2 OVERVIEW

Information and document visualization techniques are used to display structured documents (e.g. a directory hierarchy) and non-structured documents (e.g. databases). In general, these systems are used as a 3D alternative to document retrieval. However there are some document retrieval systems that also display a document's contents in the 3D space. Some of these even spill over to the category of 3D WWW browsers, allowing the user to view the hyperlink structure of retrieved WWW pages and displaying a selected WWW page. In almost all cases these 3D WWW browsers use a 2D rendering of the WWW page. It is important then to examine the past works of the various types of information visualization systems and techniques researched in the past.

## 2.1 INFORMATION VISUALIZATION

There are several techniques for visualizing large document collections as well as several systems. One group of systems is based on the Populated Information Terrain (PITs). The PIT is basically a multi-user virtual reality based data space that allows users to become immersed within the data and to explore and interact with it while making a full use of their perceptual and intuitive skills. Examples of systems based on the PIT paradigm are VR-VIBE [3], and CyberWorld [10]. These systems present documents and users as graphical objects in the space. Documents are grouped based on relevance to a query performed by the user. The query is also presented as a 3D object in a data space. User's can then view/interact with the information and see who else is interest in the same information.

Other visualization techniques are used for document structures such as directory hierarchy or hyperlinked documents. The Hyperspace system [15] used the NCSA Mosaic browser for fetching WWW pages then displaying the linked documents as a 3D graph metaphor with spherical nodes representing the documents and directed edges as the links. The system used a self-organizing technique for laying out the graph. The technique placed new nodes randomly on the graph. This system is simple and straightforward however it used another window for displaying page content. Another disadvantage was the self-organizing technique for laying out the graph. It placed new nodes randomly on the graph, which could change the structure of the graph entirely, therefore disorienting the user.

A more recent addition to this class of systems is Bittco Solution's NetReality™ (formally Vrtuoso, http://www.bittco.com/Netreality/netreality.html). This system uses artificial intelligence (AI) to analyze documents and then arrange them in the 3D space according to their meaning and content. Documents are represented as spheres and strategically placed in a 3D landscape. Both the document and landscape contain meaning about the documents. This system is mostly concerned with making order out of WWW information obtained through searches or possibly bookmarks. It does not make an effort of displaying page content and does not claim to be a WWW browser.

## 2.2 3D Information Space

The information visualizer [5, 12] is a user-interfacing system designed for integrating information access from different levels of storage. This system attempts to overcome the limitations of the 2D screen by using the "rooms" metaphor. The user can move about the rooms, examine objects and move to other rooms to find more information. It employs several information visualization techniques such as the Cone Tree for hierarchical organization, the perspective wall for linear organization, etc.

Another type of information space recently at the Georgia Institute of Technology is the Virtual Venue [4]. It provides 3D representation of various embedded media data types and hyperlinking capabilities to those data types. Users move around the space in a restricted navigational mode and obtain information about the particular objects in the environment. Text, images, audio, and animation are used to enhance the 3D objects in the environment.

## 2.3 3D WWW Browsers

One of the first applications to provide a WWW browsing interface within a virtual environment is VRMosaic [2]. VRMosaic, by Ian G. Angus and Henry A. Sowizral in 1996, used a modified version of the GUI toolkit, InterViews 3.1, to port flat-screen applications into a virtual reality system called RealEyes. The modified GUI toolkit renders NCSA Mosaic's interfacing window as 3D geometry and attach that geometry to a virtual clipboard. Because the 2D interface is rendered in the virtual environment, WWW browsing in the virtual environment is virtually the same as in the original 2D application. However, inline images were presented in front of the user viewport instead of being embedded in the document. Also, certain documents or data could be linked to the environment and objects in the environment. Objects in the VE were selectable and linked to associated WWW pages.

WWW3D [13] is another virtual reality WWW browser initially implemented with the DIVE virtual reality system in 1996 at the University of Nottingham. The browser provides a display of the WWW documents and the structure of the WWW pages the user has browsed in the recent past in a single three-dimensional window. Like the NetReality system mentioned above, documents are represented as spheres. However this system also provides WWW document representation inside the sphere. The user must enter the sphere to view the document. The primary advantage of WWW3D is that it also supports multiple simultaneous users browsing different areas of the WWW. The downside to this system is probably the inability to view more than one WWW document concurrently.

One of the main advantages of VR as an information space is the unlimited space for displaying objects. There are several situations when users need or want to view more than one WWW page at the same time. Although the system takes advantage of the spatial environment for displaying the history information, it does not do the same for presenting WWW pages. The system provides only a 2D representation of the WWW document, which doesn't take advantage of the spatial environment.

The WebBook and Web Forager [6] systems developed at the Xerox Palo Alto Research Center falls more under the class of 3D workspaces, rather than 3D WWW browsers. However workspace described was specifically designed for visualizing and storing HTML pages. Based on the information visualizer mentioned earlier these systems offer a 3D hierarchical information storage system and workspace for information access and viewing.

Using a "book" metaphor, the WebBook allows users to group together preloaded related WWW pages and to manipulate these pages as a unit. Users can link to other pages in the book or pages located outside the book. The book can also be "exploded" so that all pages are available simultaneously.

The Web Forager is the 3D workspace that provides different levels of storage and interaction with individual WWW pages and WebBooks. Books and pages can be placed on a virtual desktop in front of the user's view, floating in the space just in front of the user, or on a bookshelf.Pages.

# 3 IMPLEMENTATION

In general, WWW browsers are responsible not only for navigation in the WWW but also the interpretation and display of HTML documents, the primary source of information on the WWW. Hypertext documents are created with HTML tags that indicate text formatting, such as font size, as well as hypertext links. The WWW browser reads the tags and then interprets them to display the document. Tags are also used to incorporate multimedia files such as sound and video. These files are included in an HTML page however the browser must use helper applications or plug-ins to display the file. A fairly recent addition to WWW accessible documents is the Virtual Reality Modeling Language (VRML). Using a plug-in, a traditional WWW browser can display a VRML file or one that is embedded in an HTML document. The VRML file format is the basis of the implemented virtual environment and is used to display web documents.

The browser interprets and displays WWW information using its VR engine. Basically the engine fetches WWW documents form the WWW and translates the information into 3D objects placed in a virtual environment. It is comprised of a document retriever, an HTML parser, lookup table and a virtual environment modeler (see Figure 1).

The document retriever loads a WWW document specified by the user into the VR engine. Before sending the document to the HTML parser it must first resolve the document type (HTML or VRML). A VRML file can be included directly into the display and therefore is sent directly to the scene modeler.

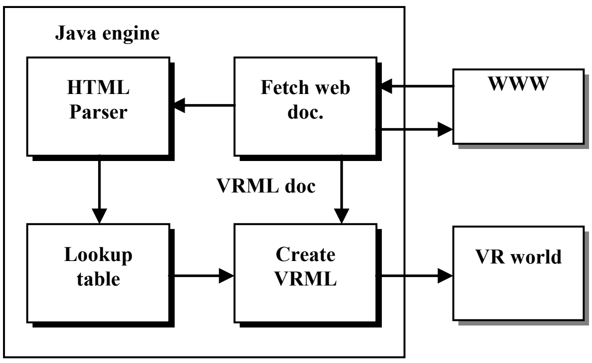



Figure 1: VR interface architecture

If the document type is HTML it is sent to the parser where it is tested for correctness and disassembled into nodes identified by a tag. In addition to storing the tag type, all the information contained in the tag is also stored in the node. For example an image tag will typically contain the URL source, along with the image width and height. The parser also determines which tags the browser currently supports. HTML is an evolving language with new tags and attributes being added with each upgrade. Therefore it is not possible to support all features offered by HTML.

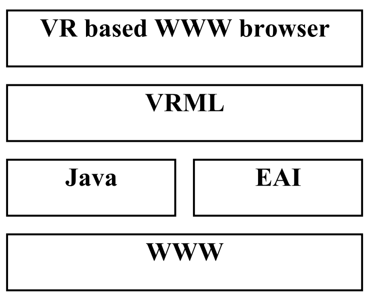
The supported tags are then translated into 3D objects via the lookup table. The table contains keys for each tag and corresponding references to 3D prototypes used as generic models for tags. The prototypes are customizable using the stored attributes for the tag. Currently, each supported tag has only one prototype, however it maybe desirable in the future to offer more than one generic model for a particular tag. The user can then set preferences for rendering a tag using a certain model.

The object modeler creates a 3D object of the tag using the prototype found in the lookup table and the tag's attributes. As each object is created it is positioned in the environment in front of the user's viewpoint. Once the document is loaded, the user can perform traditional VR transformations such as navigation, selection and manipulation of objects.

## 3.1 IMPLEMENTATION TECHNIQUES

The implementation is based on the Java programming language, the Virtual Reality Modeling Language (VRML) and the external authoring interface (EAI) (see Figure 2). Together, these systems perform fetching, parsing, translating and 3D representation of a WWW document. The VRML scene is rendered in a VRML viewer (CosmoPlayer) and accessed by a Java applet. The system uses the External Authoring Interface (EAI) to communicate between the applet and the VRML browser.

Figure 2: Overview of implementation techniques



The VRML is employed for the representation of HTML components as 3D objects. VRML as defined by the Web3D Consortium is "an open standard for 3D multimedia and shared virtual worlds on the Internet" (http://www.vrml.org/about). Like HTML, VRML is not a language, but a scene description language that describes the geometry and behavior of a 3D scene. A VRML file is essentially a collection of nodes, arranged in a particular hierarchy [7]. The node often corresponds to a 3D model that has properties such as shape, color, and position in 3D space. A 3D object is also a combination of nodes arranged in a particular hierarchy. VRML is capable of representing either static or animated objects. Each object can have hyperlinks to other media such as movies, sound and images.

Before HTML documents can be rendered as 3D objects in the scene, they must first be parsed and translated into VRML nodes. The Java programming language is responsible for fetching, parsing and translating the document into nodes. It contains components that provide classes for reading and parsing formatted documents such as an HTML document. Placing the nodes into the scene is done by the EAI [7]. The EAI is a collection of methods that allow an external application, such as a Java applet, to communicate with a VRML scene. This communication includes creating, adding, subtracting and manipulating objects in the VRML based environment.

## 3.2 HTML TO VRML CONVERSION

The Swing component of Java [8] is a new graphical user-interface (GUI) toolkit that simplifies and streamlines the development of visual components that are used in graphically based programs. Two features contained in the Swing component are used as the parsing tool in the 3D browser. The Element interface has the ability to describe various structural parts of formatted documents. Document structures such as a paragraph, a line of text or tag items in an HTML document are defined in an Element. All Elements from a document linked together to form an Element tree. Every Element is associated with an AttributeSet that store values such as the Element's font size, text color, URL source depending of what the Element represents. The system uses Swing's HTMLEditorKit to read an HTML document place it in the Element interface for further analysis.

Table 2 shows the HTML components and the corresponding VRML nodes that make up the lookup table. Currently only basic formatted documents are completely translated into VRML objects. While some HTML components can be directly translated into VRML nodes (audio), others may need multiple nodes for a minimum representation. For example, a paragraph of formatted text will need several text nodes for recreation in the VRML space. The VRML objects are stored as external prototypes in a hash table. Once a document is separated into Elements, those elements are then queried for tag names and attributes and converted to keys. These keys are used to obtain the proper prototype in the table. The prototypes are customized to display the HTML component by applying the attributes to the prototype's fields. The objects are then ready for placement in the VRML scene.

| HTML | VRML |
|---|---|
| Text | Text and FontStyle |
| Lists | Text and Group |
| Tables | Text and Shape |
| Links | Anchor |
| Image | ImageTexture and Shape |
| Video | MovieTexture and Shape |
| Audio | AudioClip |

Table 2: HTML to VRML conversion

## THE VR WORLD

As each element translated from the HTML document is defined as a VRML object, it is added to the user's browser scene using EAI methods. The EAI is one of two ways of accessing and controlling VRML nodes in a VRML scene. The alternative method is through the Script Authoring Interface, which allows access and control of nodes from within the VRML file. EAI functions, on the other hand, are placed in some external program. Both methods are just about equal in what they can do, however, the Scripting Interface is like hard-coding certain events in the VRML scene. For this application it is necessary to have the flexibility of the EAI. The application does not know ahead of time what WWW document will be loaded.

Before adding a node, the browser must get instances of the VRML browser and the scene root node. The Browser class provides methods for obtain an instance of the browser and then getting a pointer to a node. Nodes named using the DEF construct are accessible to the applet. Once the node pointer is acquired, its eventIns and eventOuts can be accessed accordingly.

The Browser class also provides methods for creating a node from a string or URL. It is important to note that nodes created this way are inaccessible, so any type of interaction or animation must be done from within the VRML file.

An example can be shown for a simple HTML file shown in Figure 3. In a 2D browser it is rendered as a line of plain text, bold text and an image with height and width dimensions. The VRML prototype for the image representations is listed in Figure 4.

```
<html>
<body>
This is an example of plain text.
<br><b>This is an example of bold text.</b>
<p><img
SRC="http://www.acim.usl.edu/~kew/HTMLtoVRML/U
sllogo.jpg height=102 width=386>
</body>
</html>
```

Figure 3: Simple HTML file

```
PROTO ImageBox [
 exposedField MFString image NULL
 exposedField SFVec3f translation 0 0 0
 exposedField SFRotation rotation 1 0 0 0
 exposedField SFVec3f scale 1 1 1
 field SFVec3f size 1 1 1
]
{
DEF ImageBox Transform {
 translation IS translation
 rotation IS rotation
scale IS scale
 children Shape {
 appearance Appearance {
 material Material { }
 texture ImageTexture {url IS image}
 }
 geometry Box {size IS size}
 }
 }
}
```

Figure 4: Image VRML prototype

As the file is read in, each tag is put into an Element tree along with its attributes. The actual text is stored in leafElements. Parsing the tree the supported elements can be retrieved and converted to VRML code. In this case only simple text and image tags are supported. Using prototypes, the images and text can be altered to maintain the attributes in the html code. In this example the first line of text is plain with no attributes, therefore the default text prototype is used with the *string* field set to the text retrieved from the leafElement. The second line of text, however, has a bold attribute. To represent this type of formatted text the *style* field is set to “BOLD” along with the *string* field.

The image VRML prototype contains a simple box with texture mapping. Again the Element tree is parsed and queried for an “IMG” Element. If one is found, the “SRC” attribute (image URL) is retrieved and applied to the *image* field in the prototype. The width and height attributes are also used to resize the box so that the image ratio dimensions are maintained. Figure 5 shows the VRML representation of the HTML document. Each line of text and the image box are separate objects and inserted as children under the root scene node.

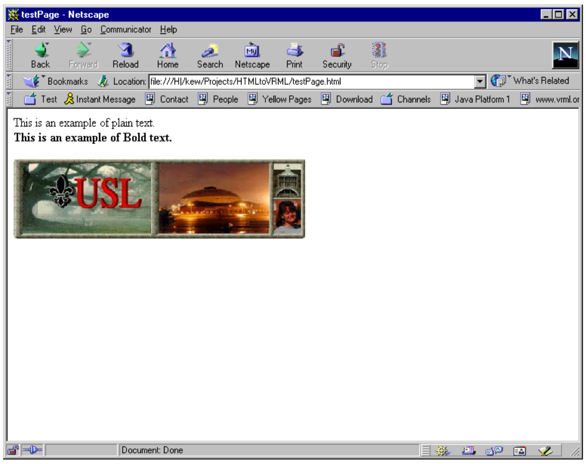



Figure 5: VRML representation

## 3.3 NAVIGATION

The significant difficulties of 3D input and display have led research in virtual worlds to concentrate far more on the development of new devices and device-handling techniques than on higher-level techniques for 3D interaction. One of the reasons is that there is no standardization of interaction techniques. Toolkits are developed from the scratch and are application specific. Furthermore, interaction techniques are dependent upon hardware devices used.

For this application, hardware devices put to use are a glove, a tracking system, and an optional head mounted display. VRML provides some standardization for interactions in a virtual environment, in that it defines a set of sensors. Classes of input sensors in VRML include pointing-device, environment, and drag sensors.

Pointing-device sensors (Anchor, CylinderSensor, PlaneSensor, SphereSensor, and TouchSensor) are activated when the user locates the pointing device over related geometry.

Environment sensors (Collision, ProximitySensor, TimeSensor, and VisibilitySensor ) detect changes in the environment.

Drag sensors (CylinderSensor, PlaneSensor, and SphereSensor) generate events when a user moves a pointing device while holding down the device’s button. The input is mapped to a specific geometry.

This type of sensors was developed due to the limitations of 2D pointing devices.

Based on the available equipment and VRML sensors, two interaction techniques were implemented, three-dimensional widgets (menus) and a simple gesture recognition. Figure 6 shows an example of implemented interaction techniques.

Basic functionality includes:

- History: an index of recently viewed WWW pages. Back and forward movement is achieved by selecting left and right "pointed" widgets in the upper part of the scene.
- Home: moving to a predefined starting WWW page by selecting a "home" widget in the upper part of the scene.
- Location: Text can be typed directly using a virtual keyboard.

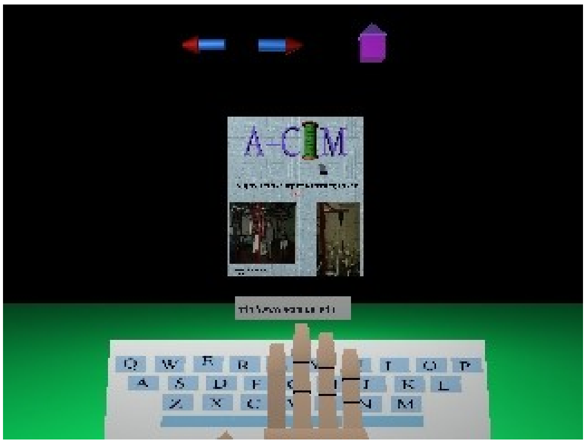


Figure 6: User interface components

The advantage of using three-dimensional widgets is an intuitive interface, a 3D reflection of a 2D browser. Widgets are selected and activated by a virtual pointer. The obvious disadvantage is the added clutter to the scene and making it more difficult to view other objects. That is somehow remedied by providing an option for hiding widgets and objects.

The virtual pointer is based on the input glove device (Figure 7). Hand postures reported by the glove are presented as a series of values related to the bend of joints in a hand. Based on these values, rotation values are calculated for the segments of the virtual hand model. Proper nesting and grouping of Transform nodes ensures proper alignment of virtual hand segments.

The Java Communications API (javax.comm) is used to implement the interface with the glove using a serial port. In addition, two classes, *SerialComm* and *Glove* have been developed to simplify interaction

Figure 7: CyberGlove™ device

with the VRML. Glove's button is used to show or hide widgets.

The similar approach is used to interface with a tracking system (Polhemus Fastrak™). The tracking system provides information regarding the glove position so that virtual pointer (a tip of the index finger) can be fully controlled.

When a head mounted display is used, it is cumbersome for user to type on a physical keyboard. A virtual keyboard provides an alternative way to type when textual information, like a URL, is needed. Only letters and space are provided at this point, fully functional keyboard is under development. The virtual keyboard can be hidden or shown the same way as widgets.

## 4 EXAMPLE

The "average" HTML documents include text, images and links. Text is arguably the most difficult HTML feature to represent since VRML is not normally used for displaying text documents. In addition to text, HTML also provides elements for visual formatting of characters and paragraphs, which also must be represented in the VRML scene in order to bring some organization to the text. One of the problems with VRML text display is that it does not provide support for all of the options provided by HTML, such as the various font families that are available. However some of the most common elements can easily be implemented directly into the FontStyle node or by using a combination of a parent Transform node and FontStyle node. The most used character formatting techniques used by HTML authors include:

- Font family (Times New Roman, Arial, etc.)

- Font size
- Font color
- Font style, (bold and italics)

The most common paragraph formatting techniques include:

- Justification (center, right, left)
- Block Quote or Indention
- Line breaks
- Headers (H1, H2, H3, H4, H5, H6)

In some cases of the paragraph formatting techniques such as the headers and address elements are rendered in a typical browser as a combination of other basic visual components. For example the different headers are rendered as a new paragraph with the text bold with varying font size depending on the header indicated.

## 4.1 DISPLAYING TEXT IN VRML.

The first step in implementing text is the creation of VRML prototypes that can be used for displaying various character styles and paragraph formatting. The basic hierarchy of the text prototype along with the field definitions is

- Transform (proto definitions: translation)
  - Appearance (proto definitions: diffusecolor)
  - Text (proto definitions: string)
    - FontStyle (proto definitions: style, justify, size, spacing, family)
  - Anchor (proto definitions: url)
    - Text (proto definitions: same as previous Text node)
      - FontStyle (proto definitions: same as previous FontStyle node)

The PROTO defines the translation field in the Transform node, which is used for the indentation, justification and line spacing needed. Although the FontStyle node provides a field for justification and supports values for left, right and center, it implements these values by positioning the text on the +x side of the y axis, the –x side of the y axis or centered across the y axis respectively. Therefore, it is necessary to use the justification field along with the translation field in the Transform node to achieve the effect intended by the author. A color field is defined for rendering not only the link colors but also text colors defined by the HTML authors.

The Anchor node provides support for text links that are indicated in the HTML page. The Anchor is a grouping node that fetches the specified file over the network when the user chooses any of its children. For this reason, a separate Text node is defined as a child of the Anchor node to provide support for text links.

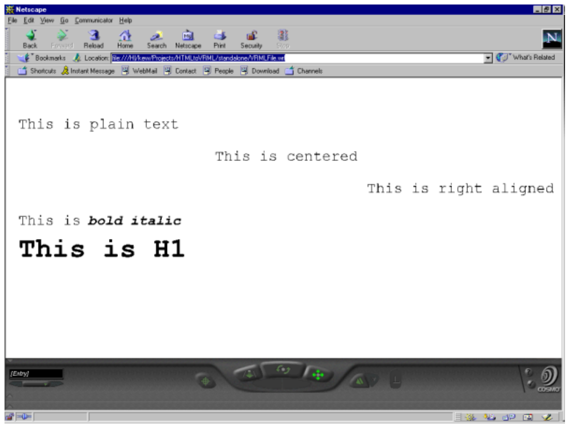



Figure 8: Text in VRML

A simple HTML page with some of the formatting described above is shown in Figure 8. The page is parsed and stored into the HTMLDocument structure provided by Java that indicate element tags and attributes found in the HTML page. The task of building the element structure is handled in part by the HTMLDocument.HTMLReader object. As the HTML tags are read from the source, the parser passes instances of the class, HTML.Tag to the reader, identifying the type of element being interpreted. HTML.Tag is simply a typesafe enumeration used to describe all of the legal HTML element types. Each of the HTML.Tag constants can also be used an attribute key that is capable of holding additional attributes associated with the tag. These attributes are keyed using an instance of the class HTML.Attribute. The HTMLDocument structure for the example HTML page is in Figure 9.

Once the Element structure is formed it can be traversed using the ElementIterator class. During the traversal each element is read and translated into the appropriate VRML component(s) in the text prototype. For example, the H1 tag is typical rendered as three points above the basefont size with bold font style text. Therefore an instance of the VRML text prototype is created with the FontStyle size field set to three points above the basefont and the style field set to “BOLD ITALIC”. The string field in the Text node is also set to the text found in the HTML.Tag.CONTENT element. This procedure is used to create a unique instance for each leaf element found in the tree.

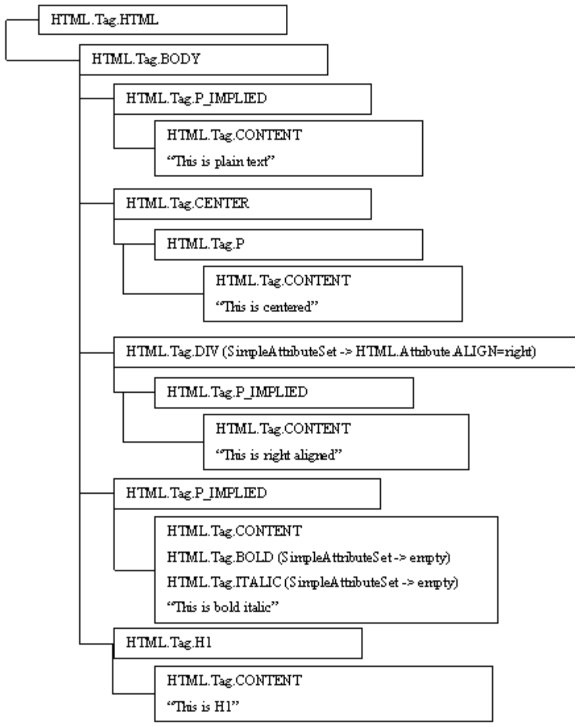



Figure 9: HTML decomposition

Although the results of converting the sample HTML page to VRML is not the identical rendition of the 2D browser, the intended structure is maintained along with the character style. Because the Text node is basically a geometric object, it becomes unreadable the further it is from the user. For this reason, the maximum line length is kept fairly short. Another difference is that currently variable widths are not supported in the VRML text prototype.

## 5 CONCLUSION

This paper presents techniques for a virtual reality browsing interface for the WWW. Using a combination of VRML, Java, and EAI programming standards and technology, WWW documents can be displayed as 3D objects in a virtual environment. The combination of these technologies has made it possible to create a virtual environment, provide 3D interfacing techniques for browsing the WWW and display a 3D representation of WWW documents. This system provides WWW document content with spatial properties and presents them in a virtual environment. The user is able to view and manipulate documents to enhance the presentation.

The presented system and the virtual reality interface for the World Wide Web is in a development stage and there are several open issues. One challenge is the problem of converting components so that the spatial properties are able to enhance the document's information. Without doing so the document simply reverts back to a 2D object in a 3D space. Another challenge is to handle components such as embedded applets. Applets are inherently 2D objects and usually require mouse interactions.

Future research will also include the Extensible Markup Language (XML) [9]. The XML, a simple dialect of the Standard Generalized Markup Language (SGML) will enable generic SGML to be served, received, and processed on the WWW, like HTML is now. XML has been designed for interoperability with both SGML and HTML. In addition, the next generation of VRML, the X3D (formerly VRML-NG) is based on the XML tagset. It is a lightweight 3D media file format highly interoperable with 2D WWW standards. Therefore, once specifications are completed, the integration of HTML and VRML documents through the XML framework will become feasible. A general approach for visualizing XML documents will provide a significant help for future virtual reality based interfaces for the WWW.

### Acknowledgments

This work has been partially supported by

- The Louisiana Board of Regents Support Fund, grant 98-00-ENH-TR-92: "A Proposal to Develop Virtual Reality Based Experiential Learning for Nursing Students".
- Department of Energy, subcontract 49X_SU016C: "The Demand Apparel Manufacturing Architecture (DAMA) University Research Center Project".

## REFERENCES

[1] P. Anders, Envisioning Cyberspace: Designing 3D Electronic Spaces, McGraw-Hill, 1999.

[2] I.G. Angus and H. A. Sowizral, "VRMosaic: Web Access from within a Virtual Environment", *IEEE Computer Graphics and Applications*, Vol. 16, pp 6-10, 1996.

[3] S. Benford, D. Snowdon, C. Greenhalgh, R. Ingram, I. Knox and C. Brown, "VR-VIBE: A Virtual Environment for Co-operative Information Retrieval", in *Proceedings of the Eurographics '95*, 1995.

[4] D. A. Bowman, L. F. Hodges, and J. Bolter, "The Virtual Venue: User-Computer Interaction in Information-Rich Virtual Environments", *Presence*, Vol. 7, No. 5, 1998.

[5] S. K. Card, G.G. Robertson, and J.D. Mackinlay, "The information visualizer, an information workspace", CHI '91, Human factors in *Proceedings of the Computing systems conference*, New Orleans, April, 1991.

[6] S. K. Card, G. G. Robertson and W. York, "The WebBook and the Web Forager: an information workspace for the World-Wide Web", in *Proceedings of the Conference on human factors in computing systems*, pp111-ff, 1996.

[7] R. Carey and G. Bell, The Annotated VRML 2.0 Reference Manual, Addison-Wesley, Reading, Massachusetts, 1997.

[8] R. Eckstein, M. Loy and D. Wood, Java Swing, O' Reilly, Sebastopol, California, 1998.

[9] E. R. Harold, XML: Extensible Markup Language, IDG Books, Foster City, CA, 1998.

[10] M. Hemmje, C. Kunkel and A. Willet, "CyberWorld – A Visualization User Interface Supporting Fulltext Retrieval", in *Proceedings of the Conference on human factors in computing systems (CHI '95)*, pp. 417-418, 1995.

[11] K. Pimental and K. Teixeira, Virtual Reality: Through the New Looking Glass, McGraw-Hill, Inc., 1995.

[12] G. G. Robertson, S.K. Card, and J.D. Mackinlay, "Information Visualization Using 3D Interactive Animation", *Communications of the ACM*, Vol. 36, 1993.

[13] D. Snowdon, S. Benford, C. Greenhalgh, R. Ingram, C. Brown, L. Fahlen and M. Stenius, "A 3D Collaborative Virtual Environment for Web Browsing", in *Proceedings of the Virtual Reality Universe '97*, 1997.

[14] J. Vince, Virtual Reality Systems, Addison-Wesley Publishing Company, Wokingham, England, 1995.

[15] A. Wood, N. Drew, R. Beale, and B. Hendley, "HyperSpace: Web browsing with Visualisation", Poster presented at *Technology, Tools and Applications, the Third International World Wide Web Conference*, April 10-14, 1995.

# The Usability of Interacting with the Virtual and the Real in COMRIS

Geert de Haan
IPO, Center for Research on User-System Interaction
Eindhoven University of Technology
P.O. Box 513, 5600 MB Eindhoven,The Netherlands
g.d.haan@tue.nl

## ABSTRACT

COMRIS (Cohabited Mixed Reality Information Spaces) [13] is a research project that aims to create a wearable assistant that provides context-sensitive information about interesting persons and events to conference and workshop visitors. Virtual and augmented reality applications generally require users to switch from one reality to another or to emerge themselves into a heavyweight VR environment. In Comris the two worlds are intimately linked and extended into each other without the need to switch or emerge.

In the physical world, the most tangible part of the Comris system is a small wearable and personal device "the parrot" that speaks in the user's ear and provides a few buttons to fine-tune messaging defaults to the circumstances and to respond to messages. The parrot also houses an active badge that keeps track of the whereabouts of its 'wearer' relative to beacons distributed around the conference premises. Conference visitors also have a number of information kiosks at their disposal to enter and adjust their interest profiles and to consult the conference schedule and their personal agenda.

In the virtual world, agents represent and attempt to match the possible (events and persons) and the actual interests (from the interest profile) of the users. A process 'competition for attention' selects the interests on the basis of the user's context and guards the user from information overload. Interests are subsequently translated into natural language and presented to the user as spoken messages.

After presenting an overview of Comris, this paper focuses on usability evaluation which deserves special attention because Comris users remain engaged in their regular working and social relations while receiving assistance and distraction from the virtual world. The paper describes the general approach and the plans concerning usability evaluation in the Comris project and discusses the preliminary results from the first experiment.

**Keywords:** usability evaluation, wearable computing, agent technology, mobile computing, natural language, speech output, augmented reality, context awareness

## 1 INTRODUCTION

During the last few years the amount HCI research on office automation has rapidly changed to the advantage of developments that promise the liberation of the user from the restrictions of the permanently fixed desktop office environment.

- Internet provided the means and architecture for world-wide information accessibility independent of specific locations
- Virtual and Augmented Reality showed that there is life beyond the 2D desktop space
- Ubiquitous computing and CSCW drew attention from the standard PC office applications
- Mobile and wearable computers are making place independent computing possible.

COMRIS (Cohabited Mixed Reality Information Spaces) [13] is a research and development project that may be seen as a logical continuation of the four developments and particularly, the way in which they are used together.

The project seeks to develop a wearable assistant for conference and workshop visitors. On the basis of a personal interest profile and an active badge system, conference visitors receive context-sensitive information about interesting persons and events in their spatio-temporal vicinity.

For example, a person who has indicated a high interest in wearable computers on her home-page or profile form may receive a message that "a demonstration of the parrot, a wearable conference assistant, is about to start in 5 minutes at the Comris booth".

To the user the most tangible part of the Comris system is a small wearable and personal device, mimicked 'the parrot' that every now and then speaks in the user's ear and provides a few buttons to fine-tune messaging defaults to the circumstances and to delete, repeat and elaborate messages. To the system, the parrot is also a transmitter that help keep track of the whereabouts of people.

Conference visitors also have a number of information kiosks to their disposal to enter and adjust their interest profiles and to consult the conference schedule and their personal agenda. Presumably, the information kiosks also provide for general internet facilities apart from to their use within Comris.

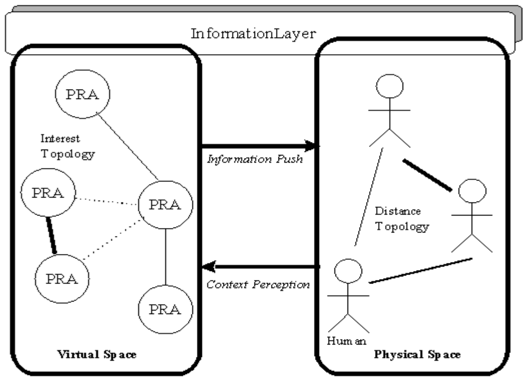
Only indirectly noticeable to the user is the virtual agent space that forms the software side of the 'cohabited information space' that Comris is about. In this space, agents represent the possible (events and persons) and the actual interests (from the interest profile) of all the users and compete for the attention of the user's personal agent that decides which interests are passed on for language and speech generation and output via the user's parrot.

## 2 THE COMRIS PROJECT

The Comris project is one of the thirteen I3 Esprit Long Term Research projects that were launched in 1997. I3 stands for Intelligent Information Interfaces and is pronounced eye cubed. It is an initiative from the European Commission DG III Industry to push research projects that explore and prototype radical new human-centered systems and interfaces for the interaction with information. The Comris project is centered around a number of key concepts.

### 2.1 MIXED REALITIES

Comris is build from two spaces which each have their own type of reality: the virtual reality of the software agents and the plain reality of the human users [13]. Many applications that use emerging technology in wearable computers, virtual and augmented reality and CSCW require that users to switch from one reality or one environment to another. Rather then emerging the human user into a heavyweight virtual reality environment or requiring the use of a web browser, in Comris the two worlds are intimately linked and extended into each other without the need to switch environments.




### 2.2 INFORMATION PUSH

Comris aims to move the prevailing information pull model toward an information push model. In the information push model, the information is actively pushed onto the user. Push technology, once a buzzword has generally failed because it failed to take account of the appropriateness of the information within the user's context. In Comris the user's temporal and spatial context are not only taken into account but are decisively used by the personal agent to determine the appropriateness of information.

### 2.3 COMPETITION FOR ATTENTION

A key problem in a radical information push approach is to avoid an overload of pushed information. In Comris this is achieved by a model of competition for attention [14].

The user's attention is limited, so the agents need to compete for it. Every user has a personal agent as a watchdog that guards the priorities among agents and the frequency with which messages are passed through, using information about system and user preferences, the messaging history and the user's context.

### 2.4 INTEREST-BASED NAVIGATION

A feature of the virtual space is that its topology is not like real space. Rather it is structured to reflect potential for interaction, not physical, as in the real space but interaction that is likely to furthers the agent's interests. Interest based navigation is the process by which agents find out about useful places, or encounters with other agents.

There is also a user-side to interest-based navigation. Evidence has pointed out that it is difficult to instruct agents beyond the most simple tasks. In a system with multiple agents it is difficult to figure out the meaning of setting the interest in one subject relative to another. A sub-project of Comris investigates the effects of allowing users to manipulate the positioning of their interests in a virtual space, among other interests and among those of other users.

## 3 TECHNICAL DESIGN

Technically, Comris follows the mainstream of wearable communication applications: it uses a standard TCP/IP network, a Java kernel and messaging layer, and Lisp agents that communicate via secretaries. The agent communication protocol derives from the structure, objects and the tasks in the conference domain as an explicit representation of the "rules of engagement" among agents at conferences [11]. All communication between the agents and with the user takes place according to a standard protocol that uses XML messages. On the input side of the system, the information kiosk is a set of web pages, and by filling out an interest profile a user creates a number of messages like:

**(interest_in_topic ((robotics, 2), (wearables, 4))**

Likewise, on the output side, an agent may have won the competition for attention with a message like the following, that is subsequently passed on to the language generation module:

**(proximity alert (event-type = session),**
**(INTEREST (TOPIC, (topic = robotics), 4),**
**(PRACT (TIME (delay = 5 mins)),**
**(PLACE (room = (name, ID L45))) )) )**

Because the language domain of Comris -messages about conference events- is fairly restricted, there is no need to use full blown language generation and instead an approach called "template based language generation" [3]. This is also to enable scaling-up of the Comris system and to keep within the limited resources of the wireless internet connection and the wearable computer. In template based language generation, each message consists of a given template with a number of variable slots that may provide additional information., for example:

**[event slot] on [subject slot] is about to start [time slot] [place slot]**

The results of language generation may be presented at the information kiosk, but also sent wirelessly to the speech generation module that is located in the parrot.

The Comris parrot is a small wearable computer with a few buttons, at least an audio output device such as a earphone, and a small LCD screen. The LCD screen will not be used for regular information display but it may be used to label the buttons. The parrot is further provided with a two-way FM radio to receive textual messages and to transmit the status of a sensor.

The sensor detects its location relative to a number of beacons which are distributed around the conference premises. The speech generation module adds the appropriate linguistic and prosodic properties to the textual message and produces a verbal message that might say:

**A session on robotics is about to start in 5 minutes in the auditorium**

## 4 USABILITY AND COMRIS

The Comris project is a co-operation between a number of universities and research institutes, as follows (with the specific research contribution shown between brackets):

- Starlab - Riverland Next Generation, Belgium (wearable computers),
- Artificial Intelligence Research Institute, Spanish Scientific Research Council (intelligent agents),
- IPO, Center for Research on User-System Interaction, Eindhoven University of Technology (spoken language interfaces),
- Visualization and Media Systems Design group, GMD, Germany (information visualisation),
- Interactive Systems Research Group, University of Reading (distributed virtual reality),
- Artificial Intelligence Group, University Dortmund (machine learning),
- Artificial Intelligence Laboratory, Vrije Universiteit Brussel (language generation).

Because the Comris project involves so many partners, a management structure and planning schedule were set up to streamline the project. A noticeable disadvantage of Comris as a large managed and technology oriented project is that usability issues only entered into the process relatively late, during the review of the initial phase of the project. After the specification of requirements, the architectural and domain models it became apparent that creating a technologically successful intelligent personal assistant is one thing, but that creating the assistant in such a way that it is actually useful and people are willing to use it is quite another.

In 1998, four years after the publication of her seminal paper on personal agents Pattie Maes writes: "In the process [of developing agents that reduce work and information overload], we learned some important lessons about what ultimately makes an agent successful, that is, accepted and used by people on a daily basis. Specifically, we learned that it is not so much the underlying algorithms that represent the hardest challenge but rather the design of the user interface between the agent and the user" [7].

With respect to Comris and similar projects, several critical barriers exist that may block the transition from a technologically successful project to a successful user product.

### 4.1 SOCIAL-USAGE BARRIERS

The social-usage barriers concern the fact that Comris is 'communityware'. In contrast to information systems for strictly personal use like agenda's and personal digital assistants (PDA's), Comris requires a sufficient number of people that actually make use of it and invest into using it by explicitly specifying information about themselves and their interests. Similar to telephone and email, Comris needs a critical number of active users to gain acceptance and fully provide all of its advantages.

Also, it is easy for people to reject the Comris system. The use of many computer systems eg. in the office is embedded within an organisational context that prescribes the use of those systems. No such formal embedding exists for using the Comris system.

On the contrary, because Comris uses a small personal device it is easy for the user not to wear it or to switch it off. This will especially occur when the Comris parrot does not provide sufficient functionality from the point of view of its users. It will be much easier for Comris to gain ground when it provides an alternative to the information appliances that people use at conferences, like notebooks, telephones, etc., instead of being an additional thing to carry along.

### 4.2 INTERFACE-TECHNICAL BARRIERS

As regards the interface-technical barriers there is the odd situation that two main features of the Comris technology also form the greatest risk factors in rejecting the system by the users: the use of push technology and the use of speech as the main output mode.

Advisory systems are not readily accepted if they do more than to answer questions on the user's initiative. Actively pushing information onto users creates the risk of disrupting the user's work process, especially when the advice is at the wrong level of abstraction [2].

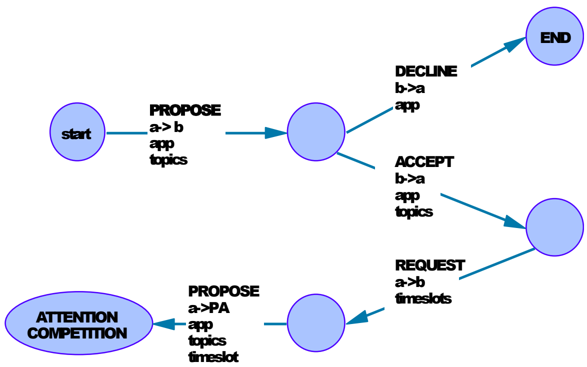
In Comris, advice information is presented by means of sound which is a particularly intrusive type of information. Auditory information is relatively hard to ignore in the form of speech, which may easily interfere with other speech signals in contexts like conferences and social gatherings where verbal speech is the main mode of communication.

Because of the threads imposed by these barriers the initial plan to perform mere evaluative usability studies was changed in favour of formative studies, and from a detailed level of investigating (e.g. what type of voice to use) to a higher level with questions involving user interaction and user preferences.

## 5 INTERACTING WITH AGENTS

From the starting point that the Comris system should provide assistance to attendees of large conferences, the opportunities for user interaction were determined in a number of subsequent design steps. Defining the functionality in Comris starts with an analysis of the domain model which describes the information that is available within the system, and a list of possible tasks.

For each of these tasks, a scene is created that depicts, first at a coarse level and gradually refined, the interactions between agents and between agents and users. The following figure shows at a coarse level the task shared by two agents to make an appointment between two users, based on either the initiative of one of them or a mutual interest in both their interest profiles.




Alternative scenes may show a different allocation of roles and (sub) tasks among agents as well as a different interaction pattern. From the scenes follow the actions that the user as a special agent may or may not execute. In analysing scenes, the interaction between the user and the parrot or the information kiosk remains largely implicit.

The analysis of agent-scenes does not yield conclusive answers because Comris users may or may not call upon the assistance of agents to do things like making appointments with other people or plan meetings. Assistance is not a singular concept since both the amount and the type of assistance may vary. From a software perspective, which options to offer to users has a profound influence on the design of the system because of its effects on the design of the agents, the agent communication system, the information kiosk and the parrot.

From a user point of view, how much assistance the agents should provide is a matter of user preferences; some users may only want to be offered suggestions whereas others may want their agents to decide about things like the time and place of the appointment.

The design of, for instance, the appointment mechanism is partially taken care of by the analysis of agent-scenes: given that appointments involve information about people, a time and a place, decisions can be made about how to exchange and decide on such information. It is not clear, for instance, whether users who want to make an appointment prefer a role in which they may only accept or reject appointment proposals from their agents, or that they prefer to make appointments by themselves or that they prefer a role in which they are able to suggest the time and/or the place of an appointment.

Initially, the usability studies were meant to be evaluative and to determine questions like, which voice should be used, how many buttons the parrot should have, etc. Provided that the analyses of agent-scenes do not provide complete information about how the interaction between users and agents should be designed to be satisfactory from a user's point of view, it may be clear that also at the conceptual level usability issues exist, and that there is also a need for formative usability evaluation studies.

## 6 USABILITY STUDIES

User interaction in Comris occurs via both the information kiosk and via the parrot, and because the parrot is by far the most restrictive device, it is regarded as if it were the only one. Because the information kiosk interface is only an internet browser displaying a number of HTML and VRML pages it is very flexible.

The Comris parrot puts forward more restrictive demands, with respect to both, the design of the hardware and as a personal device. For these reasons and because of limited resources, the usability studies at IPO focused on the functionality and interactivity of the parrot and the characteristics of the speech output.

The aim of the Comris project is the development of a usable product. According to the state of the art in human-computer interaction this would require that much attention should be spend on user requirements analysis and on task- and domain analysis, in short iterative cycles in which usability- and technical engineers co-operate closely with prospective end-users.

Unfortunately, given the schedule of the project and the resource limitations this was not possible. Also, in order to attain knowledge about characteristics of speech interfaces in general, it was decided to investigate the most critical issues experimentally and to use questionnaires and rating scales to collect information about subjective experiences and user requirements.

## 7 EXPERIMENTS PLANNED

Three usability experiments are planned, each with a similar setup. Note that in this paper only the results of first experiment will be reported.

The first experiment concerns user preferences regarding message presentation i.e. dealing with the results of information push.

Presenting a message involves a trade-off between exerting control, according to the ISO requirement that the user should be in control of the interface [4] and being interrupted by a message, according to the principle of information-push. Being spoken to may be equally distracting as telling someone to shut up. On the level of individual message presentation, this experiment investigates the use of meta-messages to announce the availability of information without the need to listen to the whole message or to press any buttons to indicate that the message is not welcome.

To address the lack of user requirements during the initial specification of the system, the subject's task in the individual working session consists of a small requirements analysis. Guided by a questionnaire and probed by examples of related technologies (e.g. buzzer, walky-talky, love getty) subjects are asked to draw, describe and list the required function for an ideal system to assist conference visitors.

The second experiment investigates ways that allow users to set interruption levels and to adapt these to the immediate circumstances. Message frequency i.e. the competition for attention threshold primarily depends on the user's preference settings. Momentary changes may be provided for in different ways: either automatically but coarsely by the user's personal agent, or manually via the parrot's button interface by user itself, or by a combination of the two.

In addition, this threshold influences the number of messages that is passed on to the user but it may also influence the form of messages between short meta-announcements and longer informative messages. Also in this experiment subjects will perform a user requirements study during the individual working session. This time the subjects are be specifically asked to describe and list the requirements for the wearable device.

Finally, the third experiment concentrates on interacting with the Comris parrot. People use the parrot for simple tasks like repeating, postponing, and deleting individual messages and for more complex tasks that involve interaction with the agent space.

Because a formal analysis of the agent-scenes does not provide sufficient information to decide about all issues, the third experiment is used to decide between promising alternative solutions and to answer questions like if people want to decide separately about times and places, and whether to use priorities among appointments.

## 8 DESIGN AND SETUP

The purpose of the experiments is primarily to create a conference environment that allows the subjects to use and judge different ways of presenting and controlling messages in would-be circumstances.

In three successive sessions, during a lecture, during a lunch break and while working on their own, subjects receive Comris messages in a situation that closely mimics the circumstances that occur while attending a workshop or a conference. The type of sessions systematically represent the situations at a conference or workshop in which attendees are involved in working relations, social interaction, or both.

During the requirements task - the individual work session - it is assumed that subjects are influenced by attempting to reach work goals but not influenced by social rules. While attending a presentation - the lecture session - it is assumed that the use of the Comris parrot is subjected to both social rules as well as work-oriented rules. During the lunch break - the leisure session - it is assumed that work goals are virtually absent whereas social rules are strongly present.

Systematically varying the work and social contents of sessions may allow conclusions about the relative importance of these factors. This is particularly important because in Comris, contrary to most VR application in which behaviour entirely takes place in an artificial reality, the personal assistant breaks into a persons social and working environment.

To determine preferences concerning the trade-off between exerting control, or pressing buttons to accept or reject the presentation of a message, and being interrupted by a message, or being spoken to while engaged in work or social interaction, messages are presented in three conditions:

- a tone-only condition in which messages are presented without user control,

**<dingdong><a message>**

- a tone-and-choice condition in which users control the presentation of messages,

**<dingdong><a yes or No choice><a message>**

- an information-and-choice condition in which users control message presentation while being provided with (and possibly interrupted by) information about the type of message,

**<dingdong><message type description><a yes or No choice><a message>**

The message types consist of one-word description such as "reminder", "proposal" and "attention" for respectively appointment reminders, meeting proposals, and proximity alerts. During each session, subjects received 17 messages in the same order with an interval varying from 2 to 3 minutes.

The order and combination of sessions and conditions was balanced according to a greek-latin square; in the following schema, each subject was randomly assigned to a row:

| | | |
|---|---|---|
| tone-only lecture | choice work | info-choice lunch |
| choice lunch | info-choice lecture | tone-only work |
| info-choice work | tone-only lunch | choice lecture |

The subjects' main task, if a task at all is to listen to these messages and whenever appropriate, to act accordingly. Examples of the messages are as follows:

- A tutorial on wireless networks, chaired by Clara Lodge, is about to start in room 22.
- Mister Richard Davies, who has a high interest in speech interaction, is at this moment in the same room as you.

- Misses Susan Bowen, from Computer Soft, proposes to meet you, to discuss augmented reality, on friday at 3 o'clock.
- A presentation on ubiquitous computers, involving Pauline Sears as a speaker, is about to start within 5 minutes. Do you want to sign-up for this presentation?

The messages presented by a high-quality female voice are similar in form and content to genuine Comris messages except that, in the experiment, they are not adapted to a personal interest profile. In addition, awaiting the creation of a new and much improved UK-English diphone database at IPO, AT&T's text-to-speech synthesiser for US-English was used [1] which is currently regarded as one of the best for US-English.

The Comris parrot was emulated by providing each subject with a combination of a wireless FM headphone (Philips HC450) and a wireless FM remote control (Philips SBC SK310). Because of serious noise and interference problems with the headphone sets that prevented running the experiment with more than 2 subjects simultaneously, subsequent experiments will use FM transceivers.

As subjects served members of IPO's junior research staff (PhD students/AIO's). All subjects were familiar with computers and used them on a daily basis, all had at least a good understanding of spoken English, and all had at least some experience with attending conferences and with voice systems. The subjects participated in the experiment in the course of their regular daily schedule. Care was taken to ensure that all sessions took place during one day.

Fitting the experimental schedule to the subjects daily schedule generally worked well. However, due to the combination of fitting the greek-latin square into a subjects schedule and the problems with the wireless FM headphones, only 9 subjects participated in the experiments whereas plans included 18 (or more) subjects.

During the sessions observations are made of the subjects behaviour with respect to reacting any of the messages by listening, acting, pressing buttons, etc. and with respect to side-effects like shifting attention, showing signs of distraction, etc.

At the end of each session the subjects are requested to fill out several questionnaires with questions derived from general usability questionnaires like SUMI [5] and QUIS [12], complemented with questions specific to the Comris system, items about the subject's experience with the conferencing domain and messaging systems, and a request for general comments about Comris and about the experiment. The questionnaire items in the first experiment include the following:

- In this session message presentation was controlled by you: you could decide to listen to a message or not. How do you evaluate the way in which message presentation was controlled?
  [Negative] 1 -- 2 -- 3 -- 4 -- 5 [Positive]

- Briefly explain how you like messages to be controlled and why.

- How do you evaluate the understandability of messages?
  [Negative] 1 -- 2 -- 3 -- 4 -- 5 [Positive]

- You have been presented with messages in three different ways. Which of these did you like best?
  - ❑ <announcing tone><message>
  - ❑ <announcing tone><yes/No><message>
  - ❑ <announcing tone><type><yes/No> <message>

Apart from collecting preference and performance data to decide on specific design choices, part of the experiments aims to collect data about user requirements. Where possible these requirements will be used to support the design of the Comris prototype.

In the next phase of the Comris project, the design of the production system, the requirements will play dual role: they become the starting point for a more user-centred approach to the design and they will serve as the political means to forward a user-centred design approach.

In the current technology-oriented phase of the project usability issues have been largely treated as a side-issue. Delivering a working prototype will strengthen the design community that has evolved that a technological orientation pays off. Without data that shows the need to do otherwise, it is unlikely that a sudden change to user-centred design will occur.

For this reason, a user requirements analysis has been 'creatively' designed into the experiments as the task for the subjects in the individual working condition. The requirements task present the subjects with a question like the following, taken from the first experiment and aiming at the functionality requirements of the Comris parrot:

> **Please describe in your own way (e.g. drawings, sketches, lists of functions, textual descriptions, etc.) how you would like to use an intelligent advise system with a wearable personal assistant. Although interest is most in the functionality of such a system (e.g.**

**facilities, functions, services or commands), you are most welcome to describe the look, feel, smell or whatever of your ideal assistant.**

Because it is very difficult to answer such a question 'out of the blue' and to keep the subjects from assuming they have to describe the system as it was used in the experiment, the task is probed by brief factual and visionary descriptions of the Comris system, by a list of features or things to keep in mind, and by a sheet with pictures of comparable systems (such as message pads, hearing aids, etc.).

The results of the requirements analysis task that the subjects performed during the working sessions, in the form of sketches and drawings, lists of features and functions etc. will be collected for qualitative analysis.

# 9 RESULTS

At the time of writing 9 subjects participated in the experiment. For one subject the session data were discarded because one of the questionnaires had not been filled out.

Given the number of subjects it should not come as a surprise that analyses of variance did not produce usable results. Although this is not desirable result from a scientific point of view, different authors (eg. [6]) have expressed the view that, at least for product development, significance levels are often useless or misleading.

## 9.1 OBSERVATIONS

The observations during the sessions did not show any clear results. When subjects were presented with a choice about message presentation, they mostly but not always made the choice to having the message presented to them within seconds.

When choosing not to listen to a message, the choice was usually but not always made by not reacting which led after a half minute to the default 'no' response.

During the lecture sessions subjects were more eager to use the 'yes' and 'no' buttons in comparison to the work sessions and especially in comparison to the lunch sessions. Although the subjects were informed that they were able to give a 'no' response by not using the remote control at all, it seemed as if they really discovered this option when engaged in personal communication.

## 9.2 CLOSED QUESTIONS AND RATING SCALES

After each of the three sessions, the subjects filled out a questionnaire with the same items repeated for the specific condition and session. Compared to the end questionnaire there is more random variation; as such, only the most important results are mentioned.

When asked to evaluate the way in which messages were controlled, the subjects in the information-and-choice condition have higher preferences (Mean 3.3, Median 5, Mode 5) then those in the choice condition (Mean 3.1, Median 3, Mode 3) and the tone-only condition (Mean 2.1, Median 1, Mode 1).

Interestingly enough, where it concerns evaluating the way in which messages were announced, evaluating the pleasantness to receive messages, and evaluating the pleasantness of the system it is slightly but consistently the subjects in the choice condition who have higher preferences. The tone-only subjects consistently give the lowest preferences.

The preferences scores that the subjects give to message announcement, enjoyability, control and the pleasantness over sessions do not follow a discernible pattern, which is good. The subjects preferences from the work session however are consistently higher then those from the lecture and the lunch sessions. This may indicate a difference in tolerance.

The items of the end questionnaire asked for direct preferences between the different ways of presenting and controlling messages. In addition, subjects were asked to express how they would like to present and control messages presentation in lecture, leisure and working circumstances. When asked who should control message presentation, the subjects are very clear about that the user should be in control and not the system. When asked about control in general, the vote is 5:2 and when asked who should be in control during a work session, a lecture session, and during a lunch break, the votes are 6:2, 6:3 and 8:1 respectively.

When forced to chose a best alternative from the message presentation conditions, there is only one clear worst condition: the choice condition only gets 1 vote while the tone-only and the information-and-choice each receive 4 votes.

When asked to rate each condition separately, the information-and-choice condition is generally favoured (Mean 3.3, Median 4, Mode 4) more than the choice condition (Mean 2.4, Median 2, Mode 2) and the tone-only condition (Mean 2.3, Median 2, Mode 1). It should be noted that the tone-only condition has no uniform answer pattern because 3 subjects assigns it with a score of 4 on a scale from 1 to 5.

When the subjects are asked how they prefer that messages are announced with or without information about the type of message, the votes are 5:3 when asked in general, and when asked specifically for each session-type, they are for work sessions 5:1, for lecture sessions 6:1, and for lunch session 3:4.

## 9.3 REQUIREMENTS ANALYSIS AND OPEN QUESTIONS

The results of the open questions and the results of the requirements analysis were listed together. Remarks that were made more then once and those that seemed particularly significant to the design of the Comris parrot are discussed.

The open questions indicated that regardless of the instructions it was difficult for some subjects to imagine that Comris will actually be different from the way it was simulated during the experiment. For example, several subjects assumed that they had to write down appointments to remember them for themselves, that they themselves had to keep track of double appointments, and that it would be useful if the parrot warned them when events were starting.

Such misunderstanding may also contain useful information. For example, there was a suggestion that Comris should warn about double appointments but with the addition that it should ask which appointment to keep. This may indicate that the mechanism that Comris agents use to keep schedules, which assumes that appointments are made once-and-for-all, may not be similar to how people deal with schedules.

Regarding receiving unsolicited advice, several subjects expressed rather negative attitudes towards agent assistance and referred to eg. Microsoft's Office Assistant or 'the paperclip' as examples of equally distracting and unhelpful assistants. At least three subjects stated that they should be in control instead of merely being allowed to press buttons. More then half of the subjects disliked the use of audio signals and asked for information to be presented on a visual display so they could decide where to pay attention to.

Quite a number of subjects asked for a facility to switch-off the parrot in order not to be bothered for a while or as an alternative to pressing buttons all the time. Several subjects remark that they didn't mind receiving messages while they worked, but that they did while talking to others during leisure or while listening to the lecture.

Some, but not many subjects remarked that the messages were not interesting to them, especially when they were repeated thrice. If they would have been more interesting, they would not mind as much as they did.

Regarding design details numerous suggestions were made to improve the design. There should not be a demand to respond within a half minute to have a message presented, the message type information should be more varied and informative, and different tones might be used instead of the distracting one-word descriptions, messages should be announced on a visual display or by vibration.

Several subjects remarked that messages that do not require an answer such as reminders should be presented without a choice. About the tone and the message information word it was regularly remarked that these did not provide enough information for deciding to listen or not to listen to a message.

There was much criticism about the earplug. In order not to distract neighbours, a closed earplug was used in one ear. Various subjects remarked that the in-ear plug irritated, partially blocked the regular hearing capabilities, and that it made it difficult to ignore the messages in the same way as it is possible to ignore sounds and voices in the environment.

Many of the subjects offered suggestions for smaller and bigger enhancements of the functionality of the parrot device. Not too far-reaching are the suggestions to use it for message navigation and to switch-off, skip, delay, save and repeat messages. It was also proposed to allow delaying specific types of messages such as meeting proposals or questions, to provide alternative voices, warning tones, etc., and to determine when and when not reminders should be provided.

More advanced are the suggestions to have the parrot provide facilities for 2-way radio, voicemail, icq, to warn when friends or foes are near, and to provide background information about people in the vicinity or people being looked at. Several subjects suggested that there should also be facilities to use the information that was collected at the conference back home.

Many suggestions were made about proving schedule overviews, descriptions of workshops that are about to start, when they start, and conference status overviews; eg. what is happening next? A few subjects suggested that the parrot be used to help find ones way around the premises and have it provide where-am-I, where-is-the-toilet, and how-do-I-get-there information. The price for the wildest suggestion is for the suggestion to have the parrot spread the scent of coffee at the start of a break.

# 10 CONCLUSIONS

This section discusses four conclusions; three from the results of the questionnaires, and one from performing the experiment itself.

First, from both the closed and the open questions it is clear that certain ways to control messages presentation and to present messages are more, and other are less preferred. The best ways to present messages is to announce them by a short tone or verbal description that summarises the contents of the message. However, not all the subjects prefer this solution in all circumstances. As such, it seems best to use the information-and-choice mode as a default way of presenting messages and to allow users to pick a different way depending on how they evaluate the circumstances.

A second conclusion that the results indicate is a strong need from many subjects to be in control over the system. Some of the suggestions have already been taken into account in the design proposals of the parrot: it should be possible to eg. switch-off messaging, and to repeat, delay and skip messages. At least for some subjects, message navigation does not go far enough since they want to be explicitly in control. This may be due to some kind of technological conservatism but it may also be that information push is only acceptable when it is possible to ignore such information as easily as it is to ignore information in the environment.

In many VR environments, users enter an artificial reality (when it suits them) and accept the information they receive as part of such an environment. In Comris, the users remain in the real world and the information provided to the users must therefore adhere to the rules of the real world. It is presumably not too difficult to have messages announced or even presented to users by means of modalities or devices that are less distracting, but it will be difficult, if at all possible, to redesign Comris into an information pull system. For the time being, it seems best to seeks ways to provide people with advice information, without, however to push it into their consciousness.

In the third place, the usability experiments have been setup to determine the interaction opportunities of the Comris parrot. The aim has been to find the best way to allow users access to the functionality of the Comris system. From the subjects remarks about improving the functionality of the parrot, a slightly different picture arises: the parrot is not only a device to provide access to Comris' functions but rather a device that may enhance visiting conferences beyond what has been planned in the project.

Suggestions to use the parrot for eg. voicemail, icq and as an external memory for use back home give a clear indication that usable systems do not arise from usability experiments alone but that they start with taking the user's wishes and requirements into account, right from the beginning of a project, during the formulation of its goals.

For the last conclusion, it is necessary to refer back to the original purposes of our investigations. On the one-had side, the purpose was to collect scientific information about the use of speech output in user interfaces. The general opinion is that this purpose is best served by an experimental approach. On the other had side, the purpose was to collect usability information to improve the design of the Comris parrot. Looking backward, and even taking into account the external factors that made experimenting more difficult, the conclusion seems warranted that effort spend on setting up and performing the experiments does not weight well against the outcomes.

It may be better, even with respect to collecting scientific evidence, to learn from experience rather then to learn from experiments. As such, we are currently investigating the possibilities to use eg. focus groups and incremental usability evaluations, at least, as the means to complement experimental studies.

## ACKNOWLEDGEMENT

We thank our partners in the Comris project for their co-operation and our colleagues within IPO (Center for Research on User-System Interaction, Eindhoven, The Netherlands) for their support in setting up the usability evaluation studies.

## REFERENCES

[1] AT&T Research, The Next-Generation Text-To-Speech Synthesiser. Available from http://www.research.att.com/projects/tts/, 1998

[2] Fischer, G., Nakakoji, K., Ostwald, J., Stahl, G., Sumner, T. Embedding Computer-Based Critics in the *Contexts of Design. Proceedings INTERCHI'93, ACM*, New York, 157-164, 1993.

[3] Geldof, S., Van de Velde, W. An architecture for template based (hyper)text generation. In*: Proceedings of the 6th European Workshop on Natural Language Generation- EWNLG'97*, Duisburg, Germany: Gerhard Mercator Universitaet, Inst. fuer Informatik; pp 28-37, 1997.

[4] ISO Ergonomic requirements for office work with visual display terminals (VDTs) - *Part 10: Dialogue principles. The International Organisation for Standardisation (ISO),* 1996.

[5] Kirakowski, J. The Software Usability Measurement Inventory: background and usage. In: Jordan, P.W., Thomas, B., Weerdmeester, B.A. and McClelland, I.L. (eds).. *Usability*

*Evaluation in Industry.* Taylor and Francis, London, 169-177, 1996.

[6] Landauer, T.K. Let's get real: a position paper on the role of cognitive psychology in the design of humanly useful and usable systems. In: Carroll, J.M. (ed.) *Designing Interaction: Psychology at the Human-Computer Interface*. Cambridge University Press, Cambridge, UK, 1991.

[7] Maes, P. Reflections on ... Agents that Reduce Work and Information Overload. In: Maybury, M.T. and Wahlster, W. (eds.), *Readings in Intelligent User Interfaces*, Morgan Kaufmann, San Francisco, USA. pg. 536, 1998.

[8] S. Mann, Mediated Reality, Tech. Report TR 260,

[9]

[10] MIT Media Lab Perceptual Computing Section, Cambidge, Mass., 1994.

[11] E. Plaza, P. Noriega, C. Sierra. The Conference Center as an Agent-Mediated Institution. *Proceedings of the first International Workshop on Agents in Community Ware (ACW'98).* La Villette, Paris, France, July 6, 1998, pp. 73-82, 1998.

[12] Shneiderman, B, *Designing the User Interface: Strategies for Effective Human-Computer Interaction, 2nd ed.*, Addison-Wesley, Reading, Mass, 1992.

[13] Van der Velde. Co-Habited Mixed Reality. *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence*, Aichi, Japan, August 23-29,. Also available as: http://comris.starlab.org/papers/ijcai-97.ps, 1997

[14] Van de Velde, W., Geldof, S., Schrooten, R. Competition for attention. In*: Proceedings of ATAL: Workshop on Agent Theories Architectures and Languages*. Providence, Rhode Island, USA, July 1997. Also available as: http://comris.starlab.org/papers/atal-97.ps

# ECHOES: A Collaborative Virtual Training Environment

G.M.P. O'Hare, A.J. Murphy, T. Delahunty and K. Sewell
PRISM (Practice and Research in Intelligent Systems & Media) Laboratory,
Department of Computer Science,
University College Dublin (UCD), Belfield, Dublin,Ireland
{Gregory.OHare| Aidan.Murphy| Thomas.Delahunty| Katherine.Sewell}@ucd.ie

## ABSTRACT

The main objective of the *ECHOES*[1] project is to build a distributed dynamic environment for educating and supporting technicians in the use and maintenance of complex industrial artefacts. To pursue this objective, Computer-Web Based Training, Virtual Reality and Multi-Agent Systems are integrated and synthesised into the ECHOES environment. The ECHOES virtual training environment remotely immerses the engineer or trainee into a familiar virtual environment and a virtual community where they can receive information and assistance, communicate and virtually meet with colleagues and collaboratively solve tasks. This environment is delivered via the internet through VRML (Virtual Reality Modelling Language).

**Keywords:** Collaborative Virtual Environments, Multi-Agent Systems (MAS), Computer Aided Training, Virtual Reality.

## 1 INTRODUCTION

This paper will present some of the results of the ECHOES (Educational Hypermedia On-linE System) Project. The main objective of the ECHOES project [1], [2] is to build a distributed dynamic environment for educating and supporting technicians in the use and maintenance of complex industrial artefacts. To pursue this objective, Computer-Web Based Training, Virtual Reality and Multi-Agent Systems are integrated and synthesised into the ECHOES environment. These technologies are used to aid users at different levels of complexity, starting from the novice, who merely wants to quickly develop a global functional view of complex systems, up to the technician, who needs a strong conceptual understanding of the underlying behaviour of complex equipment.

The ECHOES system is a complex multi-user application which seeks to: -

- help technicians to build a strong conceptual understanding of complex equipment in order that they may deduce appropriate diagnostic steps in cases of equipment failure;
- increase the structural and functional knowledge of trainees by delivering structural knowledge to them in a simulated Virtual Reality (VR) environment;
- enable novices to quickly develop a global functional view of complex equipment and to enable them to quickly interact with the appropriate systems expert;

In this context it constitutes a specific instance of the broad class of system namely Collaborative Virtual Environments (CVEs). The ECHOES training environment remotely immerses the engineer or trainee into a familiar virtual environment and a virtual community where they can receive information and assistance, communicate and virtually meet with colleagues and collaboratively solve tasks. This environment is delivered via the internet through VRML (Virtual Reality Modelling Language) [3] where the user presence is depicted through an avatar which is moved through the space by way of the External Authoring Interface (EAI) dynamically updating the .wrl file.

The ECHOES architecture is designed using a multi-agent approach. The user interaction with

[1] ECHOES (European Project Number MM1006) is partially funded by the *Information Technologies*, *Telematics Application* and *Leonardo da Vinci* programmes in the framework of Educational Multimedia Task Force.

the system is agent mediated and the chosen interface is that of the visit metaphor. Agent Factory [4,5,6] a multi-agent development environment developed by one of the authors is used to rapidly prototype the agent community which delivers the system functionality.

The ECHOES virtual environment is based upon the familiar physical metaphor and trainees can wander a spatial environment entering rooms appropriate to the assistance they desire. At a functional level the following rooms are provided; meeting room, library, training room, simulation room, first aid room and control room. Various interaction modalities are supported. Users can communicate by email, navigate the virtual world, retrieve and view technical manuals in the library, participate in a virtual meeting, engage trouble shooting assistance in the first aid room, undertake courseware modules in the training room, either in a classroom context or in trainee moderated individual training booths.

This paper will describe in detail the capabilities of the ECHOES environment and the underlying agent based architecture which supports this functionality. The purpose of section 2 is to give an overview of similar and related research conducted in this field. Section 3 initially provides a high level description of the ECHOES system, while section 4 follows up with a more detailed treatment. Section 5 depicts system interaction. Section 6 offers some conclusions.

## 2 BACKGROUND RESEARCH

The objective of the ECHOES project is to provide an effective training environment for using and maintaining complex equipment through a distributed virtual environment. There has been much research into the area of such Collaborative Virtual Environments (CVE's). By CVE's we refer to multi-user virtual reality systems which explicitly support CSCW. In this section we provide a brief roadmap of some of the research which has influenced and motivated the ECHOES project.

One of the formulative pieces of research conducted in this arena was that of the Distributed Interactive Virtual Environment (DIVE) [7]. This constitutes an internet-based multi-user VR system in which participants navigate a 3D space and see, meet and interact with other users and applications. It was developed by Scandanavian Institute of Computer Science (SICS) as an experimental system for CSCW utilising a 3D virtual environment. DIVE is based on peer-to-peer communication with no servers, where peers communicate by reliable and non-reliable multicast, based on IP multicast. The peer-to-peer approach without a centralised server means that as long as any peer is active within a world, the world along with its objects remains *alive*. Consistency and concurrency control of common data (objects) is achieved by active replication and reliable multicast protocols. That is, objects are replicated at several nodes where the replica is kept consistent by being continuously updated. Update messages are sent by multicast so that all nodes perform the same sequence of updates. Conceptually, the shared state can be seen as a memory shared over a network where a set of processes interact by making concurrent accesses to the memory.

A participant in a DIVE world is called an *actor*, and is either a human user or an automated application process. An actor is represented by an avatar, to facilitate the recognition and awareness of ongoing activities. The avatar could be used as a template on which the actor's input devices are graphically modelled in 3D space. DIVE implements a variety of embodiments. The simplest are the 'blockies' which are composed of a few basic graphic objects, which is sufficient to convey a sense of presence, location and orientation [8]. The system provides static cartoon-like features for the 'blockie' to suggest that the blockie represents a user. A user can to some extent personalise their 'blockie' by changing the body shape associated with it. A more advanced embodiment, for immersive use maps a static photograph onto the face of the body. Video conferencing users may be represented using a video window.

The COVEN (COllaborative Virtual Environments) [9] project was a European ACTS project developed to address the technical and design-level requirements of VR-based multiuser collaborative activities in professional and citizen-oriented domains. COVEN aimed at developing a computational service for teleworking and virtual presence. The overall objective of the project was to provide the facilities needed to support future co-operative teleworking systems and to demonstrate the added value of networked VR for both professional users and home users. COVEN was designed to attempt to bring together a wide range of fields, such as communication infrastructures, CSCW, Virtual Environments, Human-Computer Interaction and VR animation to provide support for a European distributed virtual environment. Target key issues are collaboration support within VEs (including awareness, communication, group interaction, etc.), corresponding requirements at network and platform level (involving concern for scalability and continuous media support), and related Human Factors aspects.

The COMIC (Computational Mechanisms of Interaction in Co-operative Work) [10] initiative undertook to investigate techniques and to develop tools for real-world large scale CSCW application developers. It adopts a multidisciplinary approach to try and understand the process of group communication more fully. It aimed to examine and overcome the practical and theoretical problems limiting effective CSCW product development at the moment. The project consisted of four main research strands dealing with: the organisational setting of CSCW systems, the derivation of requirements for the design and development of CSCW systems, the assessment and augmentation of notations to describe and represent co-operative activities and the development of novel interaction mechanisms for CSCW. One of the key concepts developed in the co-operative system in COMIC is the spatial model, allowing interaction between inhabitants of large-scale virtual environments. The space within the virtual environment is populated with objects, both human and non-human (e.g. artefacts and data). These interact with each other according to a set of concepts: aura, awareness, focus and nimbus, which define the range of perception and interaction of the objects. For instance, the focus of an object defines its awareness of other objects, while its nimbus defines how aware other objects will be of it. When the auras of two objects collide the objects will interact. This interaction takes place via media which represent traditional communication media such as audio and video, and via more object-specific interfaces. The various concepts are implemented on several platforms.

MASSIVE (Model, Architecture and System for Spatial Interaction in Virtual Environments) [11] is a VR conferencing system which realises the COMIC spatial model of interaction. MASSIVE provides textual, graphical and audio client programs (usable in any combination) to allow users to communicate by graphical gestures, typed messages or real-time packetised audio. Text users may interact with graphical users and vice-versa. There may be any number of worlds with portals to move between worlds. MASSIVE embodiments are based on DIVE blockies, automatically labelled with the users name to aid identification. MASSIVE has been used in the COMIC project for international trials across Europe (e.g. 23/5/95: 9 users, 4 sites, 3 countries - UK, Sweden, Germany). MASSIVE was used in the UK DEVRL project as one of the distributed VR systems under test.

The DEVRL (Distributed Extensible Virtual Reality Laboratory ) project [12] was designed to allow people in different parts of the UK to be able to work together in the same virtual space to accomplish joint tasks. Three testbed applications were developed: one to allow people to navigate through 3D representations of information spaces (Nottingham); for learning through the creation of a virtual classroom (Lancaster); and an environment to allow designers to work together to create geometric shapes, for example, in car design (QMW).

DeepMatrix [13] is a virtual environment system combining two innovative technologies, VRML and the Java programming language, via the EAI (External Authoring Interface) of VRML. The EAI allows for communication between a VRML world and a Java applet within the same Web page. The system implements a client–server network architecture employing a combination of TCP and UDP standards for communication. The system provides multi-user virtual reality environments, providing embodiment through the use of personalisable avatars and communication via a chat system.

The MAESTRO (Maintenance System based on Telepresence for Remote Operators) project [14] aims at developing the use of telepresence for maintenance, installation and repair of mechanical (or electromechanical) equipment. It is particularly dedicated to the training of complex maintenance/installation scenarios for remote users, such as SMEs which cannot afford on-site complex training equipment. The resulting technology should enable users to train themselves to deal with maintenance tasks by connecting to a "Virtual Showroom" where they can learn maintenance procedures through computer-augmented video-based telepresence. Techniques used will be high-speed networks, video-based telepresence and Augmented Reality.

These projects constitute the nearest neighbours in terms of research goals. Strong similarities exist with the objectives of ECHOES. These projects describe research conducted into collaboration in virtual environments and have shown this to be a valid and achievable goal. Such environments add to the co-operative experience, providing users with the ability to interact visually with other participants. The use of user embodiments (avatars) in these worlds provides a richer communication metaphor, providing perception, localization and identification of other users.

# 3 ECHOES ARCHITECTURE

## 3.1 AN OVERVIEW

The ECHOES project differs significantly from those mentioned above in three important respects :-

- It provides a personalised and contexualised training experience;
- It seeks to strengthen the notion of the connected *virtual community* immersing the user into a cohesive team of fellow users;
- It seeks to deliver it's functionality through the adoption of a multi-agent paradigm;

An abstract view of the ECHOES Architecture is depicted in Figure 3.1. As can be seen from the

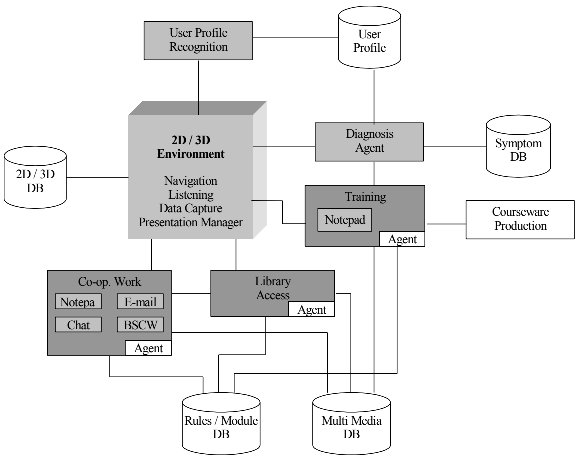



Figure 3.1: ECHOES Architecture

diagram, the major components of the system are that of the 2D/3D Virtual Training Space, User Profile Management, Diagnosis support (First Aid) and CSCW (Computer Supported Co-operative. Work), and a number of data stores.

All of these components are managed by agents. They provide a number of complementary services together with the management of the overall interaction between the user and the system, which is highlighted in figure 3.2.

The overall aim of using an agent based system is to enhance user interaction and to provide a customised training environment. The tasks that the agents perform can broadly be divided into two distinct areas; firstly customisation of the *Virtual Training Space* controlled by a User Profile Management Agent, which dynamically responds to user preferences and user actions. Secondly, support for users through CSCW and a First Aid system, whose task is to respond to detailed user queries on complex artefacts.

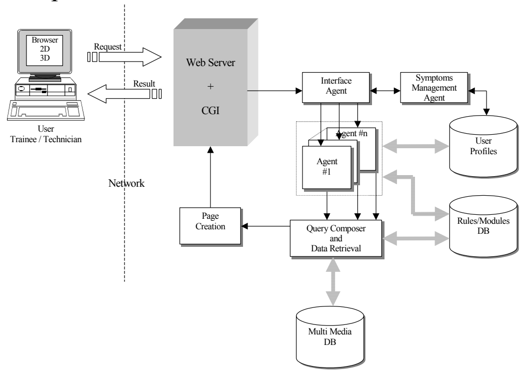



Figure 3.2: ECHOES Interaction Diagram

In the following section we outline the ECHOES agent community that deliver the machinery and functionality of the ECHOES system.

## 3.2 AGENT COMMUNITY

At a purely functional level, the system architecture can be viewed quite simply as a grouping of co-operative agents whose primary tasks are to:

- Respond to user requests & user interaction;
- Manage access to services and upkeep of the multimedia databases;
- Provide a dynamic User Profiling system.

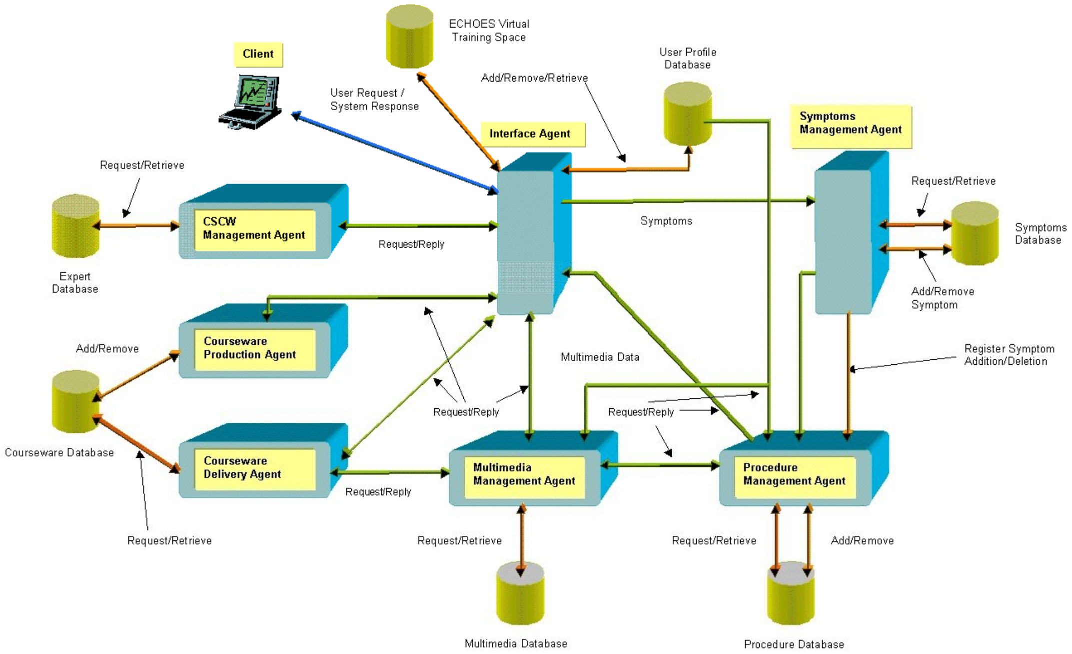



Figure 3.3: ECHOES Component Architecture

Figure 3.3 depicts a functional component view of the ECHOES architecture, and has been realised by refining the systems' tasks and assigning an agent to manage these individual tasks. We will now briefly describe the roles assigned to each of these agents.

***Interface Agent:*** manages and mediates the interactions between the system and the user in a controlled and consistent manner. Its primary tasks are to analyse user requests and forward them on to the appropriate service, to build up a User Profile and to manage and update the User Interface.

***Symptoms Management Agent:*** aids the user in the determination of a given technical fault by analysing a set of symptom descriptors entered by the field engineer. One of this components' primary tasks is to facilitate a *First Aid support service*. In this role its task will be to analyse and evaluate the responses entered by the user in an attempt to determine the technical fault. This component is also charged with maintenance and updating the symptoms database.

***Multimedia Management Agent:*** manages and maintains the multimedia database. This database will eventually contain *all* the media objects utilised within the ECHOES system, including text, VR objects, graphics, video, sound and images. Furthermore it provides a number of contexts in which to receive and navigate the data. The user may navigate the data through a number of different *view metaphors* depending on the role they play in the system, and depending on the task they are trying to accomplish. For example, a large information space may be viewed, navigated through a VR interface, where in a hyperspace distance between entities is synonymous with their cognitive binding.

***Courseware Production Agent:*** allows the trainer or a course facilitator to add their own courseware or to update/remove courseware already available on the system. This component, in close co-operation with the Interface Agent provides a number of editing and management tools to enhance the course facilitator's productivity. This agent enables the facilitator to view the current courseware in a number of different ways to enhance their management and editing tasks. At present the courseware production agent is indeed a weak agent with such functions being provided by standard courseware production tools like Toolbook II. This courseware is saved in HTML format and merely viewed within the standard browser that forms the central delivery medium of the *courseware delivery agent.*

***Courseware Delivery Agent:*** provides access to the ECHOES courseware. This courseware is set

out in textual format as a set of technical manuals. This components' task is to manage the material and to provide access to the data through a number of views, as discussed previously.
***CSCW Management Agent:*** The role of this component is to perform the task of problem routing and CSCW support. This agent is also charged with the maintenance of the expertise database, which maps a set of technical faults onto human experts.

Clearly not all of these agents are at the same stage of refinement. In the subsequent section we look in more detail at the realisation of this agent community.

# 4 THE ECHOES ARCHITECTURE: AGENT REALISATION

In this section we outline our methodology for the realisation of ECHOES agents. To this end we provide a brief description of agent systems before describing in some detail the medium and tools by which we fabricate our agents, specifically Agent Factory. By way of illustration we select one such agent namely that of the interface agent which in itself subsumes some sibling agents. In so doing the reader derives a clear understanding of the role of agents within the ECHOES system.

## 4.1 MULTI-AGENT SYSTEMS

Much research work has been commissioned on Multi-Agent Systems (MAS) and Distributed Artificial Intelligence (DAI) [4,15,16]. Specifically, competing agent architectures have been proposed in the literature. Two major architectural schools have emerged, namely those of *the reactive system school* and the *deliberative system school*.

In the delivery of computationally tractable models of deliberative reasoning, one approach that has gained wide acceptance is to represent the properties of an agent using mental attitudes such as *belief*, *desire*, and *intention*. In this terminology, an *agent* can be identified as having: a set of beliefs about its environment and about itself; a set of desires which are computational states which it wants to maintain, and a set of intentions which are computational states which the agent is trying to achieve. Multi-agent architectures that are based on these concepts are referred to as *BDI-architectures* (Belief-Desire-Intention) [4,17,18] and have recently been the subject of much theoretical research. Within ECHOES we adopt a strong notion of agenthood. We utilise Agent Factory in order to assist in their fabrication.

## 4.2 WHAT IS AGENT FACTORY?

In essence, Agent Factory is a tool that facilitates the rapid prototyping of Intelligent Agents. The Agent Factory System has been discussed more completely elsewhere in the literature [4,5,19,20].

Agent Factory is a member of the class of systems that embraces the BDI philosophy. The system offers an integrated toolset that supports the developer in the instantiation of generic *agent structures* that are subsequently utilised by a pre-packaged agent interpreter that delivers the BDI machinery. Other system tools support interface customisation and agent community visualisation. In creating an agent community three system components must be interwoven, those of *agents*, a *world* and a *scheduler*. The next section describes the high level architecture.

### 4.2.1 Schematic Functional Architecture

In order to provide the necessary functionality for the delivery of agent communities, the Agent Factory System has been divided into two key areas: the *Agent Factory Development Environment*, and the *Agent Factory Run Time Environment*.

The run-time environment provides the support necessary for the release of a completed Multi-Agent System. This environment is further sub-divided into a *Run-time Server*, and an *Agent Interpreter*. The run-time server offers two main services: access to the non-agent components of the system (a controller and some worlds), and a set of tools for viewing and interacting with the agent community. The Agent Interpreter provides the functionality necessary for the execution and visualisation of agents.

The development environment is basically an extension of the Agent Factory Run-time

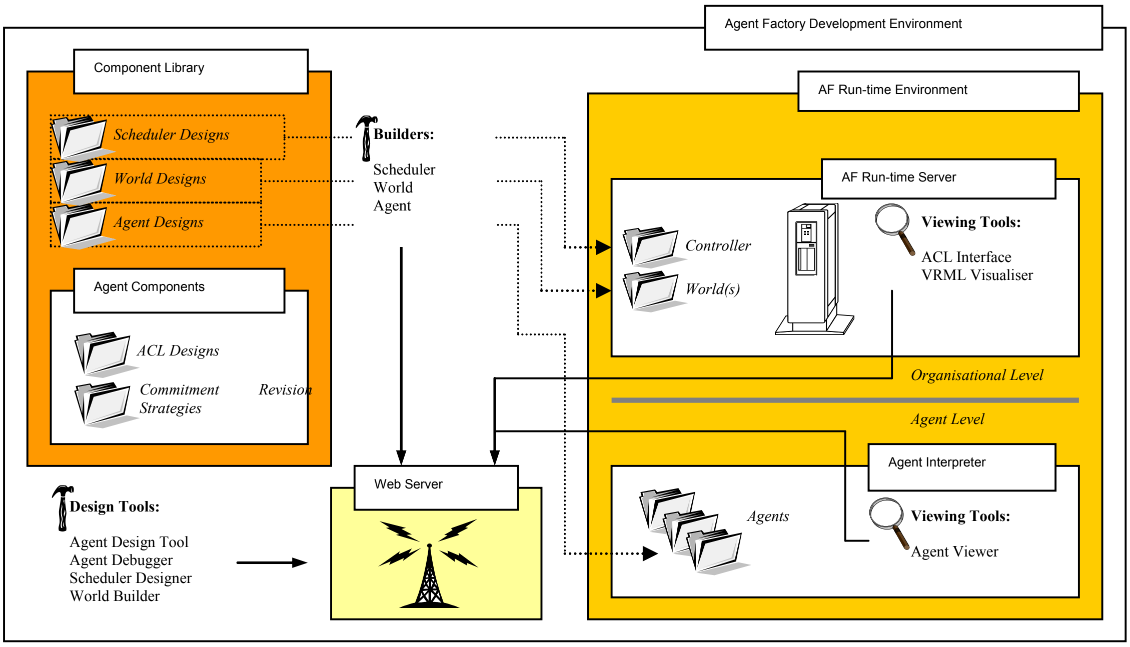



Figure 4.1: The Agent Factory Development Environment

```
<agent>                    ::= <agent-name> <belief-set> <action-set> <commitment-set>
                               <commitment-rule-set> <message-queue> <acquaintance-list>
<agent-name>               ::= <identifier>
<belief-set>               ::= {<belief-sentence>}
<action-set>               ::= {<action>}
<commitment-set>           ::= {<commitment>}
<commitment-rule-set>      ::= {<commitment-rule>}
<message-queue>            ::= {<message>}
<acquaintance-list>        ::= {<acquaintance>}
<belief-sentence>          ::= <belief> { <operator> <belief>}
<action>                   ::= <identifier><pre-condition><belief-sentence>
                               <system-call>
<commitment>               ::= COM <agent-name> <time> <action> <persistence-condition>
<commitment-rule>          ::= <belief-sentence> => <commitment>
<message>                  ::= SPEECH ACT (<sender> <recipients>) (<sa-structure>)
<acquaintance>             ::= <agent-name> <belief-set> <action-set>
<belief>                   ::= BEL <first-order structure>
<pre-condition>            ::= <belief-sentence>
<post-condition>           ::= <belief-sentence>
<persistence-condition>    ::= <belief-sentence>
<sender>                   ::= <agent-name>
<recipients>               ::= <agent-name> {;<agent-name>}
<sa-structure>             ::= <speech act verb> <when>
<speech act verb>          ::= <inform> | <request> | <commit> | <declare>
<first-order structure>    ::= <functor> [(<first-order structure>
                               {,<first-order structure>} ] )]
<operator>                 ::= & | OR | NOT | =>
```

Figure 4.2 : Agent Factory Agent EBNF

Environment. It adds a *Component Library* and a selection of tools to facilitate the rapid prototyping of agent communities. The Component Library is built from Component Design Hierarchies (CDH) that extend the standard Object Hierarchies in the OOP Paradigm. A CDH exists for each of the main system components identified as depicted in figure 4.1.

The *agent* is the base computational unit of Agent Factory, it combines a series of attributes that represent and support the Mental State model of the agent, a set of methods (the actuators), a set of perceptors, an Agent Communication Language (ACL), and a Commitment Revision Strategy. This design is then executed using a generic Agent Interpreter. Figure 4.2 provides a detailed EBNF description for this agent structure.

The *scheduler* controls execution of the community, using an algorithm that exploits parallelism where possible.

Finally, the *world interface* acts as a medium between the problem domain, the community it is being developed for, and the other components of the Agent Factory System.

Access to these environments is provided both locally through Graphical User Interfaces (GUIs) and remotely through the World Wide Web (WWW) via a purpose built *Web Server*.

In the following sections, we consider one such ECHOES agent, namely the Interface Agent and give an overview as to how this was realised.

## 4.3 The User Interface and User Profiling

As outlined earlier in this section, there are three main tasks that the ECHOES system is designed to perform. Two of these tasks are related to the Interface and interaction with the user, while the third task relates to the management and access to the various ECHOES services. As seen in Figure 3.3, it is the role of the Interface Agent to deal solely with user interaction, and to pass requests onwards to the relevant service. Figure 4.3 outlines the actions that the Interface Agent is designed to perform. Simplistically, the Interface Agent acts as a filtering mechanism – it accepts input data, potentially manipulates that data and outputs this data to all interested parties. A good analogy of the Interface Agent would be that of a Postal Clearing House. All input, either from a user, or from the ECHOES system, is analysed, and its destination is determined. Depending on the type of information and/or the information's destination, a record may be taken and stored for later use.

In describing the Interface Agent, firstly the basic functionality will be described, before then discussing in some detail the individual agents and the tasks they are to provide. Looking at Figure 4.3, it is clear that all external input from the user domain enters through the Listener Agent. This agent is responsible for passing on the user input to the requested service, and potentially the Avatar Representation Agent. The Listener Agent may also make a record of the data that it receives.

All data from the ECHOES system enters the Interface Agent by way of the Presentation Agent. This agent is charged solely with data presentation to the user.

The description of the User Interface components will be divided in terms of their function – User Interaction & User Profiling.

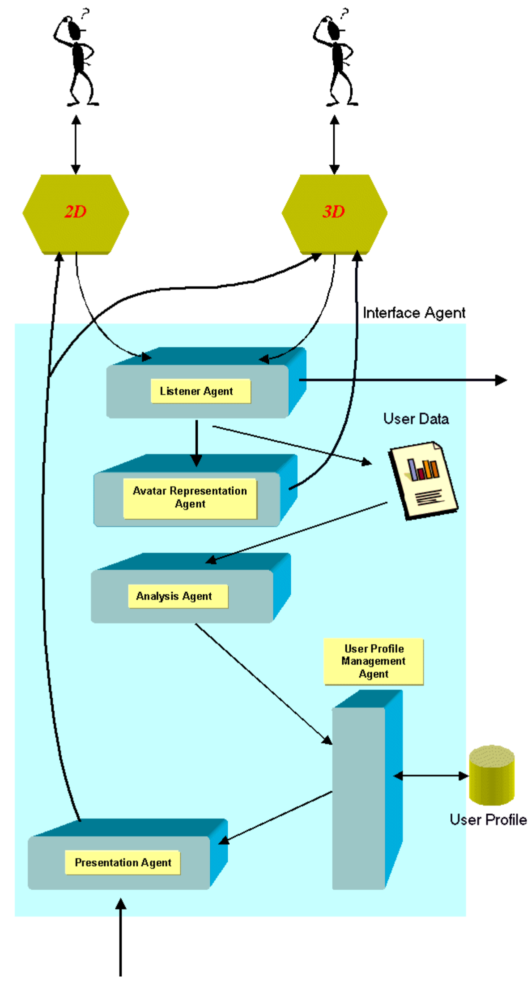



Figure 4.3: The Interface Agent

### 4.3.1 User Interaction

The primary task of the Interface Agent is to mediate interaction between the user and the ECHOES system. Figure 4.3 outlines two environments through which the user may interact with the system - a 2D environment and a 3D environment. Both of these environments provide similar services through differing media, and it is the task of the Interface Agent therefore to provide access to a rich set of services, regardless of the media environment. ECHOES is delivered through either a 2D or 3D interaction mode dependent upon user system configuration or experience or indeed preference. These two media environments will be outlined in section 4.

Three agents are collectively tasked with providing the user with a form of embodiment and accessibility to system services, namely:

- The Listener Agent;
- The Presentation Agent;
- The Avatar Representation Agent;

We view these agents as providing an outer shell that abstracts the ECHOES system design away from the external user layer.

**The Listener Agent**

The Listener Agent listens for user requests. On receipt of a request, the Listener Agent firstly determines who has sent the request, and secondly what environment they are accessing the system through (either 2D or 3D). It then determines the request type and maps it onto a system service. A package is created containing all required information, and is sent off to the appropriate system service for handling.

The Listener Agent also provides additional services relating to user profiling, and these are outlined later.

**The Presentation Agent**

The Presentation Agent provides system responses to a user independent of the environment through which they are interacting. Therefore, it is tasked with taking the input system information, containing the address of the user to whom this information pertains, and outputting it to this user.

As its name suggests, it is primarily tasked with data presentation and data formatting. This agent therefore consists of a number of routines by which it can map system responses with either 2D or 3D Interface views, and display these to the user.

**Avatar Representation Agent**

The avatar representation agent is a weak agent responsible for the creation and customisation of user avatars. In Fig. 4.4 we see the avatar selection page. Here the users may select an avatar and a face to represent them in the virtual world. The URLs of the avatar and face are then stored as part of the user profile. In addition to this it is necessary to store in the user profile at each and every instance the user's orientation and their translation in the ECHOES world.

Once a face and an avatar have been selected a combined form is then displayed along with a board which provides links to places in the ECHOES world (Fig 4.5). Of course this represents a relatively minimal approach to avatar customisation compared to other attempts at mapping images around polygonal structures and their subsequent animation [21]. Once inside the ECHOES world the user updates his location information (translation and orientation) every time he moves. This information along with the URLs for the face and body are then used to display the avatar in the world of other users.

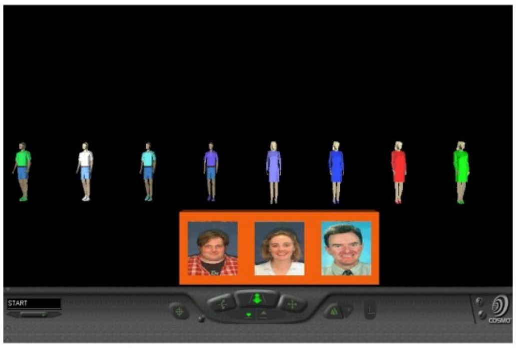


Figure 4.4: A screenshot from the Avatar selection page

The user profile is stored on the server. Clients can access this server and request information or pass information to it. The means by which this is achieved is through the use of the EAI or External Authoring Interface. The VRML browser window is controlled by an applet, on the same page, which in turn communicates with the server.

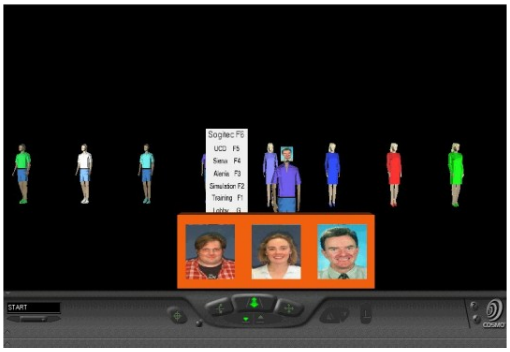


Figure 4.5: After the Avatar and Face have been selected

To monitor events in the virtual world we implement an abstract public interface called an EventOutObserver. In our case we created an object to handle a number of events called an AvatarObserver. What events should this observer handle? Well each client will have it's own copy of the virtual world. Thus it will have to keep track of where the avatars of *other* clients or users should be displayed and will have to pass on the information allowing its avatar to be displayed on other client machines. So a time sensor with an event out every second is monitored by the AvatarObserver. Thus every second the AvatarObserver will poll the server and request information on any changes in location or orientation of the other clients. If changes have occurred it will move the avatars to their new locations. The AvatarObserver will also monitor the orientation and translation of it's own avatar and when a change occurs in either of these it will instruct the server to update its user profile.

The Avatars themselves are simply just empty translation nodes. Using a CreateFromString a body and a face are added to these translation nodes. The URLs for the body and face are taken from the user profile located on the server. The Avatar can then be moved from place to place within the virtual world and this movement may be witnessed by other users. For each simultaneous user in the world an empty node is required. These are filled as users login. Thus the maximum number of simultaneous users is defined by the number of empty translation nodes declared in the virtual world (currently 10 but this could be any number).

### 4.3.2 User Profiling

The ECHOES user profile structures are pivotal in the personalisation aspects of content delivery. Several constituent agents collectively contribute to this function, namely those of:

- The Listener Agent;
- The Analysis Agent;
- The Initial Profile Capture Agent;

We consider the profiling agent as an information hiding entity that abstracts the design away from the lower level constituent components. We consider each of these briefly below.

**The Initial Profile Capture Agent**

The Initial Profile capture agent presents to each new ECHOES system user a check list by which they can express information about their general company position, their job description, expertise, training history and general preferences.

The ECHOES Initial Profile will be used as the default individual user profile. As such it presents a mechanism for bootstrapping the system personalisation. It is clear that perhaps in certain situations the user is not the best placed to provide some of this information. Where the user for example needs to assess their expertise levels they may be too self-effacing or indeed too generous in their own assessment. Patently other factors like user preferences can only really be captured in this vein. The initial profile capture agent utilises a check list approach. The static aspects of the user profile can be accrued effectively in this manner ensuring that the form filling necessitated is not unwieldy in nature.

**The Listener Agent**

The Listener Agent is responsible for gathering information about particular users behaviour within the virtual world. This information is used to supplement the static user data. It monitors sensor activation. Three key sensor types are available within VRML those of proximity, touch and time sensors. These sensors are embedded within the VRML world and their activation recognised.

Most of the data accumulated by the Listener agent is quantitative in nature. The listener agent traps user events by responding to EAI initiated events. The data captured when the user interacts with the ECHOES virtual space will typically contain details of navigation paths through the activation of proximity sensors, time spent in different VRML rooms through the activation of time sensors, activities engaged in through the activation of touch sensors.

**The Analysis Agent**

The analysis agent takes the user information accrued by the listener agent and tries to analyse the data with a view to dynamically augmenting the unique user profile for each individual user. A weighting algorithm is applied to the data in order to extract relevant features which are then used to characterise the users perceived preferences. At present this agent can analyse data derived either as a result of browsing a HTML site(s) or navigation around a VRML world. The user profile represents the addition of added value to the raw user information. For example significant time spent in lets say travel shops might enable the analysis agent to infer that the user is considering an imminent holiday.

All agents embrace the BDI structure described in section 4. By way of example, below we characterise a commitment rule that the analysis agent may contain.

```
IF BEL ( enters( training_room, userN ))  AND
    SPEECH ACT (analysis_agent,
                        courseware_del_agent,
                      NOW, inform ( field ( userN,
                                               ModuleN)))
COM courseware_del_agent,
       NOW, update ( Competence_model, userN )
```

Experience has shown that relatively crude and blunt user descriptors can result in considerable personalisation capacity.

**The ECHOES User Profile**

The ECHOES profile represents an aggregation of a static component and a dynamic component of the profile. The static component will be acquired at the time of initial system registration. Thereafter it will be supplemented with additional information accrued over time and based on user behaviour while interacting with the system.

Each profile is identified by a unique identifier and this acts as the primary key for the database. Numerous secondary keys exist including inter alia, user category and job description.

The database is comprised of a set of user profiles, each comprised of a unique name and a static and dynamic part. The former is derived from the check list approach and the latter from a combination of background monitoring and collaborative filtering

The true task is taking the raw data and converting this quantitative data into added value qualitative data. This activity will be performed by the analysis agent which will build up the user profile from this data. The Interaction history is purely quantitative, however the deriving of the interest list and the associated strength of interest is qualitative and derived from the activities list and interaction history. The activities list is inferred from sensor activation data accumulated. The training profile and associated performances will be communicated by the courseware delivery agent. The usefulness of the profile is dependant upon two key factors those of relevance and recency. Thus the ease with which it can be maintained and control over its update are of paramount importance.

# 5 ECHOES SYSTEM DEMONSTRATOR

Having presented a description of the design and delivery of the ECHOES system, we now present the reader with a glimpse of ECHOES system interaction.

The ECHOES user base includes technicians, trainees, novices and supervisors. Within the course of their work these people become familiar with environments such as libraries, training rooms, simulation rooms and conference rooms. Frequent visits to such areas would be a natural part of their jobs. ECHOES embraces this familiarity and utilises it as the basis for the interaction metaphor. A *Virtual ECHOES Training Environment* is provided where a physical metaphor seeks to harness this familiarity by ascribing system functions cohesively around a series of themed virtual rooms. Furthermore provision is made for embodiment of each system user through an avatar allowing users to observer and interact with others with whom they co-exist in the shared virtual space. To reinforce this embodiment metaphor, each user is given a hand held controller which is carried with them wherever they go. This allows the user to access key system functions: Mail, Chat, White Board and BSCW from anywhere in the ECHOES environment.

Below is a list of the key rooms provided by the ECHOES Virtual Training Space:

- Conference room
- Library
- Training room
- Simulation room
- First Aid room
- Control room

We consider each briefly.

## 5.1 CONFERENCE ROOM

Here the user can communicate and virtually meet with colleagues and collaboratively solve tasks. In the future video conferencing capabilities will be embedded into this room, as will the animation/annotation of avatars which will convey levels of participant satisfaction, alienation and interest. This could be achieved by simplistic social iconic meters.


## 5.2 LIBRARY

As in the real world the library is used to access documents. However the virtual library has a number of advantages over its physical counterpart. Multiple users can access the same document concurrently. Furthermore the library does not simply store text based documents. It can store many differing media types including text, diagrams, images, audio, video, HTML and Powerpoint.


Beyond this we can achieve the dynamic tailoring of the library to individual reader needs. Intelligent librarians may mine relevant detail and alert the user to its existence.

## 5.3 TRAINING ROOM

The training room provides an instructor with a number of different mechanisms for teaching trainees. Group training enables a number of students to sit the same course at the same time with the instructor guiding, instructing and interacting with the students. For example training videos can be shown or the White Board can be used to display illustrations to enhance a users understanding or answer a question. Thus the trainee can participate in what would be regarded as standard classroom activities.


Individual training is furnished through individual Training Booths. Here the trainee may pursue a course of individual study. This delivers a web based training course where the trainee moves at their own pace. Continuous assessment is used to establish a users current knowledge base and level of experience. This enables the system to customise an individual's future training path with respect to their requirements and abilities.

## 5.4 SIMULATION ROOM

Within this room a user may view simulations of technical artefacts. The level of detail provided and simulations the user is allowed to access will depend on the user type and level of prior experience. Ultimately, a detailed causal model could be envisaged.

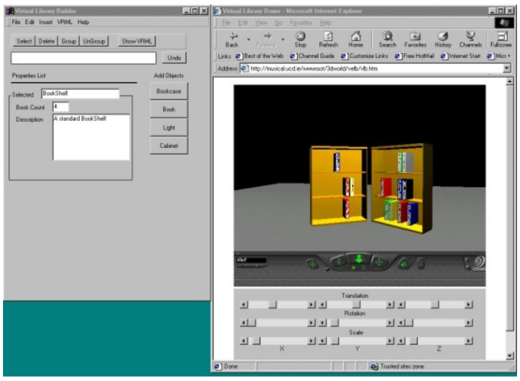

## 5.5 FIRST AID ROOM

From here technicians can register faults encountered in the field using the Fault Analysis Form. Checks are carried out to see if the fault has been previously encountered, if so detailed solutions to the problem are suggested. Alternatively the fault is registered with the system. If the problem is adequately addressed using solutions provided the priority of the fault increases.


The system may then provide a mapping between the specific problem domain and an experienced expert with whom the technician may consult.

## 5.6 CONTROL ROOM

Limited access is provided to this room. Tools are provided here that aid supervisors in their jobs. Above is a screenshot of one such tool, that of the *Virtual Library Builder*. This, as the name suggests, is used to individually tailor a virtual library environment to a specific user or domain needs.



Virtual artefacts, such as bookshelves to which books can be added are created. Differing media can be associated and bound to these books, thus creating a rich multimedia repository.

# 6 CONCLUSIONS

Within this paper, we have presented the ECHOES system. ECHOES provides a collaborative virtual training environment for multiple simultaneous users.

Users are encouraged to visit a virtual world and become a part of a virtual community. This system has been delivered through the medium of VRML and system intelligence provided through a collection of intelligent collaborative agents.

A first ECHOES prototype has been designed and implemented. Initial user trials have been conducted, with more detailed studies currently underway. Findings to date indicate after an initial acclimatisation period, user acceptance of the system is high.

# ACKNOWLEDGEMENTS

We would like to acknowledge the contribution made by our ECHOES project partners, namely Alenia Marconi Systems, University of Siena and Sogitec Industries.

# REFERENCES

[1] Pasquarelli A., de Stefani, F., O'Hare, G.M.P. Murphy, A. J., ECHOES: EduCational Hypermedia On-linE System, Proceedings of IEEE MultiMedia Conference, Firenze, Italy June 19th-21st 1999.

[2] ECHOES Web site: http://www.lti.alenia.it/ECHOES/ECHOES1.htm

[3] J.R. Vacca, VRML: Bringing Virtual Reality to the Internet, Academic Press, 1996.

[4] O'Hare, N.R. Jennings, Foundations of Distributed Artificial Intelligence, Sixth Generation Computer Series, Wiley Interscience Publishers, New York, 1996.

[5] G.M.P. O'Hare, R. Collier, J. Conlon and S. Abbas, "Agent Factory: An Environment for Constructing and Visualising Agent Communities", Pre-Proceedings of the Ninth Artificial Intelligence/Cognitive Science Conference, AICS'98, UCD, Ireland, Aug. 1998, pp. 249-261.

[6] G.M.P. O'Hare, S. Abbas, "Commitment Manipulation within Agent Factory", Proceedings of Decentralised Intelligent and Multi-Agent Systems, DIMAS '95, Cracow, Poland, 22-24 Nov. 1995.

[7] Hagsand, O. , "Interactive MultiUser VE's in the DIVE System", IEEE Multimedia Magazine, Vol 3, Number 1, 1996.

[8] Benford, S., Bowers, J., Fahlen, L., Greenhalgh, C., & Snowdon, D. User Embodiment in Collaborative Virtual Environments. CHI '95 Proceedings.

[9] Normand, V. and Tromp, J. Collaborative Virtual Environments: the COVEN Project. Proceedings of the Framework for Immersive Virtual Environments Conference, FIVE'96, December, Pisa.

[10] Schmidt, K. (ed.): Social Mechanisms of Interaction, COMIC Deliverable 3.2, Esprit Basic Research Project, Computing Department, Lancaster University, Lancaster, U.K., 1994.

[11] Greenhalgh, C. & Benford, S., MASSIVE: A Collaborative Virtual Environment for Teleconferencing. ACM Transactions on Computer-Human Interaction, Vol 2, No. 3, September 1995, Pages 239-261.

[12] Slater, M., Usoh, M., Benford, S., Snowdon, D., Brown, C., Rodden, T., Smith, G., Wilbur, S., Distributed Extensible Virtual Reality Laboratory (DEVRL), Virtual Environments and Scientific Visualisation '96, Springer Computer Science, M. Goebel, P. Slavik, J.J. van Wijk (eds.), ISSN0946-2767, p137-148.

[13] Reitmayr, G., Carroll, S., Reitemeyer, A., Wagner, M.G., DeepMatrix – An Open Technology Based Virtual Environment System.
Whitepaper on DeepMatrix System, October 30, 1998.
http://www.deepmatix.com/

[14] Yun, T.H., Kong, J.Y. and Hong, J.W., 'A CORBA-based Distributed Multimedia System', Proc. of the Fourth Pacific Workshop on Distributed Multimedia Systems, Vancouver, Canada, July, 1997, pp. 1-8.

[15] Bond, A.H., Gasser, L. eds. "Readings in Distributed Artificial Intelligence", San Mateo, CA. 1988.

[16] Durfee, E.H., Lesser, V.R., Corkhill, D.D., "Trends in co-operative distributed problem solving", IEEE: Knowl. Data Eng. 11(1), 63-8, 1989

[17] Rao, A.S. and Georgeff, M.P., "Modelling Rational Agents within a BDI Architecture", Prin. of Knowl. Rep. & Reas., San Mateo, CA., 1991

[18] Jennings, N.R. "Specification and implementation of a Belief-Desire joint intention architecture for collaborative problem solving" Int. Jour. of Intel. and Co-op. Info. Sys. Vol. II no3, 1993

[19] O'Hare, G.M.P. and Abbas, S., "Commitment Manipulation within Agent Factory", Proc. of Decent. Intel. & MAS, Cracow, Poland, 1995

[20] Collier, R. "The realisation of Agent Factory: An environment for the rapid prototyping of intelligent agents", M.Phil., Univ. of Manchester, 1996.

[21] Capin, T.K., Pandzic, I., Thalmann, N., Thalmann, D., "Realistic Avatars and Autonomous Virtual Humans in VLNET Networked Virtual Environments", IEEE Computer Society Press, October 1998.

# Why Bill Was Killed
# – understanding social interaction in virtual worlds

Mikael Jakobsson
Department of Informatics
Umeå University, SE-901 87 Umeå, Sweden
mjson@informatik.umu.se

## Abstract

This paper deals with how we should approach the sociology of virtual worlds on the Internet. I argue for the importance of establishing an inside view based on direct experiences of the phenomenon, to avoid the risks of drawing erroneous conclusions about virtual worlds based on the physical world, and not realizing that virtual worlds are full of real people engaged in real interaction. I present an incident from a world based on the Palace technology to examplify the following points: The social interaction is fundamentally different from interaction in the physical world. The interaction is real. The social structures are hierarchical. People are not anonymous.

**Keywords:** Net-life, Virtual worlds, Social interaction, Virtuality, Social structures, Identity.

## 1 Introduction

Most people have not spent any significant amount of time within a social virtual world (VW). This is probably true also for the community of researchers who are conducting research within this area. The actual participation in everyday life in a VW is often regarded as unnecessary for the work that they do. My experience from discussions about the characteristics of VWs with people who have not themselves engaged in these activities is that they tend to hold what I call an outside view of the phenomenon. This view typically includes one or more of the following positions:

- Social interaction in VWs is directly comparable to face-to-face interaction but with much lower performance.
- Interaction in VWs is not really for real.
- There are no social hierarchies in VWs.
- People are anonymous in VWs.

I have found that the inside view, i.e. a understanding of the VWs based on first hand experience, is very different from the outside view and I would like of offer the inside view as an appropriate context for understanding social interaction in VWs. In this paper I will first account for a particular incident from a VW. I will then use this case to discuss the four points listed above and conclude with a look at those points from the inside view.

Throughout the paper I have made frequent use of Erving Goffman's [3] theoretical framework for understanding the presentation of self in everyday life since I have found that it, despite being developed in the fifties, very nicely fits and helps explain my observations and findings.

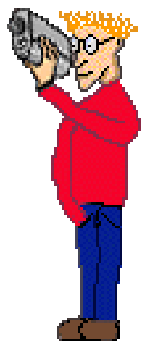


Figure 1: My avatar holding a video camera

I have conducted a participative study within a VW space called the Palace. I have mostly made observations in public spaces and saved logs of the communication and captured images from the sessions, but on some occasions I have also used video to directly record what has happened on the screen. To alert the participants that I have been recording the interaction on video I have used a special graphical representation of myself, known as avatar, holding a video camera. (See figure 1.) In line with my own reasoning I have changed the names of the people involved in the episode that I will account for shortly. One might argue that their identities already are protected by the pseudonyms that they have chosen for themselves. But as we will see, this reasoning would be quite characteristic for the outside view that I am criticizing.

The Palace is a system for making and using two-dimensional graphical worlds. In them, people are represented as small images superimposed on often cartoon-like backdrops. The citizens in these worlds belong to different classes. At the bottom of the hierarchy are the guests. They are restricted to using avatars that look like smileys and they get a new generic name every time they enter a Palace world. (See figure 2.) For a small fee you could become a member and get to decide your own name, and appearance. (Now membership registration is free.)


Figure 2: Guest and Members in the Palace

Further up on the social ladder we find the wizards. They enjoy special privileges such as being able to add, edit and delete parts of a world. They also have the power to use force against members and guest. They can take away a citizen's ability to move (pin), speak (mute) or even throw out troublemakers from the world (kill). The supreme authority of a Palace world is a god. The god is the person who runs the world server. The powers of a god are similar to those of a wizard, but with some additions. The most significant of those is the power to shut a world down, and turn it on again.

Since I control the computer that runs my own little Palace world, The Virtual MIT House (V MIT), I am also the god of that world. To help me rule this world I had a couple of wizards who are American teenagers. The third member of my staff was Bill the bartender. Bill was a bot that I had written the script for. His abilities were restricted to answering more or less intelligently to any questions that were put to him, so if I for instance asked him: "Who do you love, Bill?" He would answer: "I love money, mjson." (See figure 3.) Bill also knew how to serve virtual beer.

Figure 3: Bill the Bartender, R.I.P.

In my observations I have tried to be extra attentive to conflicts and unusual events since it is at these times that the inner workings of a society becomes most apparent. It is not until an unwritten rule is disobeyed that it reveals itself. I have also chosen an episode of conflict as a basis for my discussion in this paper hoping that it will help me mediate the inside view.

## 2 KILLING BILL

One day when I entered V MIT I found, to my astonishment, that it had been vandalized. Someone had gotten hold of the wizard password and used it to delete parts of the house and writing some rather unflattering remarks on the walls. The bar was one of the rooms that had been deleted and since the script and graphics that constituted Bill was tied to this room he was also gone. Thus the vandalism turned into a case of involuntary bot-slaughter.

I have to confess that it was not so much a need to revenge my scripted friend Bill, as a curiosity to what had happened and why it had happened that led me to initiate a little investigation into the incident. I started with a look at the server log. As one might expect the two perpetrators had not used their ordinary names but the log still gave away their Palace identities. Just as I suspected this was done by returning visitors so all I had to do was to match the IP-numbers from the break-in with the rest of the server log to find their real Palace identities. Judging from the names they were normally using, they were probably a boy and a girl. I will from here on refer to them as Bart and Lisa. I vaguely remembered Bart from a treasure hunt that my wizards had arranged in V MIT. This had been one of those occasions when I had used video to record the interaction, so I even had him on video. As for Lisa there were two frequent visitors that used that Internet service provider (ISP) and one of them was my wizard!

Save for the motive it looked like it would be an open and shut case, but there were a couple of details that did not add up. Lisa seemed to be a girl and my wizard was a boy, and I could tell by what the log had registered that Lisa seemed to be unaccustomed to the wizard commands she had been using during the break-in. But all of this could of course be just a clever trick from my wizard, so I e-mailed him to tell him what had happened and that he was my prime suspect. I figured that if he was not Lisa he should at least have some information as to who she was. I also told him that I was going to put their whole domain on the ban list to deny all users of that ISP entrance to V MIT. He answered that he was sorry about what had happened but that he knew absolutely nothing about it. Just when I felt that my investigation had reached a dead end, I got the help I needed to crack the case from a very surprising source. It was the father of both my wizard and Lisa who one day e-mailed me.

He had been reading my e-mail messages to my wizard since it actually was his e-mail address my wizard was using. And when he realized that I was going to cut off the access to V MIT for the whole family he decided it was time to "get involved" as he put it. He explained that Lisa in physical life was my wizard's eleven-year-old little sister. He made a deal with me to cyber-ground Lisa for one month and asked me in return to refrain from banning the whole family. I gladly accepted his offer. The father also gave me some information as to the reason for this incident, but Lisa had more or less just tagged along for the excitement. The brain of the operation and the key to this mystery was Bart.

Getting hold of Bart was not easy though. Whenever I saw him he discreetly left the world we were in. Once again I needed help, but this time I knew where to get it. I simply spread the word to some mutual friends that I was in the process of negotiating a multiple world banning of him with some other gods, and that his time soon was up unless he turned himself in. It did not take long before he came to see me.

Bart's explanation to his actions was that one of my wizards had promised him to become wizard in V MIT. When he found out that this had been an empty promise he wanted revenge. His plan had initially been to only destroy things that my wizard had made. But things had gotten a bit out of hand and some additional property had been damaged. He seemed absolutely terrified by the risk of getting banned in all the big Palace worlds. He said that he preferred to be grounded in physical life to getting cyber-grounded. After all, this was where he had most of his friends and where he spent most of his free time. My feelings towards Bart had up to this meeting been annoyance rather than anger, but when I met him I realized for the first time that I was not completely without blame in this affair myself. I had taken on my wizards in a rather random fashion and I had probably not bothered to be very clear about the fact that I did not have any intentions of bringing on any more. To me this wizard business had never been a big thing but to some of the people out there it had been deadly serious. I decided to go easy on Bart and only ban him from V MIT for one month. We were both very satisfied with the conclusion of the whole affair and shook hands before leaving the meeting.

## 3 NEW WAYS TO INTERACT

Let us now take a few moments to think about what we can learn from this little narrative. I would like to begin from the end by reflecting on Bart's comment that he would rather be shut out from the physical world than the Palace world. Many studies have shown that conversation over the net is very limited compared to face-to-face (FtF) communication. See e.g. Parks [4] and Chenault [1] for references to some of these studies. Much of our ability to use non-verbal cues, intonation etc is lost, and our haptic senses are rendered virtually useless. It might seems clear from this research that this medium is so limited that it is confined to offer a second rate copy of the interaction going on in the physical world. In fact some people have made that conclusion and deemed the medium to be inherently vague and shallow, one well known example being Stoll [8]. But at the same time the Barts and the Lisas are busy populating the VWs.

Judging by the popularity of worlds like the Palace, Active Worlds, blaxxun, OnLive! Traveler and Ultima Online, and the devotion of some of the people I have met in them, there must be substantial benefits of spending time there as well as good ways of reducing all the problems and limitations in terms of interaction that the medium imposes to a reasonable level. So to me there seems to be a gap between on one hand what we might expect from the outcomes of the CMC research and the conclusions drawn along that line, and on the other hand what I can see actually happening around me in my explorations of the VWs. To me this contradiction is caused by the opposing viewpoints. The outside view tries to apply the formula of FtF interaction to VWs. This is misleading in a number of ways. It is e.g. harder to tell if a person is ironic or sincere without non-verbal cues from body language and facial expressions, but the opportunity to present one self to others as a graphical image of ones own choice is clearly very compelling to many people. How well one can hide an unwanted feature is never considered in quality estimations of FtF interaction and is therefore lacking from the outside view.

But can the possibility to hide the physical body behind a digital image really be something good? Is it not deception? Yes, it certainly has an element of deceit, but so has wearing clothes and makeup for instance. Goffman [3] has shown that we constantly put considerable effort in presenting ourselves to others in a way that we hope is as favorable as possible. In doing so we will typically take on a number of different roles. One for holding presentations, one for chatting in the coffee room etc.

VWs gives us the opportunity to play yet another role, a role, which has certain properties that no FtF interaction role can have. I know from my own experiences that playing this role can relieve tension from otherwise pressing situations, and I also know from my interviews that it can help people with different disabilities to interact with other people without standing out, and feeling pitied. And not least important, it can actually be very fun to do it. Having said this, it must also be said that this particular feature of VW interaction also can cause problems. Many of these stem from difficulties in keeping the distinction between the presented self of another person, and what dwells behind that presentation. This is an effect that according to Reeves & Nass [5] is applicable to a wide range of new media.

In addition to disregarding new possibilities of the interaction, the outside view also underestimates the ability to adapt to the medium and work around the problems that it imposes. From the inside it becomes apparent that it is necessary not only to change how we do things, but also what things we do. We could try to play soccer in the Palace, but we know it would not work very well. This does not mean that soccer in VWs in general is a bad idea, only that it does not fit in with the characteristics of the Palace. On the other hand I have seen many examples of activities that are triggered by the inherent affordances of the system.

My favorite example of this is when one day a person changed her avatar name into a short sentence, so that it was shown beneath the avatar where the name usually is (see fig. 2 again). I do not remember exactly what it was, but it was something silly like: "I'm with stupid ~>". She then positioned her avatar next to her friend's avatar. The friend of course responded by typing something like: "Me too!" It was not long before the whole room was a long chain of avatars jointly creating elaborate sentences using the label intended for their names. This example appeals to me because it not only shows that VWs have unique properties that can and will be woven into the interaction, it also shows that the use of the system significantly differs from what the designer originally intended.

# 4 ARE VIRTUAL WORLDS REAL?

The most prominent ingredient of the outside view is probably the conviction that it does not really matter what goes on in a VW because, after all, it is not for real. "It's just a game." In a VW *sticks and stones can't break my bones*, but this does not mean that I not would take notice of an angry mob trying to stone me or beat me up with their sticks. My mind and my emotions are present and virtual actions can work as the cause to effects on my mental state that are as real as anything you can experience in the physical world. There are many descriptions of how real the emotional effect of things happening in VWs are. One that has been given much attention is Dibbell's [2] case concerning a virtual rape.

I can understand that it must be hard to understand how strong the emotional involvement can be in a VW setting. Picturing a person sitting in front of a computer screen seems to signal distance and detachment, and in VR terminology the situation would be characterized by a very low degree of immersion. Not many of the senses are engaged, the level of stimuli input is low, and outside stimuli have not been shut out. But from the inside, another type of immersion emerges. Consider the following extract from a paper by Richardson LeValley [6].

*I danced for my cyberspace husband, whom I had recently virtually eloped with, in-world. The dancing was a delightful and deeply moving experience. I danced with a silver teapot, with a chest, with my Asian female head and with my cyberhubby's frog head (with outstretched tongue and fly) on the back of my left hand. I placed a fern on the floor of a temple room and I danced up out of it and back into it. I danced in the silence. I danced for a long time. I was fully engaged in the floating of the dance and in the act of dancing in beauty for him.*

*The next morning, when I awoke in my primary referential context, I remembered the dancing, not only the image of the dancing but also the sensuality of the dancing. I had sensori-motor memory of the dance. I recalled the slight movement of the air on my face as I floated up and down, up and down. I remembered the funny feeling in my tummy from this movement. I remembered the feeling of my arms outstretched with objects on my hand. I remembered the silence and the way time was suspended. I remembered both the solitariness of my self expression, in this dance, as well as my deep emotional connection to my cyberhusband. And I remembered all of this in my physical waking world body.*

One might think that she has tried out some new incredible VR system with astonishing performance but actually her recollections describe an event from Worlds Away, a system fairly similar to the Palace. I would like to distinguish between on one hand the traditional concept of perceptive immersion and on the other hand the emotional immersion that Richardson LeValley describes. The emotional immersion is closely connected to engagement which arguably is best understood through experience and therefore hard to see from the outside view.

So far I have argued that the interaction in the VWs is real interaction with real emotions and real consequences, but this does not make the worlds real. I would, however, like to argue that also the inanimate objects of a VW are as real as objects in the physical world although different. Let us take one of those beers that Bill used to serve before his untimely demise, as an example. Obviously the most prominent characteristics of drinking a physical beer is absent in the VW. We will not get any less thirsty, and we will not get drunk no matter how many virtual beers we choose to guzzle down. So what is the point? Well, we all know that buying someone a beer means something more. It might serve as an invitation to a conversation, or a sign of gratitude or even friendship. It does not matter so much that we do not have to sacrifice money to buy a beer. It is still precious to have someone making the symbolic action of ordering a beer and handing it over to you in a virtual bar. The meaning of the act is conveyed regardless of how much the beer cost.

But even though the beer does not cost anything, the laws of inflation still works. If you get beers for everybody who enters Bill's bar it will lose its symbolic value. And if you take the time to design a custom made drink and offer this to someone the gesture will be more potent than offering a generic drink. Therefore it is socially advantageous in this context to be good at designing computer graphics.

The system with props that can be shared and custom made fills a function in the continuously ongoing formation and reformation of social structures. This activity will of course be found also in VWs that do not have props that the citizens can give to each other. It is a behavior that will be present wherever people meet but the functions embedded in a VW technology still influences what people will do in the world. In Worlds Away, for instance, they have a monetary system, and props can be given or bought, but not made. This makes the framework for the interaction in Worlds Away different from that in the Palace, and shapes the social interaction in the worlds in different directions.

So a beer in the Palace has different characteristics than a physical beer, but the symbolic significance is left intact. The symbolic significance is very important also in the use of physical objects. We frequently use them as equipment to try to convey a desired image of ourselves to people around us. Goffman [3] refers to objects used in this manner as sign-equipment. In this passage different beverages send different messages.

*Thus, in the crofting community studied by the writer, hosts often marked the visit of a friend by offering him a shot of hard liquor, a glass of wine, some home-made brew or a cup of tea. The higher the rank or temporary ceremonial status of the visitor, the more likely he was to receive an offering near the liquor end of the continuum. Now one problem associated with this range of sign-equipment was that some crofters could not afford to keep a bottle of hard liquor, so that wine tended to be the most indulgent gesture they could employ.* (p. 29)

Note how the limited supply is a deciding factor in the effect of the sign-equipment, just as in the Palace.

## 5 STATUS AND SOCIAL STRUCTURES

Another misconception about VWs as well as net-life in general, is that everyone is equal on the net. My experience, the inside view, contradicts this belief. According to Goffman [3] we form social relationships by adopting a coherent behavior over time, and we take on social roles by enacting the rights and duties attached to a given social status. When we strive for higher social status we also accept a stratification of the social structure and develop an admiration for those who have reached higher levels. The foundation for the social system is a consensus regarding values, and an assumption that the higher levels of the status level converges with that set of values.

So the concept of social status can be used also in VWs and it is as important in understanding a virtual society as it ever was in any physical society. What is new is that the criteria have shifted. The value set is different. In physical life things like money, work and how you look is important for how people will treat you. But those things are downplayed by the characteristics of the VWs. So the value set is replaced by a new one. Instead of money you need props, instead of a high status job, you need computer skills, and instead of looking good physically you need to look good on the screen. This means that someone like Bart suddenly has the chance of becoming someone.

I do not know very much about the real person behind the net identity "Bart" so this profile might very well not fit his situation at all, but my guess is that it fits many guys out there pretty well. Let us say he has trouble making the football team or getting a date for the school dance. Instead he tries the Palace and almost instantly realizes the opportunities this new world opens to him. If Bart's experiences were anything like this, it is no longer so hard to imagine how he felt when he first was promised to become a wizard only to later be let down on that promise.

Another example of the shift in the social hierarchy is what I experienced when I started my own world. At the time I was fairly new at the department and had been an undergraduate student not so long ago, but when I occasionally arranged some recreational activities in my Palace I became the center of attention. I could decide the rules and those who did not obey got a dose of the wrath of God (me). Compared to an ordinary department seminar things were turned upside-down. Young doctoral students who ordinarily keep a low profile on those occasions felt right at home in this environment and tended to dominate the meetings, while the senior researchers who often do much of the talking in the seminars were not even present.

The same phenomenon was apparent when I tried using the Palace in the undergraduate education. The distance between my students and me shortened considerably in this scenario. I had made an office where they could come and get their assignments and report their results and a coffee room where they could hang out and chat. Especially the coffee room worked as a status leveler. They even told dirty jokes with me present, which has never happened to me, before or since, when I have spent the coffee break together with my students in a physical coffee room. In short, the arena effects the discourse.

Traditional status structures are broken down and redefined, for better and for worse. This can in turn lead to a conflict between those who have something to gain from trying to keep the traditional structures intact and those who want a fresh start. It is important to remember that there are no absolute borders keeping the virtual world separate from the physical world. A person who is a lawyer in the physical world but a newcomer to the VW might want to try to bring the conversation into the area of occupation in the physical world in hope of transferring some status points, while a person who has attained higher in the VWs hierarchy but in the physical world is still in school, might try to change the subject of the discussion.

## 6 On Anonymity and Identity

But what if I would have wanted to really punish Bart by banning him from as much of the Palace universe as possible? Using the IP-address to identify the person would be problematic for a number of reasons. That would leave me with his chosen nickname as a signifier of his identity and since he can change that name to whatever he wishes at any time, we might conclude that it would be next to impossible to keep him out even if I would have wanted to. But there is a catch. To escape his sentence, he would have to give up his name, and by doing that he would also give up his identity.

This brings us back to the need to fit into the social context. Like everyone else in the Palace, Bart had built up a personal community of people around him. He had invested time and effort in the relations to these people and these investments resided within the connections to these people, in the form of social capital. For an extended discussion on personal communities and social capital in virtual communities see Ågren [9].

Without his identity, he would also be without the key to all the resources he had created for himself within this community. In fact, Goffman [3] notes that you simply cannot belong to a society without stability of self-presentation and Schiano & White [7] have found that there exists a social pressure to maintain a stable primary identity also in VWs. So as it turns out, I would not even have had to put his name on any ban list. I could just let it be known that he had done something that disagreed with the value set of the society, and he would see his social investment get flushed away. The lesson we should learn from this is that we are not, as the outside view leads people to think, by any means anonymous on the net. We are held responsible for our actions. All societies, virtual or physical, demand that we contribute something in order to benefit from being part of it, and to keep tabs on the contributions, there has to be identifiers, and without an identifier, an identity, there will be no payback.

## 7 Conclusions

Bart was a very committed user of the Palace worlds and had the inside view of what VW interaction is about, so  why did her behave as he did? He should have understood that he was taking a big risk and had much to lose. The answer to this question is twofold. First, I do not think that he thought he was running a risk of getting caught. Having the inside view does not imply having deeper knowledge about how Internet technology in general, and Palace server technology in particular actually works. Second, when I described the formation of a society as a convergence towards some commonly accepted ideals, I did not mention that there are always differences of opinion, and these can lead to substructures with other ideals within the society. I think that Bart belonged to a social structure where he could actually score some status points by behaving subversively.

So now we know why Bill was killed, and however insignificant my dear bartender bot's existence might have been, at least it gave us the opportunity to catch a glimpse of the inner workings of social interaction in virtual worlds. I have argued that it is easy to make erroneous assumptions about life in VWs when keeping a distance to the phenomenon. The reason for this is probably that the VWs are metaphorically problematic. It seems to be hard to intuitively understand what a VW is and how it works and easy to make unfair comparisons to the physical world.

I started the paper with mentioning some typical outside view positions. Now, I would like to offer my own alternatives based on the inside view. I would like to stress that I have used a case from a graphical VW and would like to advise anyone trying to apply my findings to net-life in general to at least proceed with caution.

- Social interaction in VWs is fundamentally different from interaction in the physical world.

Much of the interactions that occur in VWs have no counterpart in the physical world since they have been invented as a way to compensate for weaknesses in the medium, or utilizes strengths or otherwise unique characteristics of the medium.

- Interaction in VWs is real.

Watching someone engaged in interaction in a VW does in fact look like someone playing a computer game, and we all seem to share some intuitive idea about FtF interaction with other people as something important and fundamental for us humans, even if it is just chatting in the coffee room. But, imagine a game which consists of the same form of interaction as what goes on in the coffee room, and it is played with the same continuity. What is it then that makes the coffee room setting real and the game not real? I think the word real is a very unsuitable for distinguishing this difference.

- The social structures of VWs are hierarchical.

It is hard to bring symbols of social status from the physical world into VWs, so one might think that the social structures would be flat, but apparently we do not want it to be that way. We somehow always find ways to build social structures and ways to denote social status.

- People are not anonymous in VWs.

All forms of social structure demands that the participants have some form of identities to work. The names of the avatars are not the names of the people behind them, but as Schiano & White [7] puts it "pseudonymity is not anonymity".

I would like to end the paper by making the general observation that people continue to behave like people whether the arena is a VW or the physical world. But the technology that mediates our interaction has a great impact on what forms the interaction will take. This in turn seems to mean that theories about human behavior can be used in this new context, but the physical world cannot be used as the norm for comparison.

## REFERENCES


[1] B. G. Chenault. *Computer-Mediated Communication and Emotion: Developing Personal Relationships Via CMC*. 1997. http://alexia.lis.uiuc.edu/~haythorn/cmc_bgc.htm

[2] J. Dibbell. A rape in cyberspace: Or, how an evil clown, a Haitian trickster spirit, two wizards, and a cast of dozens turned a database into a society. In M. Dery (ed.) *Flame wars: The discourse of cyberculture*. Durham, NC: Duke U.P., 1994.

[3] E. Goffman. *The presentation of self in everyday life*. New York: Doubleday, 1959.

[4] M. R. Parks & K. Floyd. Making friends in cyberspace. *Journal of Communication*, 46, pages 80-97, 1996.

[5] B. Reeves & C. Nass. *The Media Equation. How People Treat Computers, Television, and New Media Like Real People and Places*. Stanford, CA: Cambridge University Press, 1996.

[6] J. Richardson LeValley. Constructing and Engaging Virtual Body-Mind in an Interactive Animated Cyberspace Community. *CyberConf Proceedings*, Oslo, Norway, 1997.

[7] D. J. Schiano & S. White. The first noble truth of cyberspace: People are people (even when they MOO). In *Proceedings of CHI 98*, Los Angeles, pp. 352-359, 1998.

[8] C. Stoll. *Silicon snake oil: Second thoughts on the information highway*. New York: Anchor Books, 1996.

[9] P. O. Ågren. Virtual community life: A disappearance to third places for social capital. In K. Braa & E. Monteiro (Eds*.). Proceedings of*

*IRIS 20 “Social Informatics”*, pp. 683-694. Oslo:
Dept. of Informatics, University of Oslo.

# Natural Interaction with Pedagogical Agents in Virtual Worlds

W. Lewis Johnson
USC Center for Advanced Research in Technology for Education (CARTE)
4676 Admiralty Way, Marina del Rey, CA 90292 USA
johnson@isi.edu, http://www.isi.edu/isd/carte/

## ABSTRACT

Conventional computer interfaces limit the ability of educational software to observe and interact with students. However, advances in virtual reality interface technology now make it possible to overcome these limitations and create software that interacts with learners as human instructors and coaches do. This presentation will describe an approach to creating animated pedagogical agents for virtual worlds. Such agents are capable of exploiting the capabilities of immersive interfaces to engage in natural pedagogical interactions with learners, either individually or in groups.

**Keywords:** Pedagogical agents, tutorial interaction, multimodal communication.

## 1 INTRODUCTION

Tutorial settings frequently involve complex mixed-initiative dialogs, both between learners and instructors and among the learners themselves. In one-on-one tutoring sessions, for example, initiative can go back and forth between the learner and the teacher, as each presents ideas and poses questions to the other. This dialog may take place in some work context, where the learner is in the process of solving some problem and the teacher is evaluating it and commenting on it. Dialogs in such situations may make frequent deictic references to the solution being constructed, and activity may alternate between working on a solution and talking about it. Nonverbal communication can be extremely important; the teacher may react to expressions of puzzlement on the part of the learner, and may try to give an impression of being concerned, caring, or authoritative. Even more possibilities for interaction may exist when groups of learners are brought together to work on problems.

These complexities and subtleties of interaction are easily lost in educational software designed for conventional text-based or WIMP (windows-icons-mouse-pointer) interfaces. The software gets no information from the learner until the learner moves the mouse or touches a key; it therefore cannot tell whether the learner is lost in thought, discussing a problem with a friend, or has walked away from the keyboard. Dialog boxes do not support multimodal communication, nor do they provide continual feedback to the learner.

This presentation will describe how the rich interaction modalities provided by virtual reality interfaces make it possible to support interaction that is closer to natural face-to-face interaction. This interaction is implemented in the form of the STEVE pedagogical agent. STEVE (Soar Training Expert for Virtual Environments) appears to the learner as a virtual human figure, capable of collaboration with learners and engaging in dialog. Communication is multimodal, making use of gaze and gestures to direct the learner's attention and give the impression of awareness and attentiveness.

## 2 INTERACTION WITH STEVE

The following is a brief example of how Steve interacts with learners. More detailed descriptions are available in the papers cited in the reference section. However, textual and even pictorial descriptions fail to give a full impression of the dynamics of interaction between Steve and learners. Readers are encouraged to view the digitized videos of Steve interaction that are viewable over the World Wide Web at http://www.isi.edu/isd/VET/vet.html.

Steve appears as an animated human figure, with head, hands, and upper body, as shown in figure 1. We have experimented with various amounts of detail in the human figure, and find this level of detail in rendering to be adequate for many tasks. Each body part has a role to play in communicating with the learner; the head and eyes indicate Steve's focus of attention, the hands and arms are used to manipulate and point to objects, and the upper body helps define the space in which Steve is currently operating in to perform the assigned task. Lack of articulated legs does not seem to be bothersome to learners, and much of the time is not noticeable within the narrow field of view of a head-mounted display in any case.

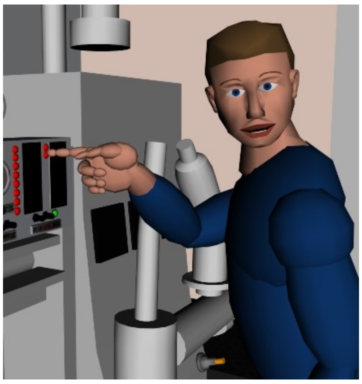


Figure 1: Steve pointing at an object

Steve engages in mixed initiative tutorial dialog with learners. A dialog might start with Steve demonstrating a task to the learner, such as operating the High Pressure Air Compressor (HPAC) shown in the figure. As Steve carries out the demonstration, he constantly monitors the location and orientation of the learner, making sure that the learner can see what Steve is doing, adjusting body position as necessary, and looking back to the learner to maintain eye contact and retain the learner's attention. At any point in the demonstration the learner can interrupt to ask questions, or take over the initiative in performing the task. When the learner has initiative, Steve watches the learner's actions, turns his head to observe the effects of those actions, and nods his head to indicate agreement with or disagreement with the learner's actions.

Team interactions are supported by Steve as well as one-on-one interactions. In the training tasks that Steve is designed to support, trainees are frequently required to perform activities in cooperation with other team members, communicating among themselves as they do so. Steve agents can assume the role of missing team members, performing the actions and communications that are required of that role, or can tutor trainees learning to perform their role within the team.

## 3 ARCHITECTURE

Steve is designed to work within the overall Virtual Environment for Training (VET) architecture developed by the USC Center for Advanced Research in Technology for Education (CARTE) in collaboration with Lockheed Martin Missiles and Spaced and the USC Behavioral Technologies Laboratory. VET is a networked virtual environment incorporating software for driving immersive displays, simulation software for modeling the states and behavior of the virtual world, speech generation and recognition software for communication between human participants and agents, and the Steve agent software itself.

Steve constantly receives data from the Vista Viewer immersive display manager concerning the learners' position and field of view. These allow Steve to track where the learner is during demonstrations, and determine whether or not the learner is able to see what is being demonstrated. These spatial data are combined with a symbolic model of the virtual world, defining the objects which must be observed and manipulated when performing the task. Thus, for example, when Steve is about to demonstrate a task step that requires turning a valve, Steve queries the world model to determine the location of the valve and of the learner, moves into a position that enables him to manipulate the valve and permits the learner to see the action, and checks to make sure that the learner can in fact see what the learner is doing. If at that point the learner is looking at something else Steve will say "Look over here!" and pause the demonstration until the learner is focused on the object being manipulated.

Steve's tutorial interactions are based on a symbolic model of the task being performed, represented as a hierarchical plan with causal links between steps. In team tasks each team role is defined, as well as the communicative speech acts that are required between members. Steve's pedagogical functions operate on the task model as needed. These functions are implemented in the Soar cognitive architecture [4], which makes possible goal-directed problem solving in a dynamic environment.

## 4 FURTHER IMPLICATIONS AND FUTURE WORK

Although Steve is designed to take advantage of the properties of immersive interfaces, some of the interaction techniques employed by Steve can be emulated with more conventional interfaces. PPP Persona [1] and Cosmo [8] make use of deictic gestures in the desktop environment. Adele (Agent for Distributed Learning Environments) [6][7] takes this idea further by glancing toward the objects on the screen that the learner is manipulating, in order to give an impression of awareness of the learner's actions.

Meanwhile, we have yet to take full advantage of the opportunities afforded by immersive displays for rich multimodal communication. Steve currently cannot recognize when a learner is pointing at an object, even though such gestures are in principle identifiable from the sensor data received from the position sensor in the learner's hand. If such deictic gestures were recognized it would be possible for support a richer variety of learner questions. It should also be possible to analyze the position and orientation data to anticipate user actions, and detect when the learner is experiencing difficulties in locating and manipulating objects.

Another area where Steve's multimodal communication capabilities are lacking is in the area of emotive gestures. Steve's face is not capable of expressing different emotions, nor is Steve's cognitive model capable of representing emotional states. This proves to be an important issue: the realism of Steve's appearance and gestures makes the lack of realism in Steve's emotional makeup all the more glaring. These issues are currently being addressed in other projects at CARTE (? Explode here?), as in projects elsewhere. Adele has a repertoire of emotive gestures such as surprise, disapproval. A new emotional model based on the work of Oatley and Johnson-Laird has been developed by Stacy Marsella, and is being used to control the behavior of characters in educational interactive stories. But in keeping with the goal of supporting natural interaction, it is not sufficient simply to create characters that can express emotion. Advances in facial feature recognition make this increasingly feasible and a potentially fruitful topic for future work.

## ACKNOWLEDGMENTS

The work described in this paper is supported by the Office of Naval Research under grant number N00014-95-C-0179 and a supporting AASERT grant, as well as by the University of Southern California, Air Force Research Laboratory, Air Force Office of Scientific Research, and the National Cancer Institute. Steve was designed Jeff Rickel and Lewis Johnson, with support from Ben Moore, Richard Angros, and Erin Shaw. Marcus Thiébaux developed the animated persona for Steve. Steve was developed in cooperation with Lockheed Martin Missiles and Space under the direction of Randy Stiles, and USC Behavioral Technology Laboratories under the direction of Allen Munro. Other members of the agent-assisted learning development team at USC CARTE include Ami Adler, Kate LaBore, Kun-Chan Lan, Stacy Marsella, Andrew Marshall, and Darren Shou.

## REFERENCES

[1] André, E., Rist, T., and Müller, J., Integrating reactive and scripted behaviors in life-like presentation agents. In K.P. Sycara and M. Wooldridge (Eds.), *Proc. of the Second Int'l Conf. on Autonomous Agents*, pp. 261-268, 1998.

[2] Johnson, W.L. and Rickel, W.L. (1998), Steve: A Pedagogical Agent for Virtual Environments, *SIGART Bulletin*, vol. 8. No. 1.

[3] Johnson, W.L., Rickel, J.W., Stiles, R., and Munro, A. (1998), Integrating Pedagogical Agents into Virtual Environments, *Presence*, vol. 7 no. 6.

[4] Laird, J.E., Newell, A., and Rosenbloom, P.S., 1987. Soar: An architecture for general intelligence. *Artificial Intelligence* 33(1), pp. 1-64.

[5] Rickel, J. and Johnson, W.L. (1999). Animated agents for procedural training in virtual reality: perception, cognition, and motor control. *Applied Artificial Intelligence Journal*, to appear.

[6] Shaw, E., Ganeshan, R., and Johnson, W.L. (1999a). Pedagogical Agents on the Web. *Proceedings of the 3rd International Conference on Autonomous Agents*, ACM Press.

[7] Shaw, E., Ganeshan, R., and Johnson, W.L. (1999b). Making a Case for Agent-Assisted Learning as a Catalyst for Medical Education Reform. *Proceedings of AIED 99*, IOS Press, Amsterdam.

*[8]* Towns, S.G., FitzGerald, P.J., and Lester, J.C., 1998. Visual emotive communication in lifelike pedagogical agents. In *Proceedings of the Fourth International Conference on Intelligent Tutoring Syst*

*[9]*

[10] *ems*, San Antonio, TX.

# Modelling Interaction to Inform Information Requirements in Virtual Environments

Kulwinder Kaur Deol, Alistair Sutcliffe and Neil Maiden
Centre for Human-Computer Interface Design, City University
Northampton Square, London EC1V 0HB
{K.Kaur| A.G.Sutcliffe| N.A.M.Maiden}@city.ac.uk

## ABSTRACT

There is a lack of understanding about the information and usability needs of users interacting with virtual environments. Therefore, interaction modelling was used to define a set of design properties required to support stages of interaction, through either information provision or basic interaction support. The design properties were evaluated in a controlled study comparing interaction success with and without their implementation. Various implementation styles were tested, such as manipulating the level of realism and the use of speech. Results showed a major reduction in interaction problems with the design properties present. Following work involves research on how to select an implementation style for the design properties and translating the properties into design guidance.

**Keywords:** Virtual Environments, Interaction Modelling, Empirical Studies, Information & Usability Requirements

## 1 INTRODUCTION

There is a lack of understanding about how users interact with virtual environments, and what information and usability requirements need to be met for successful interaction. In our previous work, we carried out studies of the design and use of VEs [5] showing that designers lacked a coherent approach to interaction design and usability, although major interaction problems existed with current VEs, such as disorientation, and difficulty finding and understanding available interactions. Similar problems have been reported elsewhere [e.g. 1,7].

Conventional interface design guidance is only partially applicable to VEs and does not cover the range of issues that arise in VE interaction. There is little usability guidance specifically for VEs and only fragmentary knowledge of some user issues, such as perception, orientation and wayfinding [2,3,9]. This paper reports work towards guidance for VE usability. Modelling of interactive behaviour was used to understand the information and usability requirements of a user. A set of required design properties for successful interaction were defined and tested in empirical studies.

## 2 MODELS OF INTERACTION

Models of interaction for VEs were developed from Norman's [8] theory of action. This theory represents interaction in cycles of seven stages: form goal, form action intention, specify action sequence, execute, perceive feedback, interpret and evaluate implications for goals. Additional stages were defined to describe a more broad range of behaviour for VEs, in particular exploration and reactive behaviour to the system. Three inter-connected models were proposed to describe major modes of interaction:

*Task action model* - describing behaviour in planning and carrying out specific actions as part of user's task or current goal/intention.

*Explore navigate model* - describing opportunistic and less goal-directed behaviour when the user explores and navigates through the environment. The user may have a target in mind or observed features may arouse interest.

*System initiative model* - describing reactive behaviour to system prompts and events, and to the system taking interaction control from the user (e.g. taking the user on a pre-set tour of the environment).

Since little previous work on understanding VE interaction existed, the models were approximate and aimed to capture the typical flow of interaction for the more basic type of VE, that is single-user systems generally modelled on real world phenomena. Figure 1 describes the explore navigate model.

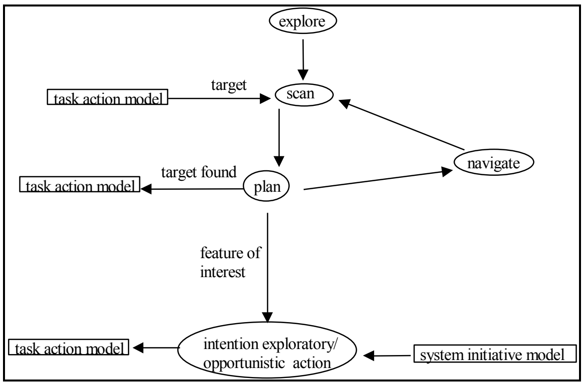


Figure 1: Explore navigate model. The user establishes an intention to explore the environment. They scan the available environment and plan what to do next. They may find a feature of interest and decide to carry out an exploratory or opportunistic action on it. Or, they may decide to explore further by navigating in a particular direction and then re-scanning. Alternatively, if they are searching for a target they will navigate in a suitable direction until the target is found, at which point they can continue planned actions on the target.

The models were evaluated in studies of user interaction behaviour, using video analysis and 'think-aloud' verbal protocols [4,6]. Results indicated that the models summarise interactive behaviour through the modes and stages of interaction, providing good coverage and general predictivity. The models were refined in light of evaluation results, to improve detailed predictions of the interaction flow from stage to stage. The resulting models provided a more accurate and representative breakdown of actual user behaviour.

## 3 USABILITY AND INFORMATION REQUIREMENTS

The models were used to identify generic properties of a VE design required for successful interaction, through systematic reasoning about contributions a VE could make in each stage in the models. For example, the property *distinguishable object* supports stages including 'scanning of the environment' and the property *locatable object* supports the 'plan next navigation movement' stage. The identified properties were organised into seven categories for different aspects of the VE, as detailed in table 1.

The generic design properties were further classified as being either for basic support, where the design provides fundamental requirements for interaction and task completion, or for the provision of information, where the design provides information useful during interaction. The information provided can be dependent on knowledge the user may have, from their real world experiences, and needs to be consistent and compatible with that knowledge. There were 17 properties for basic support and 29 for information provision. For example, to meet the property *distinguishable object* the design needs to provide basic support to the user in making objects easy to distinguish from one another. To meet the property *locatable object* the design needs to provide information to support the user in easily locating objects, which is also consistent with their prior knowledge and expectations.

| **Category** | **Requirements for** | ***Example*** |
|---|---|---|
| user task | basic support of the user task and information on task progress | *clear task state* |
| overall VE | information about the whole environment, rather than sub-section within view | *declared areas of interest* |
| spatial layout | understanding the layout of the VE and locating objects | *discernible spatial structure* |
| viewpoint and user representation | understanding the object representing the user, including the viewing angle | *clear user position/ orientation* |
| objects | investigating environment objects | *identifiable object* |
| system-initiated behaviour | perceiving and interpreting system events and ongoing system controls | *declared system control commencement* |
| actions and action feedback | finding, carrying out and assessing the success of actions | *declared available action* |

Table 1: Categories used to organise the design properties. An example is given for each category.

Use of the models of interaction meant that the identified set of properties could cover a broad range of interaction issues for VEs, including planning and task execution; exploration, navigation and wayfinding; action identification and execution; object approach and inspection; and event interpretation and response. The properties defined abstract requirements for supporting interaction that could only be applied in a practical setting by selecting techniques for implementing the properties, suitable for the application and interaction context. To evaluate the proposed properties, a controlled study was carried out comparing interaction success with and without their implementation.

## 4 STUDY METHOD

The test application was a virtual business-park, used by the Rural Wales Development Board for marketing of business units to potential leaseholders. It was a non-immersive application consisting of two worlds - an external view of the business-park and an inside view of a unit in the park. The VE included information sources, providing details about features in the units such as windows and lighting (see figure 2); actions on objects such as opening doors; and system-initiated events such as speech bubbles appearing from virtual humans and an automatic guided tour through the park. This original version of the application was assessed for the presence of the design properties. Each relevant property for the main elements in the test VE (e.g. the main objects, actions, system behaviours etc.) was judged to be supported or not supported, using detailed definitions.

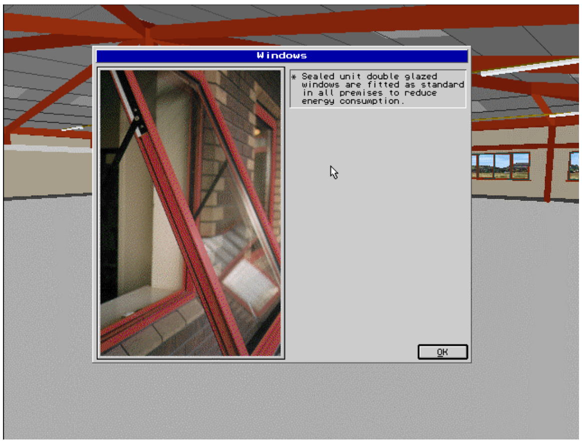



Figure 2: The inside view of a unit in the business park. From clicking on one of the windows, an information box is displayed describing the windows.

A second 'amended' version of the application was then created by implementing missing properties. Various implementation techniques were used to incorporate the design properties, such as:

- Manipulating levels of detail and realism. For example, objects such as walls sharing an edge were made more distinguishable by using textures to emphasise edges (*distinguishable object*, figure 3). A water tank object was made easier to notice and recognise by increasing detail and realism (*distinguishable object* and *identifiable object*, figure 4).
- Highlighting techniques for attentional design. For example, general amendments were made to highlight available actions for the property *declared available action*, using outlining and salient colours (figure 5).
- Additional tools or interface components. For example, information signs were used to indicate available information for the property *declared available action* (figure 6). Areas that the user could not navigate into were marked using 'no-entry' signs, which appeared on approach (*clear navigation pathways*, figure 7).
- Speech tracks providing information about the VE. For example, missing properties were implemented for an automated drive-through system control by adding a speech track (see figure 8), including required information about the start and end of the drive (for property *declared system control commencement/ termination*), its purpose (*clear system control purpose*) and the user actions available during the drive (*declared available actions during control*).
- Textual explanations, labels and instructions. For example, exit doors were made easier to identify by labeling them (*identifiable object*, figure 7).
- Constraining techniques. For example, navigation was made easier to execute (*executable action*) by adding collision detection on all walls, so the user could not accidentally fall through walls, and limiting the allowable distance from walls, so the user could not stand right up against a wall (figure 9).
- Active support techniques. For example, users were encouraged to approach an agent, who provided information about the VE, by having the agent beckon when in view (*clear object type/significance*). A handle on a drawing board was made easier to use by increasing its size and by automatically moving it to an open/closed position when clicked, instead of leaving the user to drag the handle (*executable action*, figure 5).

The techniques most appropriately fitting the context of the test application (e.g. its required level of realism) were chosen to implement properties. All amendments were examined by an independent judge, to check they represented only the requirements of the properties in question. Thirty of the forty-six properties were implemented at least once. The remaining properties could not be tested because of the lack of opportunities in the test application, or due to experimental and technical constraints.

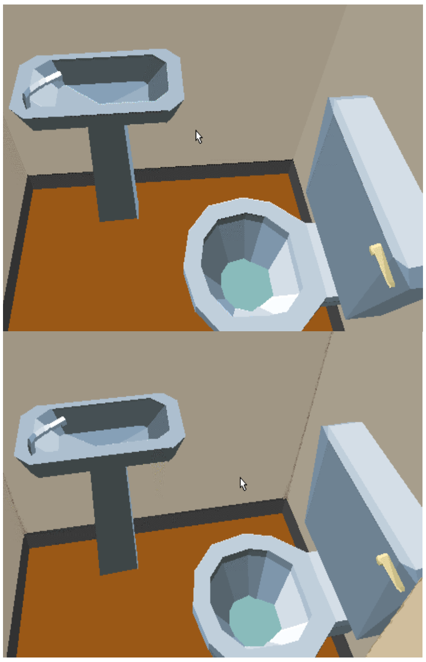


Figure 3: The original environment (top) and the amended version (bottom) where edges between toilet walls are emphasised by implementing the property *distinguishable object*.

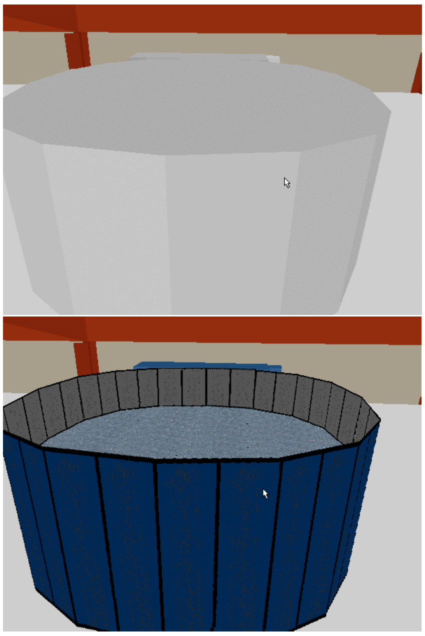


Figure 4: Implementation of GDPs *distinguishable object* and *identifiable object* for the water tank.

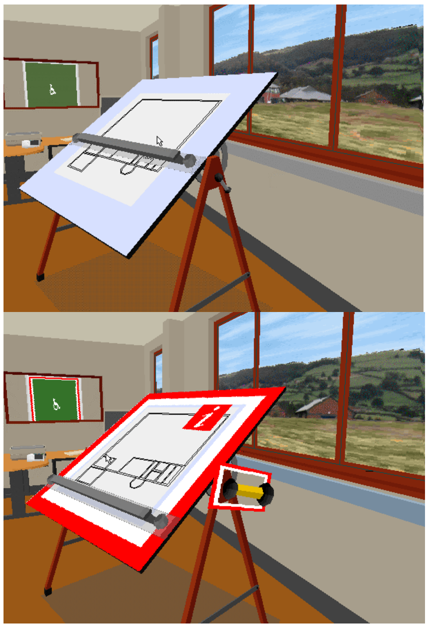


Figure 5: Implementation of GDPs *declared available action* by outlining (in red) actions on the drawing board, and *executable action* by making the handle larger and easier to move.

Figure 6: Implementation of GDP *declared available action* by clearly indicating actions for getting information about basic facilities (e.g. lighting, windows and mains boxes) through the use of '*i*' signs.

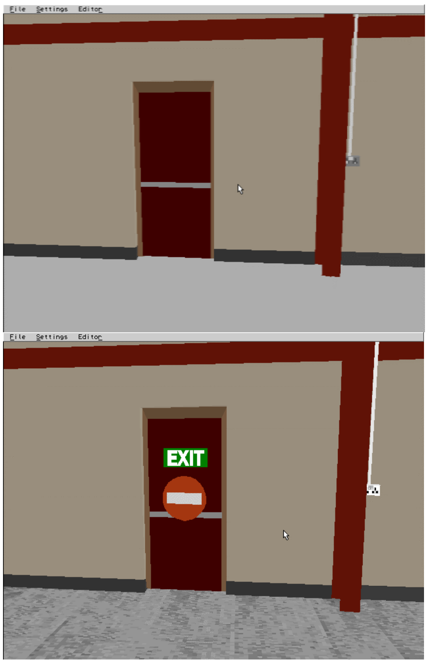



Figure 7: Implementation of properties *identifiable object* and *clear navigation pathways* for the exit door.


Figure 8: The beginning of the speech track added to the automated drive-through to implement properties for system behaviours.

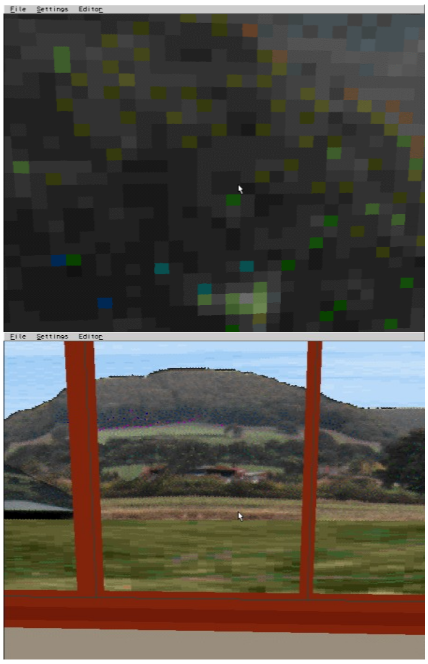


Figure 9: Implementation of property *executable action* for navigation - minimum distance from walls is limited, so the user cannot stand right up against a wall (allowable distance from wall/window in both versions is shown).

Eighteen subjects (12 male and 6 female) took part and were balanced, according to experience (e.g. VE experience), into a Control (C) and Experimental (E) group. The control group was given the original version of the virtual business-park application and the experimental group was given the amended version, with its missing design properties implemented. Both versions were run on a PC with a 21inch monitor; a joystick was used for navigation and a standard 2D mouse for interacting with objects. Subjects were given nine tasks, testing different interaction scenarios. For example, subjects were given time for free exploration, and were asked to find and investigate a water tank object, open a loading bay and compare three toilets in the unit. When subjects completed the tasks, which typically took 45 minutes, they completed a paper (memory) test on the business-park and a retrospective questionnaire on their views about the VE.

Data on usability problem incidents was gathered from observation and concurrent 'think-aloud' verbal protocols. Criteria were set for successful completion of each task, and a common scoring scheme used for the paper tests. To test the impact of implementation of the properties on interaction success, it was hypothesised that group E would 1) encounter fewer usability problems; 2) complete more tasks successfully; 3) complete tasks faster, and 4) gain more useful information from interaction (i.e. achieve significantly higher test scores).

## 5 RESULTS

There was a general improvement in interaction with use of the amended version (*all the following statistics used unpaired one-tailed t-tests, unless otherwise indicated.*):

1. The experimental group encountered significantly fewer usability problem incidents overall ($p<0.01$; avg. C=134, E=45 problem incidents per subject).
2. The experimental group successfully completed significantly more tasks ($p<0.01$; avg. C=7, E=8.4 tasks).
3. The experimental group spent less time on the tasks, but this difference was not significant ($p=0.13$; avg. C=42, E=39 minutes)
4. The experimental group achieved higher total scores for the paper test and this difference approached significance ($p=0.064$; avg. C=46, E=52%).

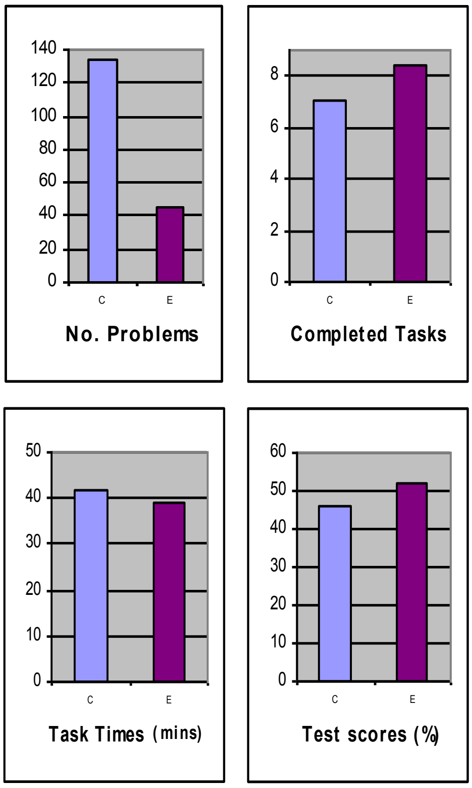



Figure 10: Comparison of the average number of usability problems, tasks completed, task times and test scores between the Control and Experimental groups.

Figure 10 shows comparisons between the groups for these four aspects of interaction. The experimental group showed some more positive views about the VE through the retrospective questionnaire. For example, they perceived the VE to be significantly better at providing information about objects ($p<0.01$; avg. C=2.9, E=4.7 on a scale of 1[low] to 7 [high]) and actions ($p<0.01$; avg. C=2.1, E=3.8). Interestingly, group E indicated weaker feelings of 'presence' inside the VE, in the retrospective questionnaire ($p=0.012$; avg. C=4.9, E=3.2 - two-tailed t-test). The reason for this is unclear but may have been due to group E having fewer major problems, which could have the effect of increasing absorption, or finding the VE less realistic because of the added highlighting effects.

Fewer problems were encountered for all of the elements of the VE which had been addressed by implementing missing properties, and in most of these cases (79%) this difference was significant ($p<0.05$). One of the greatest improvements was brought about by adding the information signs to highlight information actions, which meant that the previously hidden available information could be easily found, see figure 6. Improvements were found with all of the implementation styles used. For example:

- Manipulating the level of detail to implement *distinguishable object* (by emphasising edges between walls in the toilets, see figure 3) led to a significant reduction in the problem 'difficulty finding the entrance area to a toilet to manoeuvre out of the toilet'.
- Highlighting techniques to implement *declared available action* (by outlining active areas of the drawing board, see figure 5) led to a significant reduction in the problem 'repeatedly clicking inactive parts of the drawing board when trying to tilt it'.
- Additional components to implement *clear navigation pathways* (by using no-entry signs where access was not allowed, see figure 7) led to a significant reduction in the problem 'trying to go through the inaccessible doors'.
- Speech tracks to implement properties for system behaviour, such as *declared system control commencement* (for the automated drive through, see figure 8) led to a significant reduction in incidents of the problem 'user wrongly thinks they are controlling the movement during the automated drive-through'.
- Textual labels to implement *identifiable object* (by labeling exit doors, see figure 7) led to a significant reduction in the problem 'unsure what the back exit is and whether it is a door'.

- Constraining techniques to implement *executable action* for navigation (by setting more sensible minimum allowable distances from surfaces, see figure 9) led to a significant reduction in the problem ‘bumping into walls - nose against the wall’.
- Active support techniques to implement *clear object type/significance* (by having agents beckon to the user) led to significantly more users approaching an agent in the VE and gaining useful information from it.

The implications of these results and following work are discussed in the next section.

## 6 DISCUSSION

We are encouraged by the study results which show general improvements in interaction across all levels. There was a 66% reduction in usability problems, leading to subjects being able to complete their tasks better, gain more useful information during interaction and leading to more favorable opinions about the VE. Task time was not improved significantly, but this may have been because users tended to occasionally explore unrelated aspects of the environment. Overall results indicated that the proposed properties covered many important requirements for successful interaction, and that a VE interface could be significantly improved by implementing some of these missing properties. Various implementation techniques were put forward and detailed results provided support for these, as well as individual support for several properties.

Previous work had provided fragmentary knowledge about interaction requirements in VEs. The design properties provide a more broad and comprehensive set of requirements that are based on a theoretical model of interaction. They focus on providing basic support and the necessary information during interaction, and are at a level of granularity suitable for translation to specific design features.

The use of interaction modelling has been a fruitful approach to informing information and usability requirements. Our following work includes more evaluation studies to target untested parts of the theory and try new implementation styles. We are also refining the property set in light of detailed results, for example by including properties for uncovered problem areas. Work has started work on exploiting the theory by translating required design properties into concrete design guidelines, presented through a hypertext design advice tool. For example, the following guideline was defined from the *accessible object* property: ‘Objects should be easy to access, that is, it should be easy for the user to approach objects and take up a suitable position close to objects’. A prototype of the tool has been evaluated by industrial VE designers and this can be found at: www.soi.city.ac.uk/homes/dj524/demtool/frame.htm

Future work involves detailed research on how to select an implementation technique for information provider and basic support GDPs, and the trade-offs between different implementation styles. For example, active support techniques may be intrusive and it may be preferable to use natural affordances from the real world. Speech and language is a useful technique for information provision but may not be suited to all applications and alternative methods of providing information through more subtle forms may be needed. Constraints, such as the level of realism required and the modalities available, will affect the choice of appropriate technique, and other issues will be important, including the directing of attention and integration of information across modalities. Finally, having addressed usability for basic VEs, we would like to extend the theory to cover a wider range of interaction issues, in particular communicating and collaborating with agents or other users in a multi-user VE, and metaphor understanding and transfer for artificial VEs.

## ACKNOWLEDGEMENTS

We thank VR Solutions and The Rural Wales Development Board for loan of the test application.

## REFERENCES

[1] COVEN, D3.3: Usage evaluation of the initial applications. *Public Deliverable report of COVEN (Collaborative Virtual Environments)* ACTS Project N.AC040, 1997.

[2] Darken R.P. and Sibert J.L. Wayfinding strategies and behaviours in large virtual worlds. In: *Human Factors in Computing Systems: CHI '96 conference, Vancouver 1996. Proceedings.* New York: ACM, pp. 142-149. 1996.

[3] Drucker S.M. and Zeltzer D. Intelligent camera control in a virtual environment. In: *Graphics Interface ’94 conference, Banff, Alberta, 1994. Proceedings.* Toronto: Canadian Information Processing Society, pp. 190-199, 1994.

[4] Kaur K. Designing virtual environments for usability. PhD. thesis: Centre for Human-Computer Interface Design, City University, London, 1998.

[5] Kaur K., Maiden N. and Sutcliffe A.., Design practice and usability problems with virtual environments. In: *Virtual Reality World '96 conference, Stuttgart, Proceedings* (IDG Conferences), 1996.

[6] Kaur K., Maiden N. and Sutcliffe A.., Interacting with virtual environments: an evaluation of a model of interaction. Accepted for publication in *Interacting with Computers: VR special issue,* inpress.

[7] Miller L.D. In the context of the European project 'Metrics for Usability Standards in Computing' (MUSiC): A usability evaluation of the Rolls-Royce virtual reality for aero engine maintenance system. Masters thesis: University College London, 1994.

[8] Norman D.A., The psychology of everyday things. (*New York, Basic Books Inc*.), 1998

[9] Wann J. and Mon-Williams M., What does virtual reality NEED?: human factors issues in the design of three-dimensional computer environments. *International Journal of Human-Computer Studies* 44: 829-847, 1996.

# Virtual Campus Project - A Framework for a 3D Multimedia Educational Environment

Rainer Kuhn and Sigrún Guðjónsdóttir
University of Karlsruhe
Faculty of Informatics
Am Fasanengarten 5
D-76131 Karlsruhe, Germany
{rkuhn|sigrun}@ira.uka.de

## ABSTRACT

Virtual Universities at present generally focus on formal and functional aspects. The Virtual Campus Project has the intention to consider social facets of the student's life as well. The main goal of this experimental project is to create an educational communication and collaboration environment for the „ViKar"-Distance Learning University Project. This environment should not only provide all needs and functionality for the students but also serve as some kind of „social interface". Our actual approach is less technical but more conceptual as well as regarding content and design. Besides that we're doing research in exploring the surplus values of three-dimensional multi-user virtual environments. For the creation of a 3D-VR-environment we propose a new definition of spatial structures, a special imagery and features for orientation, communication and collaboration. Finally we present our current prototype as an experimental platform to realize these proposals.

**Keywords:** Virtual Campus, Virtual University, Virtual Environments, Distance Learning, Educational Applications, Virtual Reality (VR), VRML, Multi-User, Shared Virtual Worlds, Collaborative Virtual Environments (CVE), ViKar

## 1 INTRODUCTION / BACKGROUND

The Virtual Campus Project is a part of the „ViKar"-Project in Karlsruhe, Germany, which intends to promote the developments in distance learning research [12]. „ViKar" is the „Virtual Association of Colleges of Karlsruhe". Besides the 6 institutions of higher education in the Karlsruhe region the Center for Arts and Media Technology (ZKM) is a partner of this project. It aims at

- building a virtual college with a high degree of social interactivity
- developing multimedia based teaching modules which can be adapted to different needs by reusing and reassembling parts of them
- developing appropriate didactical methods
- the extension of the available educational supply of the six institutions of higher education as well as the extension of the supply of advanced vocational training in the Karlsruhe region of technology (in collaboration with companies)
- providing the necessary technical infrastructure in the Karlsruhe area.

In addition to the Virtual Campus Project the main development trends of „ViKar" are the development of a text and multimedia based learning environment, the creation of modularized learning content and the further development of an existing learn-server, which provides students with all necessary scripts and materials. Finally all components are going to be integrated in the superior infrastructural layer for the end-user, the virtual campus.

## 2 THE SIGNIFICANCE OF A VIRTUAL CAMPUS

Distance learning universities at present generally focus on formal and functional aspects of teaching and learning. The usual way to present a virtual „campus" is a two-dimensional webpage with an image map or a collection of links to all available functions.

With the Virtual Campus Project we intend to go a step further: Experiences gained from running a large university department and local preliminary inquiries into students' needs point out a strong demand of social interactivity and the importance of psychological aspects like well-being and identification. Therefore it's not enough to regard the formal and functional aspects while creating a virtual university, the social and emotional aspects are extremely important and have to be considered also. Accordingly the basic goal of the Virtual Campus Project is not only to provide all necessary functionality, but also to integrate the students' social needs.

The very first idea for the virtual campus of the „ViKar"-project was a sketch showing a central communication area surrounded by elements like library, learn server, student's workplace, administration etc. This figure already points out the importance of communication.


Figure 1: First virtual campus sketch

Communication and also collaboration are the basis of information exchange, team work and social life. A virtual university should offer more possibilities than text-based chatrooms and newsgroups. From this starting point we created the idea of the virtual campus as a kind of „social interface". Besides the central needs of communication and collaboration, which are main subjects of the project, the virtual campus should enable the students' identification with the virtual university like a real campus does in built reality. Furthermore the realization of the virtual campus has to overcome anonymity and accordingly has to create a sense of community and solidarity.

## 3 APPROACH

Resulting from the above mentioned demands the need of a campus concept different from usual concepts like pure image maps or link collections is evident. A corresponding concept for the virtual campus basic structure could be a web-based three-dimensional virtual reality multi-user environment, combined with multimedia content and two-dimensional information where it is appropriate. Such an environment is highly visual and spatial. It could add a new dimension to social interaction on the internet, and it is certainly a forerunner of the multimedia experiences to come. [8]

For this reason we built an experimental VRML test platform for work in different research fields and for continuos integration of new features made possible by technical progress. The near future vision is to have a multi-user communication and collaboration environment which enables audio communication and shared objects.

At the actual state of the project our approach is less technical but more conceptual as well as regarding content and design. The current work concerns the following issues:

- exploration of possibilities and surplus values of virtual reality combined with multi-user technologies
- spatial structures and formative design
- orientation and navigation
- communication and collaboration
- content for an educational environment.

### 3.1 ASPECTS OF VIRTUAL REALITY AND VR MULTI-USER ENVIRONMENTS

One of the primary goals of the Virtual Campus Project is the research into the potential and the surplus values of three-dimensional virtual reality and spatial multi-user environments. After intensive investigations and tests in this area we found the following aspects very useful and interesting for the realization of the project:

- VR-environments allow free navigation in 3D environments as known from physical reality.
- The user can interact with objects and other persons in real-time.
- Several senses are reached at the same time; while interacting in different ways the overall context is preserved.
- Multi-user systems provide additional possibilities of communication and collaboration (more about this aspect see below).
- A social framework can be created by establishing a virtual community and by providing a supply of social interaction possibilities.
- A better understanding of complex structures, connections and processes can be gained by three-dimensional visualization and interactive models.
- VR offers new ways of data visualization and search for information: spatial arrangement of information instead of linear order, the possibility of non-selective search, navigation through information etc.

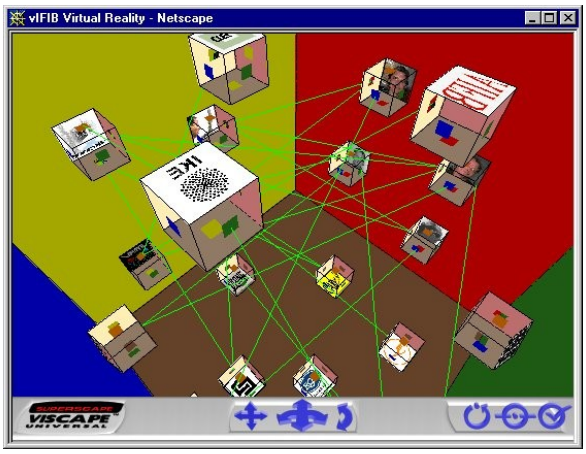



Figure 2: VR based information system
in the former project „Virtual Institute“ [5]

Nevertheless, building a VR multi-user campus does not necessarily mean to have a three-dimensional graphical user interface only. Rather the selective and appropriate use of different media, a combination of a three-dimensional multi-user environment with multimedial and text-based content is reasonable to establish a complex and useful educational environment.

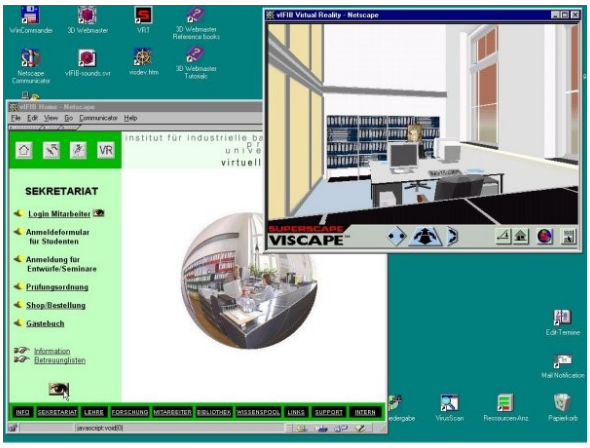



*Figure 3:* Combination of VR and hypermedia
in the former project „Virtual Institute“ [5]

## 3.2 Spatial definition and formative design in virtual space

Space, shape and design get a new meaning in virtual worlds of cyberspace. The meaning is quite different from the experiences in the physical reality. This results from both the new possibilities of virtual reality and the conditions for the creation of virtual worlds.
In this context information, communication and collaboration are the basic elements of distributed multi-user worlds. To meet their requirements, new concepts, structures and metaphors have to be developed. These should allow an extensive abstraction on the one hand and a sufficient appeal to the users' capability of association to enable orientation and navigation on the other hand.

In virtual environments objects and buildings, which are reproduced close to reality, could be useful for representation and for orientation in advance, e.g. for students who want to visit the real components of their distance learning university. But generally, for pure virtual environments there's no need to have conventional buildings with long corridors, stairways etc. just to reach an information, a functionality or a place of interest.

For the creation of the virtual campus we propose a new definition of spatial structures and a special imagery, away from reproducing reality in cyberspace. The spatial structures and the imagery of our environment are reduced to functional needs, but they should also help the user to find his way around in the environment. The definition of space and the spatial demarcation are not performed as walls or fences, but by means of linear structures, lines, panes, thin wrappers or even by applications, e.g. by aural perception of a sound when entering a special area.

The metaphors for buildings and areas are differentiated between „open“ and „closed“ structures. The extensive use of open structures and „transparent“ boundaries gives the user a spacious horizon, which is helpful for enabling a good orientation and maintaining the overall context.

Moreover we propose the use of „object classes”, comparable to classes used in object orientated programming. These classes can be used for objects with similar base functions, differentiated in attributes like appearance, shaping, colors or design details. The use of such classes enables both the recognition of functionality and the differentiation of landmarks for orientation.

## 3.3 ORIENTATION AND NAVIGATION

Orientation and navigation are central issues in virtual worlds. In complex worlds available it is often very difficult for the user to find his way around and to know where he is located. Accordingly new ideas and features have to be developed to support the user's orientation and navigation. Besides the orientation related aspects of structure and design mentioned in chapter 3.2 we propose a few more features in this area.

One important aspect is the ensuring of a spacious horizon, corresponding to the human visual perception in physical reality. This not only a question of the appearance of objects, it also depends on the appearance of the VR-browser frame as part of the web-browser window. The frame containing the virtual world should be dimensioned about three times wider than high to avoid a restriction of the horizon and to replace the horizontal eye movement in reality.

To avoid getting lost in complex virtual environments the user needs more information about the site and his position than he could see in the limited horizon of the VR-browser. We propose an additional two-dimensional sitemap, showing a ground plan and displaying the users position at runtime, as a very helpful tool to maintain an overview. This sitemap can be used not only as an orientation tool, but also as a graphical navigation interface. By applying clickable symbols for the main landmarks of the environment the symbols can be used for navigation similar to a viewpoint list. In this way the user can navigate quickly and directly to points of interest without searching in the virtual environment.

Moreover the navigation interfaces of usual VRML-browsers, e.g. the CosmoPlayer dashboard, that we currently use for our test environment, seem not to be satisfying. One of the general advantages of virtual reality should be the intuitive navigation through three-dimensional environments also for inexperienced users, but most available interfaces do not meet these requirements. Accordingly we have planned the development of a different, more intuitive interface for a later phase of the project.

## 3.4 COMMUNICATION, COLLABORATION AND SOCIAL INTERACTION

A well working and functional distributed communication and collaboration environment - which is not given by providing only IRC (Internet Relay Chat) and newsgroups - is essential for a virtual university. Therefore there's a strong need for concepts in functionality and content to make such an environment possible and to fill it with life.

An important advantage of virtual 3D environments is the integration of visual perception. This should be used to offer the additional possibilities of mimic and gestures to communication. Using these options is necessary to underline emotional expression and helps overcoming anonymity.

As a result of case studies in some 3D multi-user environments found on the internet it turned out that text-based chat is not very functional in these environments. Although while communicating in 3D-environments there were stronger emotional reactions of the users than in two-dimensional IRC, it was evident that the users couldn't take much advantage of the visual component of the 3D-environment. The users had to bundle most of their attention on typing chat text and regarding the conversation in the chat window. This points out the need of speech communication using multi-way audio. To overcome confusion by multiple participants using audio communication, there are several approaches. One idea is to limit audio to a pre-defined circuit around the user. This means that one could hear or talk to others only if they are situated within a certain distance. The better alternative to be developed would be an audio communication system which can locate the position of users and translate this in different levels of volume. So the user should hear others standing near to him louder than the ones more far way. Anyway, a minimum of possible confusion will resist like it does in physical reality.

But a chat-room in a 3D environment is not enough to make a world a useful multi-user environment, independent of the way of communicating: The environment has to offer different communication areas with different terms of use. Apart from having a public chat area there have to be forums which are related to a location or a subject as well as predefined meeting points. Besides providing different areas it's important to offer different levels of communication. Additional to a central communication area for public chat, where people can meet spontaneously and casually, there should be possibilities to limit the number of participants by spatial definition of a chat area or by inviting only selected persons to a conversation. Even single-user areas in a multi-user environment are reasonable for explicit applications, e.g. for information research and learning.

For collaboration reasons it is important to have shared persistent objects. The users need to have the possibility to modify the environment or single objects, other users should be able to see these modifications synchronized in real-time. Furthermore the changes have to be persistent, so anybody could notice them after re-entering the world at a later time, even if all former users have left the world.

Finally, the above mentioned issues of communication and collaboration enable a lot of possibilities for content and social interaction. For the Virtual Campus Project among other things we think of

- a central communication area and a network of connecting routes for casual meetings, also while walking from one component to another
- virtual lectures, seminars, presentations, discussions and other collaborative virtual events
- single-user areas for the personal workplace and learning environment
- exhibition areas for non selective information and communication
- a student's house for collaborative learning and exercising
- shared whiteboards, slide projectors, video screens
- three-dimensional demonstration objects
- portable objects a student can take to his personal workplace
- generation of temporal working rooms with individual parameters

This means, besides functional aspects we want to offer different components that enable social life at the virtual campus. When realizing the campus as a kind of „social interface", it will be unavoidable to have some means of social control, for instance behavior rules, an „ignore"-option for the user, if he feels disturbed by another user, or a „kick-off" option as a final means for the administrator.

# 4 CURRENT PROTOTYPE

The Virtual Campus Prototype, our three-dimensional VRML test environment, combines spatial structures with collaborative content, multimedial elements and two-dimensional webpages. Its current implementation is a combination of VRML 97, JavaScript 1.2 and HTML 4. The environment has been developed for the CosmoPlayer 2.1 plugin running in Netscape 4.5 on PC platforms.

The actual virtual world is developed for the use with a multi-user system, but multi-user functionality is not implemented yet. A test implementation is planned after some more conceptual work. In our current prototype we tried both to implement some of the above mentioned proposals and to indicate several ideas to be examined or realized in our future work.

## 4.1 WINDOW PARTITIONING

The graphical interface of our prototype is a window divided into three frames, each frame having different characteristics.


Figure 4: Frame structure and Pyramid's Place

The navigation-frame at the upper left of the window is the user's guide to the world. To overcome the problems of linear navigation in the virtual world a sitemap is used. The sitemap contains a structured overview of the entire world, all landmarks of the world are directly accessible by clicking their symbols. Additionally the user can always find out his current position within the world by using the appropriate button. Three more buttons serve for reloading the site, getting text-based help and starting an Internet Relay Chat, which is reachable from the whole campus world.

The virtual world frame at the upper right contains the 3D-campus and all connected worlds. The world is realized without any default routes in order to enable selective navigation. The aim is to bring the user to the desired information in an enjoyable manner and to give him the possibility to meet other people. As a result we hope that the user becomes more interested in the educational environment and accordingly will increase the number of revisits.

The information-frame is located at the bottom of the window and contains mainly text-based information or forms, but multimedial content or other virtual objects can be displayed here as well. Generally, when the user approaches a component in the campus world a new page is loaded in this frame.

This page contains detailed information on functionality depending on the appropriate location.

## 4.2 Basic structure of the virtual campus

The basic shape of the virtual campus, a circle with six surrounding elements, is derived from the logo of the „ViKar"-project and is a symbol for the unity and the equality of the six colleges in this union. The circle is divided into three main areas. The inner zone of the structure represents the central communication and collaboration area, the middle circle is an area of subject-related information and location-related communication and the outer circle stands for a pure information area. In the center of the whole circle structure a pyramid can be found, the symbol of the city of Karlsruhe and for the „ViKar"-union. Six radial streets, starting from this center, connect the three main areas.

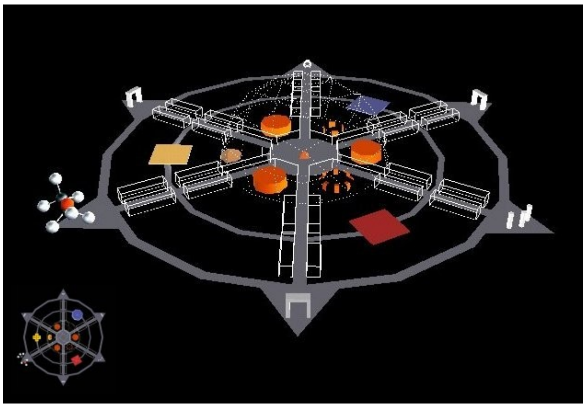


Figure 5: Basic structure of the virtual world

The most important components for communication and collaboration in teaching and learning are positioned around the central Pyramid's Place. The outer information area provides components for information and learning which can be used in a single-user mode. The in-between zone, the middle circle, contains discussion forums and exhibition areas.

## 4.3 The central communication and collaboration area

The central communication and collaboration area contains the Pyramid's Place and its surrounding components. The area is spatially defined by a transparent dotted dome.

### 4.3.1 Pyramid's Place / 2nd level campus

The Pyramid's Place (see figure 4) is a central location for communication, derived from a real campus. The students can meet casually and have a non-selective conversation while crossing the place.
The pyramid in the center of the place works as a „beamer" to another level of the virtual campus. At this second level the virtual world represents the six colleges close to physical reality. The basic structure is similar to the first level: the colleges are positioned at the outer circle as termini of six radial streets. This world is used for representation of the colleges and for preliminary orientation for students before their visit in physical reality.

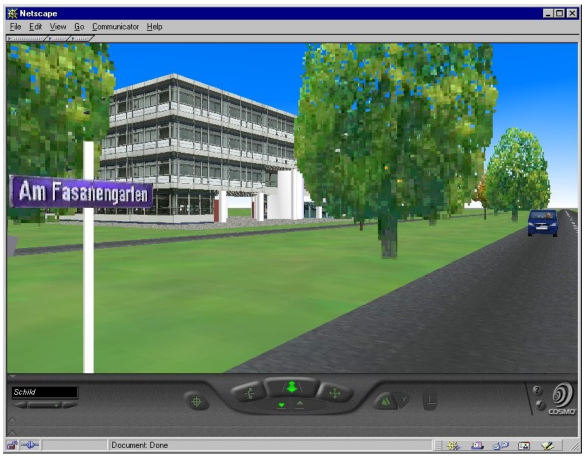



Figure 6: Virtual representation of the Faculty of Informatics at the University of Karlsruhe

### 4.3.2 Surrounding components

The surrounding components of the Pyramid's Place on the first level campus are differentiated into two formal groups, depending on appearance and content. The three components „auditorium", „bulletin board" and „multimedia lab" belong to the group of open structures. They have a different appearance, but the spatial demarcation of their areas is defined by linear structures, which allow a transparent vista through the components. The other three components „administration", „seminar building" and „student's house" are closed structures with an orange wrapper of the same appearance. Inside these components selective communication takes place.

## Auditorium

Lectures, presentations and official gatherings take place in the auditorium. The video screen offers to reuse films of former lectures and probably live video streaming in the near future. At the current state a video begins to run as soon as someone enters the area of the auditorium and only the users inside this area can hear the sound of the video. This means an additional spatial definition by function.

## Administration

The administration building area is planned to offer all administrative options a student, a staff member or a visitor needs, e.g. getting information, registration for the „ViKar“-college, modification of personal information, registration for exams or download of general materials.

## Multimedia Lab

The actual multimedia lab indicates a place for students to present their work in progress on several virtual terminals. This presentations can be used for discussing the projects with other people.

## Student’s house

The student’s house is established for collegiate working groups to collaborate. The students pick their exercising room depending on the project or subject they’re working on. The rooms are equipped with whiteboards, so students can illustrate their ideas. Furthermore the students can use a virtual object for discussion and exercise. The objects can be uploaded from the computer by using an appropriate form in the information-frame. Before uploading the chosen object has to be scaled and positioned by using the form.

## Bulletin board

The actual prototype indicates the idea of realizing a bulletin board in a three dimensional environment: Four boxes contain boards with predefined advertises; learning groups, travelling, assistant jobs and general. If the user finds an interesting advertise he can click on it and response by mail. After the realization of this idea the user should also be able to use a form in the information-frame to send his own advertise to be loaded on the bulletin board.

## Seminar building

Within the seminar building users can enter already existing rooms or even generate new temporal rooms for special needs after presetting individual parameters.

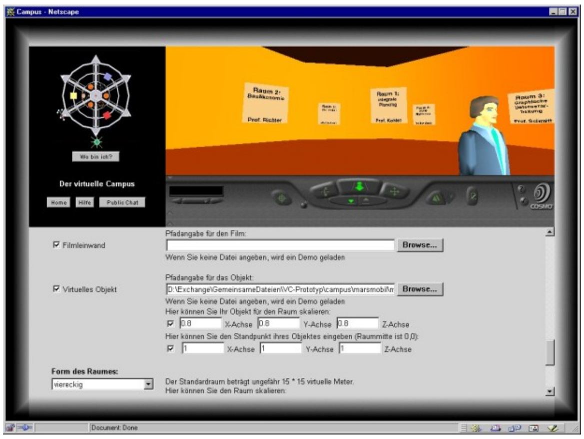



Figure 7: Room generation form

An input form in the information-frame offers options for presetting the shape, dimension, color and duration of usage of the new room. Furthermore the user can define which components he wants to be loaded into the room. The form offers a choice of whiteboard, slide projector, video screen and a virtual object. Slides for the projector, movie files for the video screen and virtual demonstration objects can be uploaded by browsing on the own computer. The objects can be scaled and positioned in the same way as in the student’s house.


Figure 8: Generated seminar room with slide projector, video screen and demonstration object

After sending the form the room will be generated. The idea is to store this temporal room on the server for the predefined duration of time. The option of generating new rooms should be reserved to tutors or staff members, for instance by password protection. Additional ideas are limiting the usage of the room to certain persons, e.g. participants of a course, and automatically sending e-mail information to participants when creating a new room.

## 4.4 THE MIDDLE CIRCLE

Within the in-between zone of the middle circle the user can find selective communication and non-selective information. There are three colored places that serve as discussion forums. Here we use the above proposed „object classes": The places have the same function, but they're differentiated in color and appearance.

The user becomes aware of entering a place because of its spatial definition and the height limited by some kind of roof. In our current prototype an IRC window appears within the information-frame. Only the people located in the same place at the same time can chat together through this channel. All discussion forums are equipped with a whiteboard which the participants can use in their discussion.

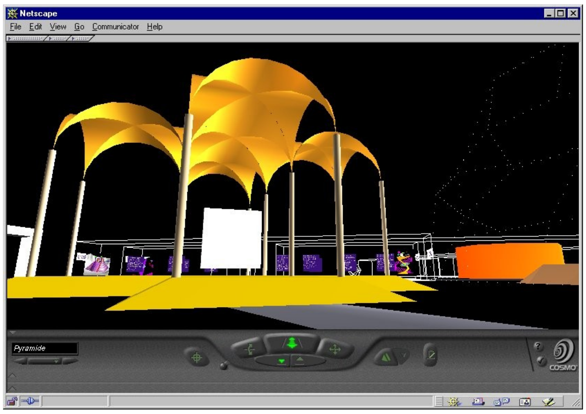



Figure 9: Yellow forum with whiteboard

The radial streets with abstract exhibition areas on both sides also belong to this area. Inside the galleries, which are defined by line structures, all kinds of objects or information can be presented. The galleries could show for instance finished students projects, college developments, research results and artwork. By approaching or clicking objects or images further information and multimedial content can be displayed both in the virtual world and in the information-frame.

The user is invited to rummage among these objects and to gain some new information, cultural experience or inspiration for the own work. By the way the exhibitions can add cultural components to the virtual campus as well as they encourage casual communication. With changing exhibitions the user is also motivated to come again and look for new presentations the next time.


Figure 10: Radial street with exhibition area

## 4.5 THE OUTER CIRCLE

At the outer circle there are three components that deliver information to the student. They are planned to work as single-user areas for learning and research. The components „knowledge tree", „search engines" and „personal workplace" alternate with the three exits at the end of the radial streets. All components in the outer circle serve as experimental objects; they are not yet elaborated.

### Knowledge Tree

The symbol used for knowledge is a model of molecule. By clicking the molecule a connected abstract world containing a cone tree is being loaded into the world frame. By clicking the elements of this cone tree all subjects the student can study and all courses he can participate are displayed in a three-dimensional hierarchical order. Using this cone tree is an experiment with 3D data visualization, we are not sure yet if this could be more reasonable than a two-dimensional information display.

## Search engines

Here the user can gain information by applying three different search engines for the internet, the internal databases and the electronic library of „ViKar". The actual representation serves as a presentation of this idea and is to be developed further.

## Student's personal workplace

Each student needs a personal workplace. The workplace is symbolized with a linear structure containing a computer display. By clicking this display the student loads both the workplace in the virtual world frame and the appropriate HTML-interface in the information-frame.

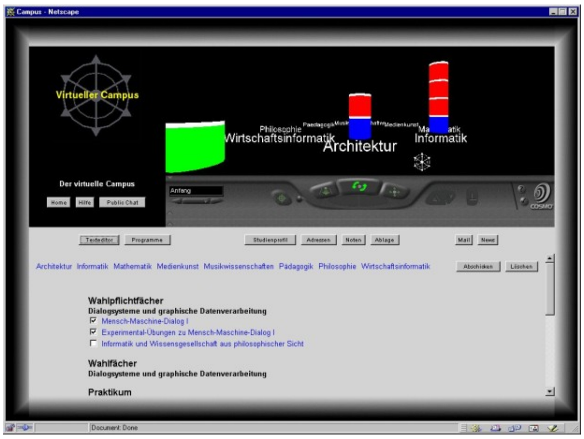



Figure 11: Student's workplace

The working place could offer interfaces to the learning environment and other „ViKar"-components, an interactive personal profile with 3D display of the student's actual status of studies (see figure 11), the storage of collected 3D exercising objects, buttons to access software, an address book, the college's newsgroups, an overview of grades and a lot of other features.

## 5. Conclusion and Outlook

In this paper we discussed the requirements for a virtual university campus considering social needs. We described the possible advantages a 3D virtual reality multi-user environment can contribute to a virtual campus and suggested the combination of a virtual world with multimedial and text-based content. In our opinion, a lot of solutions in the field of large-scale 3D virtual worlds are not satisfying concerning the issues of design, orientation and navigation. For these issues different features and strategies have been proposed; the implementation of these means is presented in our prototype, as well as some content components for a virtual campus environment.

Regarding the aspects of communication, collaboration and interaction we defined requirements for the Virtual Campus Project. The available systems we tested so far do not meet these requirements at present state. But we're optimistic that technical progress in the near future will offer particularly higher bandwidth and usable solutions to realize our vision of a collaborative, social interactive virtual campus with audio communication and shared persistent objects.

Our future works will contain further research in the design of virtual environments, especially in the psychological aspects of shapes and colors as well as in issues of orientation. After some more investigations in VR multi-user systems one of the next steps will be the test implementation of our prototype in a multi-user system. On this basis we're going to do some case studies with students using this environment.

Apart from these investigations we will work on the realization and implementation of suggested ideas, features and functions. Furthermore the integration and the cross-linking of other components, which are currently developed in the „ViKar"-Project, are main tasks to establish a holistic educational environment.

## References


[1] J. Kloss, R. Rockwell, K. Szabó, M. Duchrow. *VRML97*, Addison-Wesley-Longmann Verlag, Bonn, 1998.

[2] M. Pesce. *VRML - Cyberwelten erkunden und erschaffen*, Carl Hanser Verlag, München, 1997.

[3] N. Stephenson. *Snow Crash*, Goldmann Verlag, München, 1995.

[4] S. Guðjónsdóttir. *Virtueller Campus*, Unpublished Master Thesis in Architecture, University of Karlsruhe, 1999. http://ifib41.ifib.uni-karlsruhe.de/lehre/diplom/sigrun_gudjonsdottir/index.html .

[5] R. Kuhn. *Virtuelles Institut,* Unpublished Master Thesis in Architecture , University of Karlsruhe, 1998. http://ifib41.ifib.uni-karlsruhe.de/lehre/diplom/rainer_kuhn/virtinst/index.htm .

[6] R. Rockwell. *From Chat to Civilization: The Evolution of Online Communities*. http://www.blaxxun.com/company/vision/ index.html

[7] R. Rockwell. *Wanted: Cyberspace Architecture*. http://www.blaxxun.com/company/vision/index.html .

[8] J. Suler. *The Psychology of Cyberspace,* April 1999. http://www.rider.edu/users/suler/psycyber/psycyber.html .

[9] Blaxxun Interactive. Corporate Web Page. http://www.blaxxun.com/ .

[10] Active Worlds. Corporate Web Page. http://www.activeworlds.com/ .

[11] K. Saar. VIRTUS: A Collaborative Multi-User Platform. *Proceedings of the VRML 99 Symposium*, pages 141-152, ACM SIGGRAPH, New York, 1999.

[12] University of Karlsruhe, Homepage of the „ViKar"-Project. http://vikar.ira.uka.de .

[13] P. Deussen et al.. *Finanzierungsantrag für „ViKar - Virtueller Hochschulverbund Karlsruhe"*, Karlsruhe, 1998.

# Natural Language Generation in Multimodal Learning Environments: Lifelike Agents And 3D Animated Explanation Generation

James C. Lester
The IntelliMedia Initiative
Department of Computer Science
North Carolina State University

## Abstract

Natural language generation (NLG) has witnessed great strides over the past decade. Our theoretical underpinnings are firming up, our systems building activities are proceeding quickly, and we are beginning to see significant empirical results. As a result of this maturation, the field is now well positioned to attack the challenges posed by a new family of computing environments: interactive 3D worlds, which continuously render the activities playing out in rich 3D scenes in realtime. Because of these worlds' compelling visual properties and their promise of a high degree of multimodal interactivity, they will soon form the basis for applications ranging from learning environments for education and training to interactive fiction systems for entertainment.

As the visual complexities of interactive 3D worlds grow, they will place increasingly heavy demands on the visual channel. To complement their rich visualizations, interactive 3D worlds will require the linguistic flexibility and communicative power that only NLG can provide. In interactive learning environments, the spatial complexities and dynamic phenomena that characterize physical devices must be clearly explained. NLG delivered with speech synthesis will need to be carefully coordinated with 3D graphics generation to create interactive presentations that are both coherent and interesting. In a similar fashion, lifelike agents roaming around the same 3D worlds through which users guide their avatars will require sophisticated NLG capabilities, and 3D interactive fiction systems will benefit considerably from virtual narrators that are articulate and can generate interesting commentary in realtime.

In this talk, we will explore the major issues, functionalities, and architectural implications of natural language generation for interactive worlds. Our discussion will examine NLG issues for two classes of interactive worlds: lifelike pedagogical agents and self-explaining learning environments. Lifelike pedagogical agents generate advice combining speech and gesture as users solve problems by guiding avatars through 3D worlds and manipulating devices housed in the worlds. Self-explaining environments dynamically generate spoken natural language and 3D animated visualizations and produce vivid explanations of complex phenomena. We will illustrate this work with systems under development in our laboratory.

# Speaking Spaces: Virtual Reality, Artificial Intelligence and the Construction of Meaning

Stephen N. Matsuba
The VRML Dream Company
3537 Dundas Street
Vancouver, BC
Canada V5K 1S2
matsuba@vrmldream.com

## ASTRACT

This paper explores how linguistic methodologies can be used to analyse virtual reality applications. VR environments function in ways that are similar to instances of human languages in that both are endowed with some form of signification through the construction of meaning. We can, therefore, describe the syntax of a VR world. By combining syntactic analyses of VR with a consideration of other factors such as culture and discourse, we can develop a "grammar" of VR. This grammar can be used in AI/VR applications. It can also become an important factor in understanding how the human mind works.

**Keywords:** virtual reality, artificial intelligence, VRML, virtual environments, linguistic analysis, semiotics, culture, discourse

## 1 INTRODUCTION

In the folklore of computer technology, two icons have emerged as its holy grails. One is the HAL 2000: the artificial intelligence (AI) from *Stanley Kubrick's 2001: A Space Odyssey*. The other is the holodeck: the virtual reality (VR) system from *Star Trek: The Next Generation*. There are, of course, no HALs or holodecks among today's computer applications. However, this situation has not diminished interest in AI and VR development. Corporations and research institutions are spending millions of dollars to develop natural language systems that provide computers with a more "human-like" interface. VR is coming into the mainstream through games and business applications. Indeed, HAL and the holodeck are so embedded in the popular psyche that they have become the touchstones by which current AI and VR applications are assessed.

The buzzword that underlies this interest is *interactivity*. Both HAL and the holodeck represent the most engaging forms of human-computer interfaces. The former not only can beat you at chess, but will also engage you in a lively conversation while doing so. The latter immerses you in a virtual environment that is full of visual, tactile and auditory stimuli that are indistinguishable from the real world. The level of interactivity they depict continues to fuel the public's—and researcher's--interest in AI and VR.

At first glance, the kind of interactivity found in AI seems very different from that of VR. AI focusses on cognitive interactions involving language. People become engaged with the computer by speaking to it and having it talk back. AI's components are phonemes, lexicons, syntactic and semantic systems, and knowledge representation. Interaction in VR, on the other hand, seems to be primarily physical in nature. People walk through a virtual environment, grab and move objects, and listen to spacialized sound cues. The terminology associated with this technology involves actions and physical materials: objects, textures, events and behaviours.

However, interaction in today's VR applications—and those of the foreseeable future—is not physical at all. We do not actually walk through a virtual environment; we use input devices to navigate through it. We do not really pick up a virtual object, we activate a behaviour that simulates our picking it up. Our interactions with VR applications have less to do with kinesiology and more to do with cognitive processing.

In fact, our responses to VR are similar in many ways to those of AI. We provide linguistic inputs to an AI application and interpret its language-based output. In VR, we view what is happening on the screen or head-mounted display, interpret what we are seeing and then respond to it. Indeed, we can have a better understanding of interactivity in VR if we view it in light of the work being done in linguistics and artificial intelligence.

## 2 VR AS LANGUAGE

The cognitive aspects of VR are readily apparent. We view a scene, analyse the information, and then react to it. However, this system does not end with visual processing. We also interpret what we see in the virtual world in order to make sense of it. In many respects, a virtual world takes on the characteristics of a linguistic utterance.

The idea that VR has some relationship to language is not new. Indeed, researchers have drawn parallels between linguistic and visual processing for some time. In 1970, R. L. Gregory attempted to show how Chomsky's Transformational Grammar is related to visual processing. He states:

> The deep structure surely represents hypothetical meanings allowed by the words and the surface structure in a sentence. This is very similar to the meaning given to retinal patterns in terms, ultimately, of the structure of the world of objects. (165)

Gregory's application of Transformation Grammar to an understanding of visual images is, admittedly, rather tenuous. However, he raises an important point. The cognitive processes involved in understanding the world that we see are directly related to how we understand language.

This view is adopted by many of the commentators on VR. Brenda Laurel, for example, argues that VR is more than a visual medium. She notes:

> While it is true that its principal lines of research all began before VR had a name, VR was the nexus that revealed new synergies and gave each of them new impetus and a new arena of expression. Although vision is certainly the means by which we perceive text and numbers on computer screens or hard copy, the primary activity is not sensory or perceptual but cognitive (202-03)

N. Katherine Hayles echoes this view when she defines virtuality as "the perception that virtual structures are interpreted with informational patterns" (4-5).

Both Laurel and Hayles believe that our experiences with VR are not limited to sensory or visual aspects. According to them, it is not enough for us to see a virtual world and navigate through it. We must also mentally process it in order to endow it with meaning. In other words, we need to assign some kind of signification within a virtual world in the same way that we impose signification on a linguistic event such as a conversation or a novel.

When we navigate through a virtual environment, our expectations are very different from the ones that we have while walking down the street to buy a liter of milk. In the real world, we view our surroundings looking for visual cues to establish our position relative to our destination. We may or may not appreciate the weather, notice someone walking on the street, or any number of other possible stimuli. We may choose to take a different route to alter the sensory inputs that we receive. However, our experiences while navigating through a real space are different from those we experience in a virtual one. No person or group of people have set up the particular set of stimuli that we experience as we go to buy milk. The people, sights, sounds and smells that we perceive are the result of a myriad of random elements coming together at a particular moment.

A virtual environment, however, has a specific creator or group of creators, and the events and stimuli experienced in it have been planned to the smallest detail. Moreover, the creators of such a world generate a desire among users to interact with this space. In other words, a virtual environment must be created with a purpose. Users interact with the virtual space and interpret these experiences in order to fulfil this purpose.

This act of interpretation involves the construction of meaning. Indeed, we can compare how we react to a virtual environment to how we read a novel. A writer creates a text that must have some qualities that attract people to read it. It also has some underlying purpose that may, or may not, be made explicit. Readers interact (that is, read) with the text in some manner and interpret their experiences. During this process, they impose a structure for signification on the text and may, or may not, gain some form of satisfaction.

In a VR application, a developer creates a virtual world that entices people to interact with it. Like the novel, it has an underlying purpose that may, or may not, be made explicit. Users navigate through the space manipulating objects and triggering events as they go along. They will also be cognitively processing their experiences and imposing some kind of significance on what they see and do. The success of the virtual world depends on whether or not this signification generates a sufficient level of satisfaction among users.

We can, therefore, draw a direct link between the cognitive processes used in understanding language and those involved in interpreting VR. In fact, one can argue that language is an underlying component of this technology. In discussing landscape and narrative in virtual environments, Margaret Morse asserts that cyberspace[1] is "not merely a scenic space where things could happen; it also incorporates the artificial intelligence or

[1]Morse equates cyberspace with virtual environments.

*agency* that orchestrates the virtual scene" (198). Morse goes on to describe how Ivan Sutherland's concept of the "ultimate display"[2] is dependent on language for its effect:

> Note that the realism of the ultimate display does not depend on appearing like reality or verisimilitude or *referential illusion* … but on the power of language or the display to make reality itself, or enunciative illusion (that is, appears to call a symbol forth into existence by speaking or drawing it; or what is known in linguistics as performative speech). (207-08)

For Morse, our experiences in VR are directly associated with our interpretation of the symbols that are presented by the virtual environment. In fact, she goes on to state that "the very notion of an environment or landscape … becomes more and more difficult to distinguish from the symbolic field itself" (227).

If we accept Morse's premise that a virtual environment is indistinguishable from the "symbolic field," it is possible to see VR as a form of language. Moreover, this language can be examined and interpreted using the same paradigms we use to understand human languages. In other words, we can apply a syntactic analysis to VR.

## 3 THE SYNTACTIC ANALYSIS OF VR

If VR is a language, we can assume that there is a production system that governs how objects, events and behaviours can be combined in various applications: that is, there must be a grammar of VR. This grammar, like those of human languages, would allow us to generate an infinite number of possible virtual environments. Moreover, these environments will have meaning and a recognizable stream of signification for other users.

But how do we establish the rules that govern a grammar of VR? While we can question VR developers about their methodologies for creating a virtual environment, they will not have a full comprehension or awareness of all the factors that influenced their decisions. In discussing a speaker's knowledge of the rules of a language, Chomsky notes:

> A person who knows a language has mastered a set of rules that assigns sound and meaning in a definite way for an infinite class of sentences…. Of course, the person who knows the language has no consciousness of having mastered these rules or of putting them to use, nor is there any reason to suppose that this knowledge of the rules of language can be brought into consciousness. (103-04)

The same conditions apply to our knowledge of the linguistic rules governing VR. Let us say that we wish to create a virtual world that contains a table and chair. We know that under normal conditions, both the bottom of the table's legs and the bottom of the chair's legs lie along the same horizontal plane. We also know that there is a physical relationship between the two objects. There must be a certain distance between the top of the chair's seat and the lower edge of the table. The chair should not be wider than the length of the table. However, how do we determine where to place the chair relative to the table? What angle should the chair's orientation be in comparison to that of the table? How do we know when the positions of the two objects are "just right"?

Given that we cannot gather direct evidence for the linguistic rules governing VR, we need to work by observation and analysis. We must work in the following three stages:

1) collect a set of data relevant to the phenomenon we are studying,
2) develop a set of rules that will account for that data, and
3) test our rules against other sets of data.

As with linguistic investigations of human languages, the ideal source for our corpus would be virtual environments that are already in existence. This criterion is not as restrictive as it may seem. It is true that early VR applications were few in number because of the requirements of the computer hardware needed to render a virtual world and the level of programming needed to create the application. However, a remarkable drop in the cost of computer hardware with 3D-accelerated graphics and the proliferation of applications such as 3D games and the Virtual Reality Modeling Language (VRML) have resulted in an exponential increase in the number of VR applications that are currently available.

Once we have determined the source for our data, we must find a way to describe it in linguistic terms. On the surface, linguistic description seems ill suited for describing VR applications. We are not dealing with nouns, verbs and modifiers; instead, we are dealing with objects, textures,

---

[2] In 1965, Sutherland described a computer interface that is very similar to the concept of the holodeck:

> The ultimate display would, of course, be a room within which the computer can control the existence of matter. A chair displayed in such a room would be good enough to sit in. Handcuffs displayed in such a room would be confining, and a bullet displayed in such a room would be fatal. With appropriate programming such a display would literally be the Wonderland into which Alice walked. (As quoted in Morse, 207)

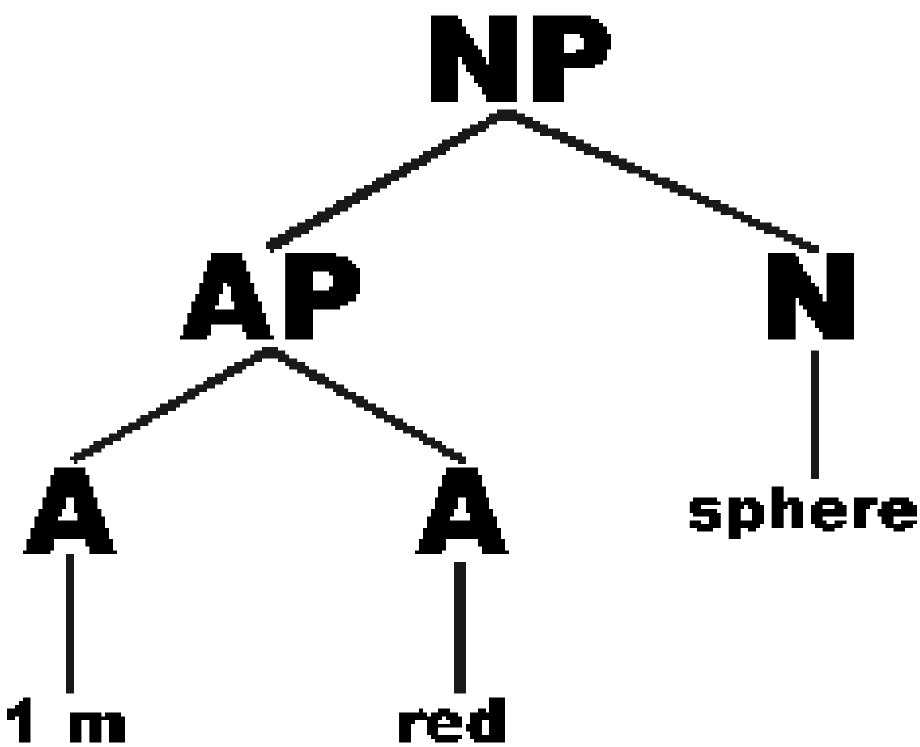



Figure 1: The Syntactic Parse for a Red Sphere with a One Meter Radius

materials, events and behaviours. Nevertheless, it is possible to apply a syntactic analysis to VR.

Consider a simple red sphere in an otherwise empty virtual environment. The sphere is one meter in radius. We will label this sphere as a *noun*. If we use a system such as Transformational Grammar for our analysis, we would say that we have a *noun phrase* (NP) with a noun (N) with the value *sphere*. We must also note that N is modified by an adjectival phrase (AP) consisting of two adjectives (A): one representing the radius and the other representing the colour *red*. Figure 1 shows the tree diagram depicting this structure.

At this simple level, we can begin to see how a linguistic description of a virtual environment can work. This is not to say, however, that the application of these methods to VR will be transparent. We can label objects such as a ball or a globe as *spheres*. However, what happens when the globe is not a texture-mapped sphere but a model of the Earth complete with geometry depicting mountain ranges, oceans and cities? We cannot simply categorize it as a sphere. We need to make a distinction between the two.

A key component of our syntactic study of VR will be to establish a criteria for labeling objects. These labels have to be both flexible and detailed. For example, the NPs for a texture-mapped sphere and a complete model of the Earth can be both labelled as *globes*. However, N for the former will have the value *sphere* while the latter can be called *geographic model of the Earth*.

The issue of describing objects becomes more complex when we are dealing with models that have different parts. For example, we will usually agree whether or not an object is a chair. However, how do we distinguish between a chair with round legs versus a chair with square ones?

One solution is to treat a chair not as a single object, but as an NP made up of several NP components. Under this rubric, a chair would be NP” and each of it different parts—the back, the seat and the legs—would be NP's that are subsets of NP”. The back of the chair (NP') is comprised of N, representing the geometry, and a number of APs that define its attributes (colour, size, et cetera). The seat is another NP' that has N (geometry) and AP (attributes). The legs are also represented by NP'. However, this noun phrase is divided into four separate NPs. Each NP defines one of the chair's legs and is comprised of N (geometry) and AP (attributes). We can now draw a diagram depicting the structure for our chair (see figure 2). While this method of description is cumbersome (and in the case of complex objects, practically overwhelming), it provides a degree of accuracy that enables us to develop a more complete hypothesis for the rule governing a grammar for VR.

Describing our adjectival phrases will also be problematic. When you read “the red sphere,” the task of parsing this phrase is not difficult because all the data is readily apparent. When presented with a red sphere in a virtual environment,

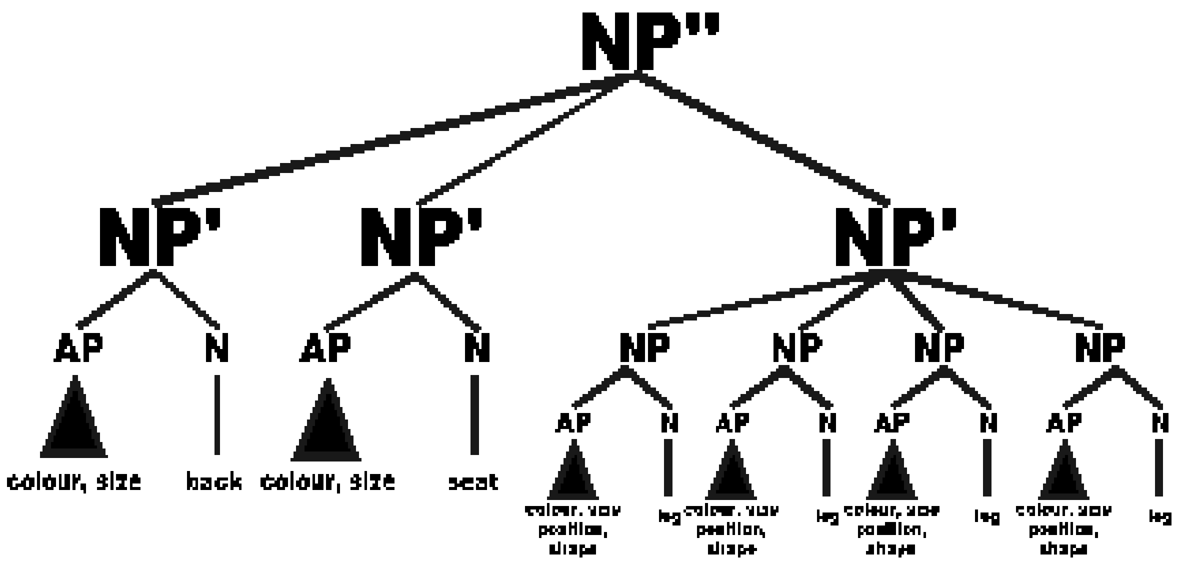



Figure 2: Syntactic Parse for a Chair

however, you must deal a number of factors that are not easily discerned. How do we describe the quality of the colour? Is it shiny? Does it glow? Moreover, what is the sphere's position in relation to the rest of the environment? What is its orientation? Standard language-based descriptions cannot adequately make these distinctions. We need to find a way of classifying and describing these modifiers using a methodology that can be applied to any VR application.

The Virtual Reality Modeling language (VRML) provides one such a solution. VRML is a scripting language that is a standard format for creating virtual environments and objects. Its goal is to be flexible so that any type of object or behaviour can be created using its system of nodes and fields. It consists of a set of nodes that instructs a VRML browser to render objects, textures, behaviours and events. Each node has a number of fields that delineates its attributes. For example, we can assign a colour to an object by using the *Material* node. This node has a set of fields that controls the colour of the object, states its level of transparency, defines its shininess, and whether or not the object emits a colour. To make our sphere appear red, we can set the *Material* fields as follows:

| | | | |
|---|---|---|---|
| diffuseColor | 1 0 0 | ambientColor | 0.2 |
| emissiveColor | 0 0 0 | shininess | 0.0 |
| specularColor | 0 0 0 | transparency | 0 |

These values will give our sphere a solid bright red colour with no reflections. We can also specify the position of an object relative to the entire environment using the field values of the *Transform* node. For example, we can position the sphere in the center the world along the *x* and *z* axes and raise it one meter along the *y-axis* by using the following field: *translation 0 1 0*. If we use these field names and values to describe our adjective phrases, we can provide a consistent, accurate and complete categorization of all the descriptors that are modifying an object. Figure 3 shows the tree diagram for this example.

Describing events and behaviours poses the greatest obstacle to our syntacti analysis. These elements function as verb phrases (VP). However while we can say that an avatar walks from Point A to Point B, the number of variables involved in rendering this image makes such a description totally inadequate. How quickly is the avatar walking? What direction is it going? How do we describe its gait? These elements are not clearly defined by the term *walking*.

Moreover, employing VRML to describe events and behaviours will not provide us with an acceptable solution. It uses keyframes and keyframe values to define these elements for a virtual environment. For example, we can set our sphere to rotate 45 degrees at a particular moment of time by setting a *OrientationInterpolator* with the appropriate *key* and *keyValue*. We then use the*ROUTE* node to define which object is affected by the interpolator. In this way, we are able to create a wide range of behaviours and events in a VRML world.

Using *key* and *keyValue* to define our verb phrases, however, is not a practical method for describing a VP. While simple behaviours can be created using a relatively small number of keyframes, these values are simply a set of numbers denoting time signatures as well as translation, scalar, rotation and colour field values. They will appear to be meaningless to a researcher. Even

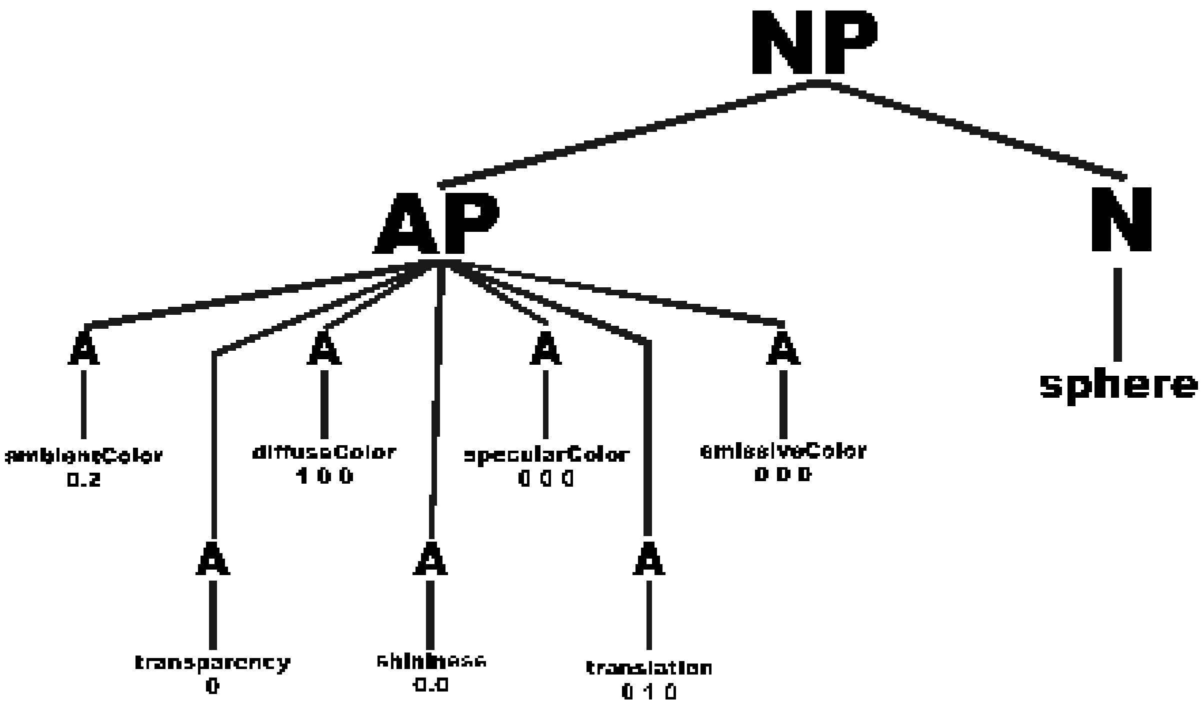



Figure 3: Syntactic Parse Using VRML Nodes and Fields

with a trained eye, someone examining such a description would have no idea as to what the verb phrase was actually denoting. Moreover, complex behaviours and events can require hundreds or even thousands of keyframes. Trying to draw a tree diagram with this number of values would render any analysis impossible.

A compromise solution would be to establish a standardized set of labels that can account for the majority of events and behaviours. These labels must provide a certain level of accuracy. They will also need to be descriptive enough so as to allow someone looking over the data to visualize the event or behaviour. If our red ball was to fall slowly for one meter, for example, we might describe this event as follows:

| <u>The red ball</u> | <u>falls one meter for 2 seconds</u> |
|---|---|
| NP | VP |

Note that we have added the duration of the fall as well as its distance in order to clarify the nature of the event. These elements function as an *adverbial phrase* (ADVP) with an adverb representing the distance of the fall (*one meter*)[3] and an NP with N representing the type of modifier (*time in seconds*) and a determiner (D) representing the value of N (*2*). We can represent this event in a graphical manner using a tree diagram (see figure 4).

The above description is, of course, only a preliminary model for a linguistic description of VR. A number of issues still need to be determined. The model needs a way of dealing with triggered events (such as clicking on a box to make it float in the air). It also requires a method for dealing with dialogue and ambient sounds. Nevertheless, we can see that a linguistic description of VR is possible.

## 4 The Next Phase: VR as a Social Semiotic

We have so far explored the methodology for determining a grammar of VR, but not the reason for taking on such an endeavour. For Chomsky, exploring the linguistics of human languages is a way to discover the nature of the human mind. He notes:

> Personally, I am primarily intrigued by the possibility of learning something, from the study of language, that will bring to light properties of the human mind. We cannot not say anything particularly informative about the normal creative use of language itself. But I think that we are slowly coming to understand the mechanisms that make possible the creative use of language,

[3]Although the ADVP in this case would be parsed as an NP if we were examining English syntax, it functions more like an adverb modifying a VP in our analysis of VR.

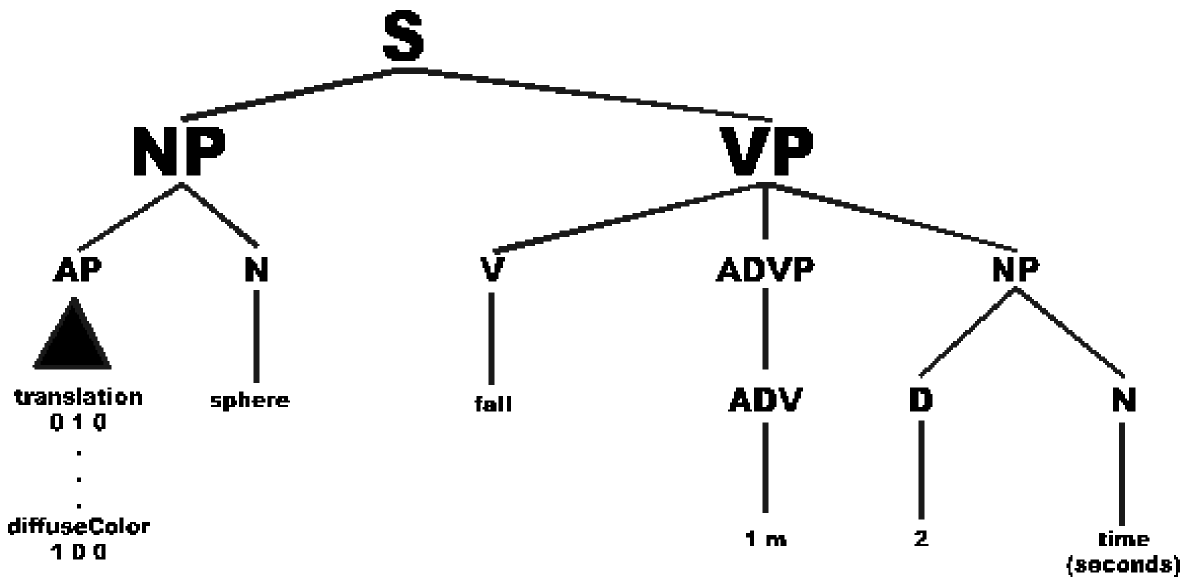



Figure 4: Representing Behaviours in a Syntactic Parse

> the use of language as an instrument of free thought and expression. (103).

An exploration of the linguistics of VR, therefore, can provide us with another avenue through which we can understand the construction of meaning. Moreover, it will allow us to link visual and other sensory elements with language-based ones. By combining these two areas, we can begin to establish a unified model of cognition. We will learn not only how we derive meaning and significance from human language, but will also establish paradigms for understanding how this process is related to the ways in which we interact with the world around us.

However, a syntactical analysis of VR alone is not sufficient for a cognitive model of the mind. The former tells us about the structures that are possible in VR, but it does not explain why they are significant to us. Therefore, we must also consider ways to extend our analysis.

An important component of this extension is to consider the social aspects of language. In *Language, Context, and Text*, M. A. K. Halliday and Ruquiya Hasan argue that language is a "social semiotic": that is, language can be understood in terms of its relationship to social structure. Halliday notes:

> A text, then, is both an object in its own right … and an instance—an instance of social meaning in a particular context of situation. It is a product of environment, a product of a continuous process of choices in meaning that we can represent as multiple paths or passes through the networks that constitute the linguistic system. (11)
>
> Hasan argues that if a text can be described as "language doing some job in some context," we can also declare that it is "the verbal expression of a social activity" (56).

We can certainly point to areas where visual elements in one culture do not have the same connotation in another. In most Western countries, the colour white can denote purity and innocence. It is not unusual, therefore, to see a white dress on the bride during a wedding. However in Chinese cultures, white is the colour of death and is considered bad luck. Traditional bridal garb in this social context is red. If we place a white wedding dress in a virtual environment, a Western viewer will probably have a positive reaction to seeing it. A Chinese viewer, however, will find this image disturbing. The latter's cultural knowledge would alert her to the fact that certain assumptions—a wedding is a celebratory event; a wedding should bring good luck; red is the colour that signifies good luck; white signifies bad luck—are being broken.

This kind of cultural coding is present in any form of communication, but it is particularly important in visual forms. Gunther Kress and Theo van Leeuwen argue that as we move from a verbally-oriented society to a visually-oriented one, the need to understand the cultural influences on our interpretation of visual communication becomes even greater. They note:

> Visual communication is always coded. It *seems* transparent only because we know the code already, at least passively—but without knowing what it is we know, without having the means for talking about what it is we do when we read an image. A glance at the "stylized" arts of other cultures

> should teach us that the myth of transparency is indeed a myth. We may experience these arts as "decorative," "erotic," "mysterious" or "beautiful," but we cannot understand them as communication, as forms of "writing," unless we are, or become, members of these cultures. (32)

In order for VR to work as a language, the "speaker" (the person or persons who created the virtual environment) and the "receiver" (the user of that environment) must belong to the same culture. Seeing a man dressed in black tights contemplating a human skull may initiate a flurry of meanings and associations to someone with a knowledge of English-speaking culture, but it is doubtful that the same image will have a similar resonance to someone with an awareness of only Bedouin culture.

Moreover, we cannot limit our understanding of the social aspects of VR to the study of specific icons. Our cognitive model of VR must also deal with virtual worlds on a discoursal level. If we can say that a VR environment is a linguistic act--that it can be used to express some kind of meaning--we need to determine the rules that govern how its various components can be linked to generate some form of signification. Indeed, much of a VR application's impact depends on the narrative structures that underlie it. Laurel notes that the study of how we "construct contexts and narratives in VR is a recent but equal partner to the study of human sensation and perception as we ponder what is going on in virtual worlds" (206). According to her, our reactions to VR are dependent on dramatic forms. She notes:

> Our experiences with VR to this point also confirm that human experiences in virtual environments are enhanced by dramatic forms and structures that support complex emotional textures. Indeed, these characteristics emerge from the constructive activities of human interactors even when they are not purposefully embedded in the potentialities of the virtual world. (207)

The detection of these forms in VR does not mean that the creator of a virtual world intentionally included them. They may have been subconsciously inserted, or the viewer's own biases and idiosyncrasies may have imposed them on to the world. However, the rationale for the presence of these dramatic forms is not important. What is important is that a common set of cultural assumptions must be shared between the creator of a virtual environment and its users in order for that world's communicative potential to be fulfilled.

An early attempt to use VR as a discursive medium is "Construct(s) and Même-ing." It is a series of five virtual worlds that form an "essay" on new media and virtual reality. It is also an experiment to see if virtual reality can function as the "text" for making an argument. Each of the five virtual environments in "Construct(s)" deals with a particular issue relating to virtual reality and new media. In the third environment, for example, a transparent model of a human brain floats above the ground (see figure 5). Within this model, three spheres are linked to a cube. Behind it are five pairs of panels that alternate position (see figure 6). Each of these panels contains an image that is linked to the other in the pair. The environment also contains a large model depicting the characters "2B." When a user approaches these characters, a gong sounds, the models sink into the ground and another pair of characters ("D4") rises to the surface. The different elements in this environment are meant to convey the idea that new media is dependent on fixed cultural assumptions. Users wandering through the virtual environment must make a number of connections between a variety of elements in order to comprehend this pattern of thought. The brain model with its fixed spheres and cube contrasts with two other environments in which models of a brain have spheres orbiting a rapidly spinning cube. The image panels deal with allusions and their referents: Mishima and the samurai; Proust and Wagner; T. S. Eliot and Shakespeare; Kurosawa and Van Gogh; Mrs. Peel (from the television series *The Avengers*) and the Gunpowder Plot. The characters "2B" and "D4" are the hexidecimal equivalents of the line from *Hamlet*: "To be or not to be."

However, the success of this environment is entirely dependent on the ability of the users to recognize the various references associated with the

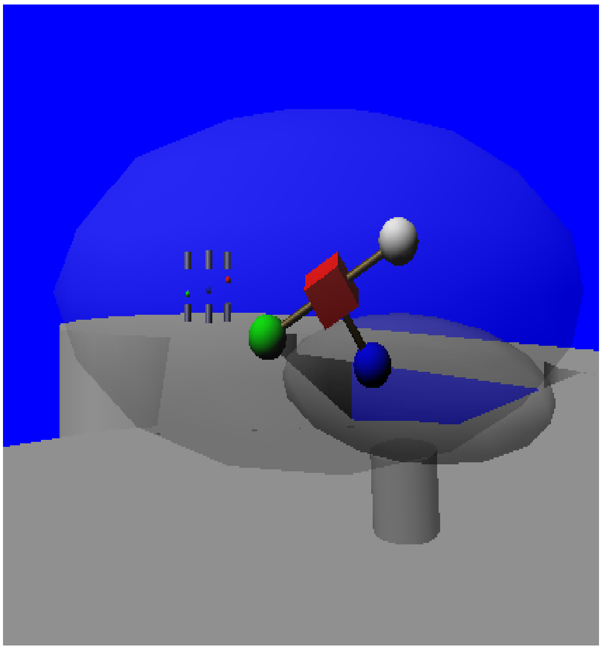


Figure 5: A Brain Model from "Construct(s) and Même-ing"

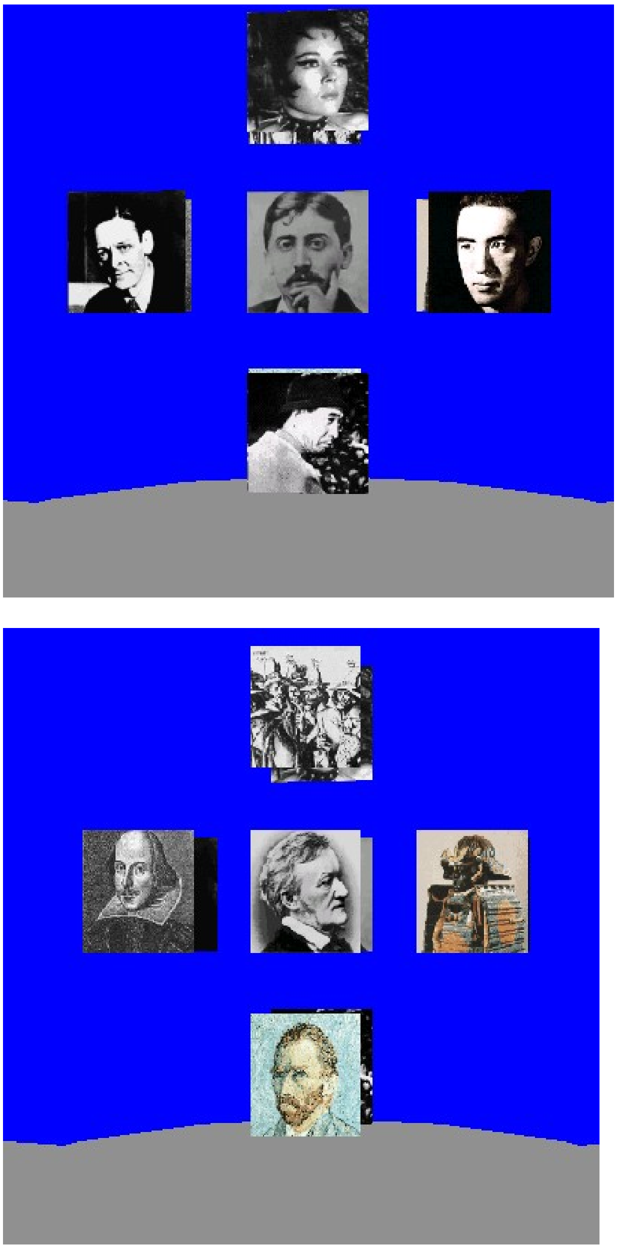


Figure 6: The Image Panels from "Construct(s) and Même-ing

various objects. They must have a certain level of cultural knowledge in order to find any significance in what they see: Eliot's use of Shakespeare, Kurosawa's film *Dreams*, Proust's references to Wagner's *Parsifal,* British television of the 1960s, and the works of Mishima. It is readily apparent that these referents all have a specific cultural bias. Users without the appropriate knowledge of English, European or Japanese culture would not be able to identify the different elements. More importantly, they would not be able to make the cognitive links necessary to perceive the underlying message of this environment.

Therefore, a cognitive model for a VR grammar must find ways to incorporate the social and cultural factors that influence the production of meaning. Morever to achieve this goal, we have to consider a grammar that not only can account for the relationship between syntax and semantics, but will also deal with the technology's discoursal function. It is not enough to understand how different elements come together to create an object and its behaviour. We also need to understand how all the elements in a virtual environment work together to fulfil a certain purpose. In essence, we must endeavor to turn our model of the language of VR into a model of the mind.

## 5 VR, AI AND BEYOND

The ultimate goal of a VR grammar would be to create an AI application that can generate original and compelling virtual worlds. This AI would have its own form of the Turing Test. We would place users in a room with a computer display and the appropriate input devices. They would be told that several virtual environments will appear on the display for them to try out. They would also be informed that some of these worlds are created by humans and the others by the AI. The users would then be asked to evaluate whether a particular virtual environment is man-made or generated by the computer program. The AI would pass the test if every user is fooled into believing that its worlds were created by a human.

While this goal would prove that our VR grammar is an accurate model of the human process, it would not find many practical applications. However, practical applications can be developed by applying a VR grammar to AI. We could, for example, create AI systems that translate virtual environments from one cultural context to another. The AI would determine the culture of the user and then replace certain objects and behaviours with ones that are more appropriate. VR developers would not have to worry that a user from a different culture would not be able to use their creations. This kind of application will become more important with the continued growth of the Internet and as globalization affects more markets.

Another possible application would be an AI system that translates virtual worlds and objects into signals that can be understood by the visually impaired. This type of AI would use a VR grammar to analyze a virtual environment and then translate its analysis into sounds and tactile feedback. It could also enable this group of people to create their own VR applications.

We can see that the linguistic analysis of VR is more than an academic exercise. It has implications for understanding the workings of the human mind. It also has real-world applications that can greatly expand the ways in which we communicate with each other.

However, the immediate result of this kind of analysis will be to expand our understanding of VR and its impact on us. If a virtual environment can be seen as a linguistic act, then we must first determine what we want to say and then design how we want to say it. If signification is achieved not by visual cues alone but by also incorporating narrative and dramatic forms, then we must be sensitive to the social and cultural influences that affect these forms. In the end, it is not enough to make our VR worlds visually pleasing; we must endow them with meaning. We must learn how to make our virtual spaces speak.

## REFERENCES


[1] Carey, Rick, and Gavin Bell. *The Annotated VRML 2.0 Reference Manual*. Reading, MA: Addison-Wesley, 1997.

[2] Chomsky, Noam. *Language and Mind*. Enlarged ed. New York: Harcourt Brace Jovanovich, 1972.

[3] Gregory. R. L. *The Intelligent Eye*. London: Weidenfeld and Nicolson, 1970.

[4] Halliday, M.A.K., and Ruquiya Hasan. *Language, Context, and Text: Aspects of Language in a Social-Semiotic Perspective*. 2nd ed. Oxford: Oxford UP, 1989.

[5] Hayles, N. Katherine. "Embodied Virtuality: Or How to Put Bodies Back into the Picture." In Moser, 1-28.

[6] Kress, Gunther, and Theo van Leeuwen. *Reading Images: The Grammar of Visual Design*. London: Routledge, 1996.

[7] Laurel, Brenda. *Computers as Theatre*. 1991; rpt., Reading, MA: Addison-Wesley, 1993.

[8] Matsuba, Stephen N. "Construct(s) and Même-ing: An Essay in Three Dimensions." 1996. http://ece.uwaterloo.ca/~smatsuba/constructs.

[9] Morse, Margaret. "*Nature Morte*: Landscape and Narrative in Virtual Environments." In Moser, 195-232.

[10] Moser, Mary Anne, with Douglas Macleod. *Immersed in Technology: Art and Virtual Environments*. Cambridge, MA: MIT Press, 1996.

# The Twente Virtual Theatre Environment: Agents and Interactions

Anton Nijholt (Parlevink Research Group)[1]
Centre for Telematics and Information Technology (CTIT)
University of Twente, PO Box 217
7500 AE Enschede, the Netherlands
anijholt@cs.utwente.nl

## Abstract

In this paper we discuss our research on interaction in a virtual theatre. It has been built using VRML and therefore can be accessed through Web pages. In the virtual environment we employ several agents. The virtual theatre allows navigation input through keyboard and mouse, but there is also a navigation agent which listens to typed input and spoken commands. Feedback of the system is given using speech synthesis. We also have an information agent which allows a natural language dialogue with the system where the input is keyboard-driven and the output is both by tables and template driven natural language generation. In development are several talking faces for the different agents in the virtual world. At this moment an avatar with a cartoon-like talking face driven by a text-to-speech synthesizer provides users with information about performances in the theatre. We investigate how we can increase the user's commitment to the environment and its agents by providing context and increasing the user's feeling of 'presence' in the environment. Societal and ethical implications of VR environments are discussed. Moreover, we spend some notes on real-time and video performances in our virtual theatre.

**Keywords:** Virtual Reality, Talking Faces, Text-to-Speech Synthesis, Agent Technology, Speech Recognition, Presence, Societal Aspects, Electronic Commerce, Web Theatre

## 1 Introduction

World Wide Web allows interactions and transactions through Web pages using speech and language, either by inanimate or live agents, image interpretation and generation, and, of course the more traditional ways of presenting explicitly pre-defined information by allowing users access to text, tables, figures, pictures, audio, animation and video. In a task- or domain-restricted way of interaction current technology allows the recognition and interpretation of rather natural speech and language in dialogues. However, rather than the current two-dimensional web-pages, the interesting parts of the Web will become three-dimensional, allowing the building of virtual worlds inhabited by interacting user and task agents, and with which the user can interact using different types of modalities, including speech and language interpretation and generation. Agents can work on behalf of users, hence, human computer interaction will make use of 'indirect management', rather than interacting through direct manipulation of data by users.

In this paper we present our research on the development of an environment in which users can display different behaviors and have goals that emerge during the interaction with this environment. Users who, for example, decide they want to spend an evening outside their home and, while having certain preferences, cannot say in advance where exactly they want to go, whether they first want to have a dinner, whether they want to go to a movie, theatre, or to opera, when they want to go, etc. During the interaction, both goals, possibilities and the way they influence each other become clear. One way to support

[1] This paper describes research carried out in the Parlevink research group of the CTIT (Centre of Telematics and Information Technology) of the University of Twente. The following members of the group have contributed to this paper: Joris Hulstijn, Arjan van Hessen, Hendri Hondorp, Danny Lie, Mathieu van den Berk and Boris van Schooten.

such users is to give them different interaction modalities and access to multimedia information. We discuss a virtual world for representing information and allowing natural interactions that deal with an existing local theatre, and of course, in particular, the performances in this theatre. The interactions between user (visitor) and the system take place using different task-oriented agents. These agents allow mouse and keyboard input, but interactions can also take place using speech and language input. In the current system both sequential and simultaneous multi-modal input is possible. There is also multi-modal (both sequential and simultaneous) output available. The system presents its information through agents that use tables, chat windows, natural language, speech and a talking face. At this moment this talking face uses speech synthesis with associated lip movements. Other facial animations are possible (movements of head, eyes, eyebrows, eyelids and some changes in face color), but at this moment these possibilities have not yet been associated with utterances of user or system.

It is discussed how our virtual environment can be considered as an interest community and it is shown what further research and development is required to obtain an environment where visitors can retrieve information about artists, authors and performances, can discuss performances with others and can be provided with information and contacts in accordance with their preferences. In addition, but this has not been realized yet, we would like to offer our virtual environment for others to organize performances, meetings and to present video art, but also for experiments on mediated communication between visitors and for performances, done by avatars with or by avatars without user participation. The virtual environment we consider is web-based and the interaction modalities that we consider confine to standards that are available or that are being developed for world wide web.

## 2 HISTORY AND MOTIVATION

Some years ago, the Parlevink Research Group of the University of Twente started research and development in the area of the processing of (natural language) dialogues between humans and computers. This research led to the development of a (keyboard-driven) natural language accessible information system (SCHISMA), able to inform users about theatre performances and to allow users to make reservations for performances. The system made use of the database of performances in the local theatres of the city of Enschede. The system is rather primitive. However, if a user really wants to get information and has a little patience with the system, he or she is able to get this information. A more general remark should be given: When we offer an interface to the general audience to access an information system, do we want to offer an intelligent system that knows about the domain, that knows about users, their preferences and other characteristics, etc., or do we assume that any user will adapt to the system that is being offered? Clearly, the latter point is extremely important. It has to do with group characteristics (men, women, old, young, naive, professional, experienced, etc.), but also with facilities and alternatives provided by the owner of the system. As an example, consider the Dutch public transport and railway information system. Human operators are available to inform about times and schedules of busses and trains. However, the number of operators is insufficient. Callers can wait (and pay for the minutes they to wait) or choose for a computer-operated system to which they can talk in natural speech, but possibly have to accept that they need more interactions in order to get themselves understood. Hence, it really depends on the application and the users involved (do they want to pay for the services, do they want to adapt to the interface, does the provider offer an alternative, etc.), whether we can speak of a successful natural language accessible dialogue information system.

We do not really disagree with a view where users are expected to adapt to a system. On the other hand, wouldn't it be much more attractive (and interesting from a research point of view) to be able to offer environments, e.g. worldwide web, where different users have different assumptions about the available information and transaction possibilities, have different goals when accessing the environment and have different abilities and experiences when accessing and exploring such an environment? We like to offer a system such that we can stimulate and expect users to adapt to it and find effective and efficient ways to get or get done what they want.

## 3 EXP

Wher one r moda syste short moda be al shoul in or

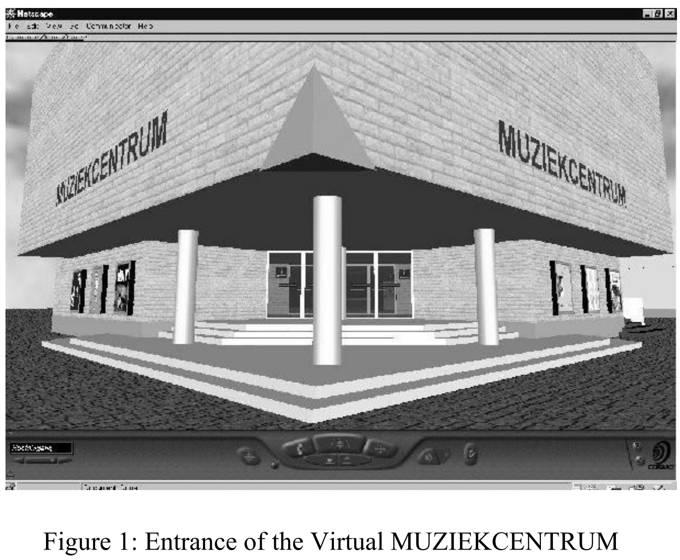



Figure 1: Entrance of the Virtual MUZIEKCENTRUM

communication devices that are currently available for information access, exploration of information and for transaction allow more than one modality for input or output. This is especially true if we look at world wide web interfaces. Research done on information access and transaction in the context of modalities (and especially the sequential and simultaneous combination of modalities), that is, a multi-modality approach for WWW, can be embedded in the attempts to develop standards for access to the web and presentation on the web. For example, standards are being developed for speech access (voice browsing), 3D visualization (virtual reality modeling languages) and the combination of access and visualization (MPEG standards).

When we look at multi-modal human-computer interaction it is clear that hardly any research has been done to distinguish discourse and dialogue phenomena, let alone to model them, for multi-modal tasks. The same holds for approaches to funnel information conveyed via multiple modalities into and out of a single underlying representation of meaning to be communicated (the cross-media information fusion problem). Similarly, on the output side, there is the information-to-media allocation problem.

Our second observation, certainly not independent from the observation above on modalities for access, exploration and presentation, deals with the actors in a system that has to deal with presenting information, reasoning about information, communicating between actors in the system and realizing transactions (e.g. through negotiation) between actors in the system. In addition to a multi-modality approach, there is a need for a multi-agent approach, where agents can take roles ranging from presenting windows on a screen, reasoning about information that might be interesting for a particular user, and being recognizable (and probably visible) as being able to perform certain tasks.

Both multi-modality and multi-agent technology can be considered from a cognitive science point of view, an artificial intelligence point of view or a computer science (i.e., design, algorithmic & data structures) point of view.

At this moment the cognitive science point of view, at least at our side, is rather undeveloped. The ideas that are available on the cognitive science point of view deal with syntax, semantics and pragmatics of natural language communication. That is, although we would like to see it differently, they are more closely related to linguistics than to cognition science in general. On the other hand, some modest approaches to include concepts of cognitive science in the definition and the behavior of agents are available and cognitive ergonomics helps to design user interfaces and interaction modalities for given tasks and users.

From the artificial intelligence point of view we know we can use results on domain-independent and domain-dependent representation and reasoning. Frame- and script-based methods in AI are available and compromises have been established between cognitive science, artificial intelligence and computer science, in order to design and develop useful applications. From the computer science point of view we can discuss methods for design, specification and implementation of multi-modal and multi-agents systems. In the next sections we will return to these topics. The roles of speech, language and visualization will be emphasized.

## 4 PROVIDING CONTEXT: VISUALIZATION

We decided to visualize the environment in which people can get information about theatre performances, can make reservations and can talk to theatre employees and other visitors. VRML, agent technology, text-to-speech synthesis, talking faces, speech recognition, etc., became issues after taking this decision. They will be discussed in the next sections. Visualization allows users to refer to a visible context and it allows the system to disambiguate user's utterances by making use of this context. Moreover, it allows the system to influence the interaction behavior of the user in such a way that more efficient and natural dialogues with the system become possible.

Our virtual theatre has been built according to the design drawings made by the architects of a local theatre. Part of the building has been realized by converting AutoCAD drawings to VRML97. Video recordings and photographs have been used to add 'textures' to walls, floors, etc. Sensor nodes in the virtual environment activate animations (opening doors) or start events (entering a dialogue mode, playing music, moving spotlights, etc.). Visitors can explore the environment of the building, hear the carillon of a nearby church, look at a neighboring pub and movie theatre, etc. and they can enter the theatre (cf. Figure 1) and walk around, visit the hall, admire the paintings on the walls, go to the balconies and, take a seat in order to get a view of the stage from that particular location. Information about today's performances is available on a notice board that is automatically updated using information from the database with performances. In addition, as may be expected, visitors may go to the information desk in the theatre, see previews and start a dialogue with an information and transaction agent called 'Karin'. The first version of Karin looked like other standard avatars available on World Wide Web. The second version, available in a recent prototype of the system, makes use of a 3D talking face.

One may dispute the necessity of this realistic modeling of theatre and agents, the environment and the information and transaction services. We have taken the point of view that (potential) visitors are interested in or are already familiar with the physical appearance of this theatre. Inside the virtual building there should be a mix of reality (entrance, walls, paintings, desks, stages, rooms, etc.) and new, non-traditional, possibilities for virtual visitors to make use of interaction, information, transaction and navigation services that extend the present services of the theatre.

It has become clear from several studies (cf. Friedman [9]) that people engage in social behavior toward machines. It is also well known that users respond differently to different 'computer personalities'. It is possible to influence the user's willingness to continue working even if the system's performance is not perfect. They can be made to enjoy the interaction, they can be made to perform better, etc., all depending on the way the interface and the interaction strategy have been designed. It makes a difference to interact with a talking face display or with a text display. People behave differently in the presence of other people than they do when they are alone. In experiments it has been shown that people display different behavior when interacting with a talking face than they do with a text-display interface. This behavior is also influenced by the facial appearance and the facial expressions that are shown. People tend to present themselves in a more positive light to a talking face display and they are more attentive when a task is presented by a talking face (cf. Sproull et al. [28]).

From these observations we conclude that introducing a talking face can help to make interaction more natural and to make shortcomings of the technology more acceptable to users.

The use of speech technology in information systems will continue to increase. Most currently installed information systems that work with speech, are telephone-based systems where callers can get information by speaking aloud some short commands. Also real dialogue systems wherein people can say normal phrases become more and more common, but one of the problems in this kind of systems is the limitation of the context. As long as the context is narrow they perform well, but wide contexts are causing problems. One reason to introduce task-oriented agents is to restrict user expectations and utterances to the different tasks for which agents are responsible. Obviously, this can be enhanced if the visualization of the agents helps to recognize the agents tasks.[2]

## 5 PROVIDING CONTEXT: COMMUNICATION

In the previous subsections we have looked at possibilities for users to access information, to communicate with agents designed by the provider of the information system and to explore an environment with the goal to find information or to find possibilities to enter into some transaction. It is also interesting to investigate how we can allow communication between users or visitors of a web-based information and transaction system. For that purpose it is useful to look at experiences with web-based digital cities, chat environments and interest communities.

Web-based digital cities have been around for some years. Like computer games they have evolved from text environments to 2-dimensional graphical and 3D virtual environments with sounds, animation and video. Visitors, or maybe we should call them residents, of these cities visit libraries, museums, pubs, squares, etc., where they can get information, chat with others, etc. In these environments people get the feeling of being together. They are listening to each other and, in general, they take responsibility for the environment


Figure 3: Karin at the Information Desk

period, have the same therapy, belonging to the same political party, etc.).

As an example we mention a virtual world developed by the virtual worlds group of Microsoft in co-operation with The Fred Hutchinson Cancer Research Center in Seattle. This so-called “Hutch World” enables people struggling with cancer to obtain information and interact with others facing similar challenges. Patients, families and friends can enter the password protected three-dimensional world (a rendering of the actual outpatient lobby), get information at a reception desk, visit a virtual gift shop, etc. (Figure 2). Each participant obtains an avatar representation. Participants can engage in public chat discussions or invitation-only meetings. A library can be visited, its resources can be used and participants can enter an auditorium to view presentations. Part of the project consists of developing tools to create other applications.

# 6 AGENTS IN THE TWENTE VIRTUAL THEATRE

## 6.1 AN AGENT PLATFORM IN THE VIRTUAL ENVIRONMENT

In the current prototype version of the virtual theatre we distinguish between different agents: We have an information and transaction agent, we have a navigation agent and there are some agents under development. An agent platform has been developed in JAVA to allow the definition and creation of intelligent agents. Users can communicate with agents using speech and typed dialogue. Any agent can start up other agents and receive and carry out orders of other agents. Questions of users can be communicated to other agents and agents can be informed about each other’s internal state. Both the information & transaction agent and the navigation agent are in the platform. But also the information board, presenting today’s performances, has been introduced as an agent. And so can other objects in the environment.

## 6.2 THE INFORMATION & TRANSACTION AGENT

Karin, the information/transaction agent, allows a natural language dialogue with the system about performances, artists, dates, prices, etc. Karin (Figure 3) wants to give information and to sell tickets. Karin is fed from a database that contains all the information about performances in our local theatre. Developing skills for Karin, in this particular environment, is one of the aims of our research project. This research fits in a context of much more general 'intelligent' (web-based) information and transaction services.

Our current version of the dialogue system of which Karin is the face is called THIS v1.0 (Theatre Information System). The approach used can be summarized as rewrite and understand. User utterances are simplified using a great number of rewrite rules. The resulting simple sentences are parsed. The output can be interpreted as a request of a certain type. System response actions are coded as procedures that need certain arguments. Missing arguments are subsequently asked for. The system is modular, where each 'module' corresponds to a topic in the task domain. There are also modules for each step in the understanding process: the rewriter, the recognizer and the dialogue manager. The rewrite step can be broken down into a number of consecutive steps that each deal with particular types of information, such as names, dates and titles. The dialogue manager initiates the first system utterance and goes on to call the rewriter and recognizer process on the user's response. Also, it provides an interface with the database management system (DBMS). Queries to the database are represented using a standard query language like SQL. Results of queries are represented as bindings to variables, which are stored in the global data-structure, called context. The arguments for the action are dug out by the dedicated parser, associated with the category. All arguments that are not to be found in the utterance are asked for explicitly. More information about this approach can be found in Lie et al [15].

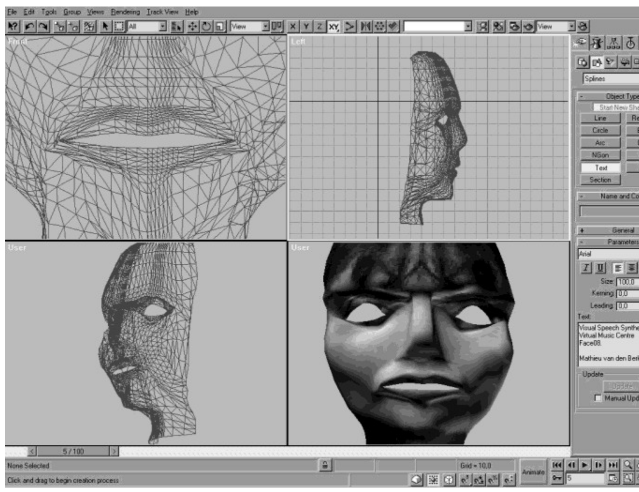


Figure 4: 3-D Face Development

Presently the input to Karin is keyboard-driven natural language and the output is both screen and speech based. In development is an utterance generation module. Based on the most recent user utterance, on the context and on the database, the system has to decide on a response action, consisting of database manipulation and dialogue acts.

## 6.3 THE NAVIGATION AGENT

The WWW-based virtual theatre we are developing allows navigation input through keyboard and mouse. Such input allows the user to move and to rotate, to jump from one location to an other, to interact with objects and to trigger them. In addition, a navigation agent has been developed that helps the user to explore the environment and to interact with objects in this environment by means of speech commands. A smooth integration of the pointing devices and speech in a virtual environment requires has to resolve deictic references that occur in the interaction. The navigation agent should be able to reason about the geometry of the virtual world in which it moves. The current version of the navigational agent is not really conversational. Straightforward typed commands or similar speech commands make it possible for the user to explore the virtual environment. Navigation also requires that names have to be associated with the different parts of the building, the objects and the agents, which can be found inside of it. Clearly, users may use different words to designate them, including implicit references that have to be resolved in a reasoning process.

Speech Recognition on local machines is pretty good, but speech recognition on the World Wide Web results in various problems. Many of these problems are caused by the lack of standards and the lack of interest of big companies (providing operating systems, WWW browsers and Virtual Reality languages and environments) to cooperate in order to establish standards. When we confine ourselves to speech recognition, we distinguish between two approaches.

- First Solution: Every user should have a speech recognition engine that can recognize their commands and send this information to the server system. However, good speech recognition systems are very expensive, require large resources and bad systems result in badly recognized commands.
- Second Solution: Another solution would be to have the speech recognition on the server side. This requires the recording of commands on the client side and a robust transporting of the audio files.

Figure 5: Cartoon Face

In our system we have chosen for the second solution. It does not require users to install speech recognition software or to download a speech recognition module as part of the virtual world from the server. They do need audio-software which is usually available anyway.

# 7 SPEECH GENERATION AND ANIMATION

## 7.1 SPEECH GENERATION THROUGH TEMPLATES

In the design of utterance generation by the information agent a list of utterance templates is used. Templates contain gaps to be filled with information items: attribute-value pairs labeled with syntactic and lexical features. Templates are selected on the basis of five parameters: utterance type, the body of the template and possible empty lists of information items that are to be marked as given, wanted and new. The utterance type and body determine the word-order and the main intonation contour. The presence and number of information items in the given, wanted and new slots, as well as special features affect the actual wording and intonation of the utterance.

For pronouncing the utterance templates we use the Fluent Dutch Text-to-Speech system (Dirksen [7]). Fluent Dutch runs on top of the MBROLA diphone synthesizer (Dutoit [8]). It uses a Dutch voice, developed at the Utrecht institute of linguistics (OTS). Fluent Dutch operates at three levels: the grapheme level, the phoneme level and a low-level representation of phones where the length and pitch of sounds is represented. For many words, the phonetic description is taken from lexical resources of Van Dale dictionaries. Other prosodic information is derived by heuristic rules. It is possible to manipulate prosody by adding punctuation at the grapheme level, by adding prosodic annotations at the phoneme level or by directly manipulating the phone level.

## 7.2 FACING THE INFORMATION AGENT

We developed a virtual face in a 3D-design environment (cf. Figure 4). The face consists of various three-dimensional coordinates and is connected through faces. These faces are shaded to visualize a three-dimensional virtual face. The 3D data is converted to VRML-data that can be used for real-time viewing of the virtual face. A picture of a real human face can be mapped onto the virtual face. We are researching various kinds of faces to determine which can be best used for this application. Some are rather realistic and some are more in a cartoon-style (cf. Figure 5). The face is the interface between the users of the virtual theatre and the theatre information system. A dialogue window is shown when users approach the information-desk while they are navigating in the virtual theatre.

The face is capable of visualizing the speech synchronously to the speech output. This involves lip-movements according to a couple of visemes. The face has to visualize facial expressions according to user's input or the system's output (see further sections on facial features). Figure 6 represents the architecture of the visual speech system.

The last element in the chain of this figure (the VRML-browser) is also the first element. We use Cosmo Player, which is a plug-in for an HTML-Browser, for viewing VRML-files. These files are specifications of a three-dimensional virtual environment. The whole virtual theatre is a collection of VRML files, which can be viewed by the browser. As mentioned earlier, the user will see a virtual face when the information desk is approached. A dialogue window, the JAVA Schisma applet, is available for the user to formulate questions or to give answers to the system's questions. The user types the questions on a keyboard in Dutch sentences. The answers to the questions are to be determined on the server side: the Schisma server. Answers or responding questions are passed to the JAVA Visual Speech Server Application on the server side.

This application filters the textual output of the dialogue system in parts that are to be shown in a table or a dialogue window and parts that have to be converted to speech. The parts that are to be shown in the dialogue window or a table, like lengthy descriptions of particular shows or lists of plays are send to the Schisma Client Applet where they are showed on the screen. The parts of the Schisma output that are to be spoken by the virtual face are converted to speech with the Text-to-Speech Server. The input is the raw text and the output is the audio file of this spoken text and information about the phonemes in the text and their duration.

For example, the Dutch word for "speech generation" is "spraakgeneratie". This word contains the following phonemes: S p r *a k x e n @ r a t s I. When the resulting audio file is played, each phoneme has it's own duration. This information is gathered from the TTS-server:

s 79 p 71 r 38 a 106 50 127 k 53 x 90 e 113 20 102 n 60 @ 38 r 53 a 101 t 23 s 113 I 119 20 75

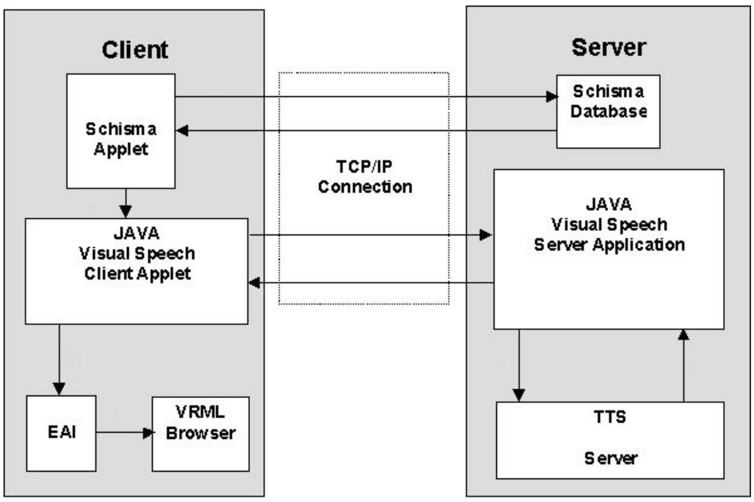
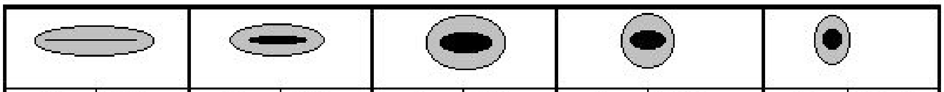





Figure 6: Architecture Visual Speech System



| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| p | Pot | D | Dak | t | Tak | S | Sjaal | f | Feest |
| b | Bak | k | Koe | tj | Tjarda | H | Huis | v | Veel |
| m | Mok | g | Goal | dj | Djoerd | j | Jas | w | Wel |
| | | n | Nat | z | Zon | J | Aai | W | Eeuw |
| | | nj | Ranja | s | Sla | y | Fuut | | |
| | | i | Pit | Z | Rage | 2 | Keus | | |
| | | E | Pet | G | Geven | Oe | Freule | | |
| | | Y | Put | x | Chaos | Y: | Keur | | |
| | | A | Pad | N | Bang | O: | Koor | | |
| | | O | Pot | l | Lat | E~ | Timbre | | |
| | | @ | Geval | L | Kaal | A~ | Chanson | | |
| | | i | Niet | a | Taak | Y~ | Parfum | | |
| | | R | Paar | e | Keet | O~ | Bonbon | | |
| | | r | Raak | o | Pook | | | | |
| | | u | Boek | I: | Keer | | | | |
| | | | | Ei | Tijd | | | | |
| | | | | 9y | Huis | | | | |
| | | | | Au | Koud | | | | |

Figure 7: Phoneme Classes and Visemes

The characters are the phonemes and the first number after the characters are durations of the corresponding phonemes in milliseconds. If more numbers follow then the first number is a percentage of the whole duration in which the pitch of the voice changes to the following number. So the first 'a' is spoken for 106 milliseconds and on 50% of this 106 milliseconds the pitch changes to 127 Hz. The previously described information from the TTS-server will be sent to the JAVA Visual Speech Client Applet together with the audio file. The Visual Speech Client Applet uses the phoneme information to map the phonemes onto different mouth states or visemes. All the phonemes are categorized in five visemes (cf. Figure 7).

When the audio file is loaded on the client side, the mouth states and their durations are passed to the External Authoring Interface (EAI). This is an interface between JAVA and the VRML browser. This interface triggers animations in the virtual environment. It starts the sound playback and all the corresponding animations. Only the mouth states are specified in the VRML-file. The animation is done by interpolating between mouth states in the given amount of time. This results in reasonable smooth lip-movements.

## 7.3 Prosody, Facial Expressions and Emotions

How do we control the responses of the system, the prosody and the artificial face? The central module of a dialogue system is called the *dialogue manager*. The dialogue manager maintains two data-structures: a representation of the *context* and a representation of the *plan*, the current domain-related action that the system is trying to accomplish. Based on the context, the plan and a representation of the latest user utterance or signal, such as a pointing gesture, the dialogue manager selects a certain response action. Planning and action selection are based on a set of principles, called *dialogue rules*. A response action is a combination of basic domain related actions, such as database queries, and dialogue acts to convey the results of the query. Dialogue acts describe the intended meaning of an utterance or gesture. The *response generation* module selects a way to express it. It determines the utterance-structure, wording, and prosody of each system response. Now, it should also control the orientation and expression of the face, the eyes, and the coordination of sounds and lip movement. What parameters are needed to control response generation?

## 7.4 Prosodic Features

The spoken utterance generation module uses a set of parameters to control prosodically annotated utterance templates. Templates contain gaps to be filled with *information items*: attribute-value pairs labeled with syntactic, lexical and phonetic features. An appropriate template for a given dialogue act is selected by the following parameters: *utterance type*, *body* of the template, *given* information, *wanted* and *new* information. The utterance type and body determine the word-order and main intonation contour. The given, wanted and new slots, as well as special features, affect the actual wording and prosody. Templates respect rules of accenting and de-accenting. As a rule, information that is assumed to be given in the dialogue is de-accented, expressed as a pronoun, or even left out. Given information is repeated whenever the system is not confident it was recognized correctly by the speech recognition module. Such verification prompts are distinguished by a rising intonation. Information that is to be presented as new, is accented. Quoted expressions, like artist names or titles of performances, are set apart from the rest of the utterance. For reading the texts and reviews that describe the content of performances, the system assumes a 'reading voice'.

| **Feature** | **Manipulation** | **Meaning** |
|---|---|---|
| Eyes | Gaze direction | Idle behavior, attention, indexing |
| Eyebrows | Lift, lower | Surprise, distress, angry |
| Lips | Form visemes<br>Stretch, round | Talk<br>Smile, laugh, neutral, angry, kiss |
| Mouth shape | Stretch, round | Smile, neutral, angry |
| Color | Blush | Shyness, embarrassment |
| Head | Orientation<br>Idle behavior<br>Movement frequency | Nodding, shaking head, attention<br>Neutral<br>Emotional 'volume' |
| Shoulders | Shrug | Indifference |

Figure 8: Facial Features

## 7.5 FACIAL FEATURES

Apart from the lips, the virtual face has a number of dynamic control parameters (Figure 8).

The *eyes* can gaze at a certain direction. This can be used to direct attention towards an area. The *eyelids* may be opened and closed, for blinking. The *eyebrows* can be lifted to indicate surprise or lowered for distress. The shape of the *mouth* can be manipulated into a smile or and angry expression. The *color* of the face can be deepened, to suggest a blush that indicates shyness or embarrassment. The *orientation* of the head can be manipulated, leaning forward and backward or tilting left and right. This may produce important facial gestures like nodding and shaking one's head. It can also be used to indicate attention; leaning forward means being interested, leaning backward means loosing interest. In general the character is not still. The head will wiggle a bit and its eyes will wonder. This is called *idle behavior*. Many existing 'talking heads' look artificial because of their stillness. Moreover, not moving can also be taken as a sign. For instance, Clermont et al. [5] found that a fixed stare indicates a misunderstanding in the dialogue. The *frequency* of idle movements is an indicator of the liveness of the character; it serves as a type of volume, to the existing emotion. So, many random movements of the head, combined with smiles and attentive eyes, indicate a very happy personality; stillness, a neutral mouth shape and looking away, indicate a withdrawn and unhappy personality. But an angry face, combined with a blush and a lot of movement, indicate increased anger (see Figure 9). Jerky movements with wondering eyes indicate nervousness. Since our agent is supposed to be professionally friendly, she will be generally smiling and will have a moderate movement frequency.

Each of these basic features can be combined into facial *gestures* that can be used to signal something. Gestures like nodding, shaking and shrugging can be used separately, but often utterances are combined with gestures or utterance related facial expressions. The timing of the gesture or the expression must be aligned with the uttera

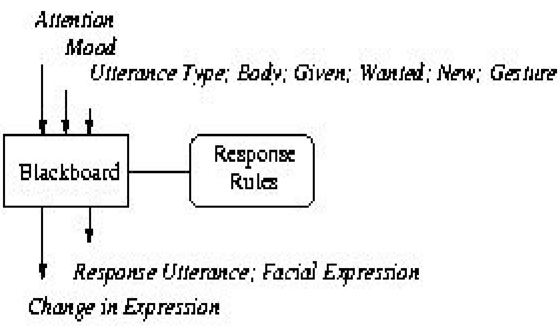



Figure 10: Blackboard Architecture

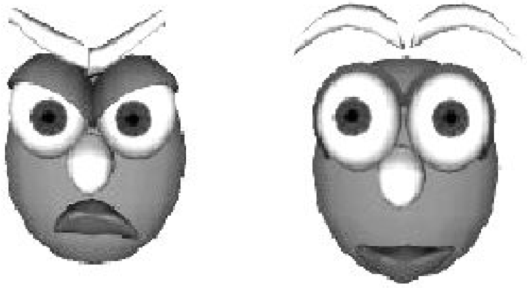


Figure 9: Angry and Uncertain

nce. We use the following general heuristic for alignment of gestures.

Like any event, an utterance and a gesture have an *entry* and an *exit* point. Moreover, an utterance can be broken down into phrases; each phrase has a so called *intonation center*, the moment where the pitch contour is highest. Since pitch accents are related to informativeness, we can assume that the accent lands on the most prominent expression. Usually the accent lands towards the end of an utterance. Similarly, each gesture has a *culmination point*. For instance for pointing, the moment that the index finger is fully extended. The visual animator extrapolates a nice curve from the entry point to the culmination and again to the exit point. Our current working hypothesis is that gestures synchronize with utterances, or precede them. So we link the gesture's entry and exit points to the entry and exit points of the utterance and make sure that the culmination point occurs before or on the intonation center.

So how do we control this wealth of features? We propose a blackboard architecture, as depicted in Figure 10. Combinations of input parameters trigger a rule that produces a response action, or a more permanent change of expression. The reason for a blackboard architecture is that the parameters influence each other. Roughly, there are two types of facial behavior that need to be modeled.

Firstly, permanent features like the facial expression, gazing direction and general movement characteristics, both when speaking and when idle. These can be controlled by two parameters: *mood* and *attention*. The *mood* parameter indicates the general attitude of the personality in the conversation. It is a state, that extends over a longer period. Is the agent happy, sad, angry or uncertain? The *attention* parameter controls the eyes and gazing direction. We believe that one of the benefits of a talking face is that turn taking and attention management in dialogues will be made easier. The gazing direction of the eyes and the head position are crucial for this (Vertegaal [30])[3]. Usually mood and attention are fixed for a given personality. Temporary changes in emotion and attention, may result from previous utterances or to the general conversation. For instance, anger at an insult, or increased interest after a misunderstanding.

Secondly, utterance related attitudes. Since we cannot monitor the user's utterances in real-time, at the moment this is limited to system utterances only. Think of smiling at a joke, raising eyebrows at a question or a pointing gesture at an indexical. Conventional gestures can be modeled as a special instance of response actions. Nodding or shrugging are coded like any other utterance synchronized with a gesture, except that they can be silent. Utterance related features are controlled by the existing utterance parameters, extended with a new parameter, *gesture* , that labels one or more facial movements to be synchronized with the utterance template. Because we know all utterance templates in advance, the synchronization can be manually adjusted if needed. The extend of the gesture and its final shape also depend on the general emotional state and attention level.

We also hope to introduce some variation in the exact choice of movement. Variation is important. For instance, it is natural to combine "yes" with a nod, but when every affirmative is combined with the same nod it looks mechanical. Another example is the raising of eyebrows. In an early version of the system the eyebrows were directly controlled by pitch level. Thus, the agent would nicely express uncertainty on a question, which has a rising intonation. But of course, pitch is also used for accenting. So the agent wrongly expressed surprise at expressions that were stressed. Synchronizing the apparently random movements with fixed speech from templates is difficult. We have chosen to align the culmination points of the movement with the intonational centers of the phrases uttered. But the exact frequency and extent of the movements will be randomly distributed, partly based on mood and attention.

## 8 Speech, Animation & Web-based Streaming Audio and Video

At this moment VRML97 is the standard specification for VRML. This specification allows the definition of AudioClip Nodes in a VRML world. AudioClip Nodes have stereometric properties, that is, the volume of sound increases when approaching a sound object and when a user moves in the world the sound will adapt. Hence, when the user moves to the right the volume in the left speaker will increase and the volume in the right speaker will decrease. The present standard

[3] Experiments with different video conferencing environments have shown that gaze information is more important to a smooth conversation, than a simple television type image of the talking person, when it is unclear what he or she is looking at.

AudioClip Nodes are uncompressed WAV and MIDI, respectively.

Our experiences until now show that the use of uncompressed WAV slows down animation considerably (often 30 seconds or more) because the WAV file has to be written to the hard disc by the TTS Server and both the EA (External Authoring) Interface and the VRML browser have to read this file completely before the animation can be started. Short sentences hardly cause problems, but long texts often take 300 kB or more.

It is investigated whether it is possible to process the audio output of the TTS Server in such a way that a compressed audio stream can be created that can be synchronized with the VRML animation. When we look at RealAudio compression/streaming there are the following disadvantages:

- For the VRML browser this is not a standard format. However, maybe we can deal with this since it is possible to add unknown Nodes to a VRML world using properties of existing Nodes. In his way it is possible to define a RealAudio Node which can be driven with the help of RJMF (RealAudio Java Media Framework).
- The VRML browser claims the full Audio System of the PC if it has to use WAV files; therefore it is not possible for applications outside the VRML world to play audio fragments. This would mean that all WAV files to be used in the VRML world have to be played as mentioned above.
- If outside the VRML world an Audio file is played then the stereometric properties of the sound can not be modeled in the VRML world. A partial solution can be obtained since the EAI can observe changes of position and the RJMF can change the volume of the sound. Whether it allows implementation of stereo sound is not clear yet.

Roehl [25] discusses audio streaming in VRML. He argues for a standard way for a content creator to indicate to the browser that the data in an AudioClip should be streamed, rather than being completely downloaded prior to being presented to the user. Moreover, whenever possible, we should use existing open standards. Examples are RTSP, SMIL and RTP.

RTSP (Real Time Streaming Protocol (RFC 2326)) is an existing draft Internet standard for accessing streaming media. The content creator is able to identify which data should be streamed by specifying “rtsp:” as the scheme in the URLs instead of “http:” or “ftp:”, so the browser should use RTSP to obtain data for that node. RTSP does not specify the use of any particular transport mechanism for the actual streaming data itself. RTP (the Real Time Protocol (RFC 1889)) does. It is an application level protocol for the transport of streaming multimedia. Synchronization of the audio and video data can be achieved using the timestamp information provided in the RTP headers. Part of the RTP standard is a separate protocol RTCP (Real Time Control Protocol) which, among other things, provides NTP based timestamps for the purpose of synchronizing multiple media streams.

An important recent development is SMIL (Synchronized Multimedia Integration Language), a proposed World Wide Web Consortium Standard. SMIL is HTML-like and describes multiple sequential or concurrent media streams along with timing information. Hence, it allows the synchronization of Audio/Video files with other events. We have to investigate how VRML fits in this development and we plan to investigate whether it is possible to generate SMIL information from the phoneme output of the Text-to-Speech Server. Together with the RTP and RTSP transport mechanisms it should be possible to obtain exact synchronization with the help of the timestamps in NTP.

In Osterman et al. [20] it is explained how MPEG4 can specify a set of face animation parameters and a sound output interface such that phonemes and their timing (obtained from the TTS) can be translated into sequences of facial animation parameters and there is synchronization between speech and animation. The TTS system is explained in Beutnagel et al [4]. We have not yet investigated their approaches in detail.

# 9 FORMAL MODELING OF INTERACTIONS IN VIRTUAL ENVIRONMENTS

Both from an ergonomical and a software-engineering viewpoint, the design of interaction in virtual environments is complex. Virtual environments may feature a variety of interactive objects, agents which may use natural language to communicate, and multiple simultaneous users. All may operate in parallel, and may interact with each other concurrently. Next to this, the possibility of using Virtual Reality techniques to enhance the experience of virtual worlds offers new ways of interaction, such as 3D navigation and visualization, sound effects, and speech input and output, possibly used so as to complement each other.

One new line of research we have taken is an attempt to address both of these issues by means of a formal modeling technique that is based on the process algebra CSP (Communicating Sequential Processes).

For that reason, in our virtual theatre a simplified flow of interaction has been specified, showing all relevant interaction options for any given point in time. The system architecture has been modeled in an agent-oriented way, representing all system- and user-controlled objects, and even the users themselves, as parallel processes. The interaction between processes is modeled by signals passing through specific channels. Interaction modalities (such as video versus audio and text versus graphics) may also be modeled as separate channels.

This modeling technique has some strong points. Firstly, and most generally, such a simplified and formal model enables a clear and unambiguous specification of system architecture and dynamics. Secondly, it may be useful as a conceptual model, modeling the fact that a user experiences interaction with other users and agents in a similar way than in a completed system, and explicitly showing which options are available when and through which modalities. Thirdly, it enables automatic prototyping, such as architecture visualization and verification of some system properties.

For more details about this approach we refer to Schooten et al. [26]. There it is also shown how a CSP description can be coupled to a simplified user interface and executed, so that the specified system can be tried out immediately. Specifications map closely to software architecture, reducing the cost of building a full prototype.

## 10 THE ROLE OF PRESENCE

'Presence', as defined by Lombard and Ditton [16], is the perceptual illusion of nonmediation, that is, 'a person fails to perceive or acknowledge the existence of a medium in his/her communication environment and responds as he/she would if the medium were not there.' They mention that this illusion can occur in two distinct ways:

(a) The medium can appear to be invisible, with the medium user and the medium content sharing the same physical environment; and

(b) The medium can appear to be transformed into a social entity.

The authors show that many conceptualizations of presence that appear in the literature are more detailed viewpoints on this distinction. These viewpoints are:

- Social richness. Is the medium perceived as sociable, warm, personal, etc., in the interaction? How does the medium transmit cues of human communication?
- Realism. How real is the experience, how accurate is the representation of objects?
- Transportation. Does the user have the feeling that he/she has been transported to a different place (or the place to the user), does the user have the feeling to share the place with others?
- Immersion. How much are the user's senses immersed in the virtual world, how much is a user involved in the interaction with the medium?
- Social actors within a medium. Do mediated entities play the role of social actors? That is, do users overlook the mediated and artificial nature of entities within a medium with which they interact?
- Medium as social actor. Do users respond to the medium as a social entity? Do they follow social rules in the interaction?

Other definitions of presence exist, often closely related as in Witmer et al. [31] ("presence is defined as the subjective experience of being in one place or environment, even when one is physically situated in another"), sometimes using new names ('telepresence'), sometimes introducing new viewpoints, e.g., in Zahorik et al. [33], where in the tradition of Heidegger presence is strongly related to one's successfully supported action in an environment.

Causes and effects of presence have been studied in experiments, but the number of parameters involved, e.g., task and user characteristics, makes systematic research difficult.[4] We mention Thie et al. [29], who attempt to measure the effects of social virtual presence in a decision making task. Social virtual presence is the feeling that other people are present in the virtual environment. Their hypothesis was that presence will be higher if social presence cues are maximized and that decision task performance improves if presence increases. While the first part of this hypothesis was supported by their experiments, due to lack of good measuring questionnaires, technical malfunctioning, etc., no useful results could be reported about an increase of performance.

From the previous sections it may have become clear that rather naturally emerging topics of our interest are closely related to the issue of presence. The environment that is offered and the locations that can be visited look familiar, the functions of several objects and what to do with them is clear from their

[4] Papers on presence can be found in the journal: *Presence: Teleoperators and Virtual Environments*, MIT Press.

appearance and the multimodality approach allows a variety of user input and the production of different sensory outputs. The agents in the environment are assumed to be friendly and cooperative and the embedding of talking faces and moving avatars in the environment will increase the tendency to treat agents as social actors, rather than as software modules.

However, increasing the user's feeling of presence has not been an explicit goal. Rather we have looked at possibilities to increase a user's commitment to the system (like we would in similar systems, e.g., for electronic commerce) with the aim to obtain co-operative behavior. One obvious reason which makes us loose a user is when use of clumsy technology (like speech and language technology) is not sufficiently backed up by context (including other modalities) which seduces the user to a certain interaction behavior and which helps to disambiguate the users utterances.

Needless to say that for many tasks and activities one may not expect that presence will necessarily enhance performance. On the contrary, presence can be noise which distracts a user from performing a task and which throws away the advantages of individual access from a home PC to, e.g., an information service. There is no need for a virtual waiting queue or the sound of a leaving plane when all we want to know is when the next flight to Amsterdam leaves.

## 11 Societal and Ethical Aspects

Long time ago, in the early eighties, Terry Winograd mentioned that in presenting machines as 'intelligent' we produce an illusion which may be beneficial, may lead to breakdown in the interaction, or may be used by parties to deceive and exploit others. His concern was illustrated with the following quote (from a researcher at a major computer firm):

> *"From my point of view, natural language processing is unethical for one main reason: It plays on the central position which language holds in human behavior. I suggest that the deep involvement Weizenbaum found some people to have with ELIZA, is due to the intensity with which most people react to language in any form. When a person sees a linguistic utterance in any form, the person reacts, much as a dog reacts to an odor. We are creatures of language ..... Since this is so, it is my feeling that baiting people with strings of characters (tokens) clearly intended by someone to be interpreted as symbols, is as much misrepresentation as would be your attempt to sell me property for which you had a false deed. In both cases, an attempt is made to encourage someone to believe that something is a thing other than what it is, and only one party in the interaction is aware of the deception."*

We would like to add a more recent quote (late nineties) by Ben Shneiderman (head of the HCI-Lab of the University of Maryland):

> *"Designers should restrain themselves from exploiting feelings that can be evoked by machines. Computers should not be laden (!) with emotion."*

Interesting it is to observe that in the early eighties the emphasis was on natural language, and in the nineties the emphasis is on feelings and emotions. More interesting for our purposes is to note that in both quotations the authors assume that computers can be used to deceive and exploit users and that that is wrong.

One reaction to that may be: What's wrong about that? Newspapers do, PR departments do, political parties do and advertisements do. What makes computers more special than newspapers that they shouldn't be used for those purposes? However, even if we accept such a view – or understand that we have to live with it - it nevertheless is useful to consider the social and ethical consequences of research that aims at making or presenting computers as intelligent. Clearly, in our research we are developing systems - or agents in systems – that exhibit human behavior. For that reason we would like to give some views on societal and ethical aspects.

In a previous section, when we discussed the role of 'Presence', it became clear that in our world mediated entities (our agents in the virtual world) play the role of social actors, they increase the feeling of presence and they help to increase the user's commitment to the system. A more technical reason to have agents as social actors is that they can influence the interaction behavior of the users in such a way that it stays in a certain domain of task and/or domain knowledge, hiding shortcomings of imperfect interaction technology, in particular speech and language technology. With other words, in order to increase the quality of the web-based information and transaction services we are offering, it seems to be useful to exploit the possibility to increase the role of social actors in our environment.

As already mentioned in Bates [2] (in the context of agents performing in virtual worlds), rather than build smart but narrow agents we can build agents with a more broad though perhaps more shallow capabilities. That is, for several applications we may be able to use agents that are not intelligent, as long as they are not clearly stupid or unreal. That is, *"An agent that keeps quiet may appear wise, while an agent that oversteps its abilities will probably destroy the suspension of disbelief. Thus, we propose to take advantage of the "Eliza effect", in which people see subtlety, understanding, and emotion in an agent as long as the agent does not actively destroy the illusion."*

Now we would like to introduce a second view, presenting support to the quotations given above. For that purpose we want to refer to Friedman and Kahn [10], who give a clear overview of the relation between social actors in computer systems (or computer systems as social actors) and the moral perspective, that is, aren't we introducing relationships between users and computational systems that delegate responsibility from the human user to the computational system? Can computers or entities which become visible in the interface between user and computer held responsible for actions initiated (directly or indirectly) by a user?

In order to answer the question whether a computer can be a moral agent (and thus be held morally responsible for a decision) they argue that this can only be the case if we attribute (human) intentionality to a computer (or an agent in the interface). They take the position that this can not be the case. If so, either the human is reduced to the status of computers, or the computer is raised to the status of humans. Since, in their opinion, nobody has been able to undermine Searle's 'Chinese room' argument, computational systems cannot have intentionality and cannot be held responsible for their decisions and the consequences. Unfortunately, systems can be designed such that the user's sense of his or her moral agency is diminished. This can be done by placing human users into mechanical roles without understanding of their individual actions, or by masquerading the system as an agent with beliefs, desires and intentions. Their conclusion is, humans should not inappropriately attribute agency to computational systems and design practice should discourage this perception.

In order to discourage such a perception, designers should refrain from anthropomorphizing computational systems. Non-anthropomorphic design increases responsible computing. Hence, do not try to model human-human communication and do not (necessarily) try to model human intelligence when designing interaction systems. Rather than blurring the boundaries between people and computers they should be sharpened.

These viewpoints return in the well known debate on 'direct manipulation' vs. 'interface agents' (see e.g. Shneiderman and Maes [27]). While Shneiderman argues that anthropomorphic representations are misleading (destroying the users' sense of accomplishment), Maes has advocated agents that appear on the screen with animated facial expressions and body language.

It should be clear from previous sections that in our environment we decided to explore the possibility to increase the user's feeling of presence in order to increase his or hers commitment to the system. We certainly do not want to advocate such an approach in general. In our application we think this is a useful approach. We don't think this should be the general approach to the design of computer interfaces.[5]

## 12 THEATRE-RELATED PURPOSES OF THE VIRTUAL ENVIRONMENT

In the previous sections we concentrated on:

- the public, that is, men, women, children who want attend a certain performance or who want to know about performances in general, in a certain city or region, and at a certain date or in a certain period; the public has expectations about the information that is provided, it knows, for example, that different newspapers have different opinions about performances, hence, it is necessary to be careful in pushing the visitor to attend a certain performance
- the theatres, that is, the organizations that want to sell tickets, want positive reviews for the performances they hire, want to give correct and relevant information to the public, and have to offer contracts to the managers of artists and theatre companies in such a way that they are not loss-making

However, now that we have a virtual theatre where people can look around and get information on performances, wouldn't it be nice to apply this virtual reality environment to other theater-related purposes? Why not look more closely at possibilities that can be offered to:

[5] There are successful approaches using virtual reality environments in the treatment of certain phobia. This success in what can be called 'desensitization' should be a warning against careless design of human-computer interfaces. How does the interface influence the user, not only in the interaction with the system, but also in his or her interactions with reality?

- the professionals such as stage directors, choreographers, stage crew, sound and light people, etc.
- the performers, hence, the actors, the musicians, the dancers, the artists, authors and poets who present their work and prefer more or other interaction with each other or/and the audience
- the public in its role of audience attending a performance; not necessarily a passive audience just enjoying a performance, but also a web-audience that can (real-time) influence the running of things during a performance or can even more explicitly take part in a performance by taking the role of an actor

In this paper we will not elaborate the possibility to use our environment for scenographic simulations. There are projects aiming at providing professionals tools and environments to help in pre-producing performances. In these projects users can build a scenography of a performance, they can move through virtual models of stage sets in real time, they can experiment with lights or camera effects, change points of view, etc. An example of such an project is CATS (see Gil et al. [11]). See also Diamond et al. [6]).

Rather we would like to add the possibility to look at our environment as a stage on which we can have on-line performances or pre-recorded performances on request. It has been mentioned before, that the computer screen can be looked at as a stage and it has been argued that the theater metaphor can help in understanding human-computer interaction (Laurel [14]). The metaphor needs to be explored further, especially with regard to interface agents and the 'artistic' agents that have been introduced in CMU's Oz project (cf. Mateas [17]) and the Virtual Theatre project of Stanford University (see http://www-ksl.stanford.edu/projects/cait/publicity.html). This requires further investigations in agent theory, but also in the possibilities to use WWW and the computer (not only its screen, but also data gloves, head-mounted devices, special clothing, etc.) to stage performances with real and virtual actors and (real) audience.

It is not unusual today to have meetings in virtual environments. Lectures have been given (and attended) in chat environments and meetings have been held in visualized meeting places. So, why not have live theatre performances over the web?

In the traditional theatre, performers and audience are physically together. There is a focus of attention of the audience in things happening on stage and performers are aware of the audience's attention. Rather than to have one special physical space where performers and audience gather, performers can be geographically dispersed and so can the audience. Moreover, there is no need to maintain the distinction between audience and performers. The environment should allow an (web) audience that can (real-time) influence the running of things during a performance or can even take part in a performance by taking the role of an actor. This requires special attention for the presence issue, both for actors and audience (see Reeve [22]).

Early online performances include a *Hamlet* parody on IRC (Internet Relay Chat) and *The Odyssey* by Homer. Well known is a VRML production of Shakespeare's *A Midsummer Night's Dream* performed live on April 26, 1998. The various avatars playing the roles were controlled by actors and the performance could be seen from any avatar's point of view of the stage (see Figure 11 and http://www.vrmldream.com/ for more details).

## 13 FUTURE RESEARCH & CONCLUSIONS

In this paper we reported about on-going research and it is clear that all issues that have been discussed here need further research. We intend to continue with the interaction between experimenting with the virtual environment (adding agents and interaction modalities) and theoretic research on multi-modality, formal modeling, natural language and dialogue management.

As may have become clear from the previous sections, our approach to designing a virtual environment for an interest community is bottom-up. At this moment the system has two agents with different tasks and with no interactions between them. Moreover, the agents do not employ a model of a user or of user groups. In general, when we talk about interface agents we mean software agents with a user model, that is, a user model programmed in the agent by the user, provided as a knowledge base by a knowledge engineer or obtained and maintained by a learning procedure from the user and customized according to his preferences and habits and to the history of interaction with the system. In this way we have agents that make personalized suggestions (e.g. about articles, performances, etc.) based on social filtering (look at others who seem to have similar preferences) or content filtering (detect patterns, e.g. keywords) of the items that turn out to be of interest to the user. These agents can be passive that wait until they are put into action or they sense changes, take initiative and perform actions, e.g. to inform the user without being asked about new information.

Figure 11: Hermia and Lysander

One of our concerns in the near future will be the introduction of a conversational agent (which has some general knowledge about well known artists and some well known performances). It may be the case that this agent will resemble Erin ("the coolest virtual bartender in cyberspace") who serves drinks in Spence's Bar, one of the virtual characters built by Extempo Systems (http://www.extempo.com/).

With this conversational agent we have obtained three kinds of dialogues (information & transaction dialogues, command-like dialogues and conversational dialogues). Another concern is an agent that is able to demonstrate how to play musical instruments. This agent requires a much more detailed visualization, including body, arms, hands and fingers and natural movements. In the Spring of 1999 Ph.D. research in this area will start.

In 1999 some versions of our virtual environment have become available for other research groups to work on. For example, in a joint project with the TNO Human Factors Research Institute user evaluation studies will be done and we hope this will help in future decisions about the direction of our work on the theatre information and transaction service interactions and the environment where they take place. Together with KPN Research we hope to investigate the possible role of MPEG4 for visualization and interactions (see Koenen [13]). A simplified and localized version of the virtual environment has been placed at a Dutch technology activity center (Da Vinci). Here, visitors are allowed to play with the system and their (verbal) interactions with the system are logged.

When developing our system there are similar systems which inspire us and from which we hope to learn. We mention:

**Trilogy** (cf. Norman & Jennings [19]) is a project from the University of London for the development of a virtual research laboratory with intelligent agents. In the laboratory, students are trained in the area of 'traffic engineering' for telecommunication. Agents are used to present information and to give access to tools. Agents can make suggestions when a user is not familiar with the possibilities of the system. In addition they can take care of the efficient use of the available resources.

**Steve** (cf. Rickel & Johnson [23,24]) stands for Soar Training Expert for Virtual Environments, a project performed at the University of Southern California, Marina del Rey, CA. Steve is a pedagogical agent with the task to help students learn to perform procedural tasks, such as maintaining equipment. It demonstrates how to perform actions and it uses locomotion, gaze, facial expressions and deictic gestures in its communication with the student. Speech recognition and speech generation have been added to allow task-oriented dialogs between Steve and student.

**MUeSLI** (cf. Wyard & Churcher [32]) is a project of British Telecom Labs on MUltimodal Spoken Language Interfaces. The multimodal interface that is designed allows language and touch access to a 3D retail system via a kiosk or over the internet. The retail system shows a 3D virtual living room, a 2D fabric palette and a virtual assistant (a 3D talking head). The user may select fabrics and have them displayed on furniture, curtains and walls in the living room.


Figure 12: Jennifer James

It is also worthwhile to look at already existing attempts to commercialize (aspects of) systems similar to the one we are building. A particular nice 'In-Person Service Character' is **Jennifer James** (Figure 12, see again (http://www.extempo.com/), a spokeswoman at a virtual auto show who greets visitors, engages them in a dialogue and a presentation of available cars. She listens to comments and questions typed by a visitor and talks using speech synthesis technology. Her face and body are 3-D animated, actions and reactions are consistent with role and personality and are coordinated with the events of the dialogue. The information she obtains from visitors is stored for follow-up and market research.

In addition to the projects which concentrate on VR environments and interaction modalities there are projects on virtual interactive studio television (e.g., the European VISTA project with applications on Interactive Drama and Interactive Presenters; in this environment viewers can actively participate and direct the programme being transmitted), on digital cinema (multi-treaded movies, interactive series with audience participation, etc.) and on agent-based TV broadcasting.

## REFERENCES


[1] Agah, A. & K. Tanie. Multimedia Human-Computer Interaction for Presence and Exploration in a Telemuseum. *Presence: Teleoperators and Virtual Environments* 8 (1), February 1999, 104-111.

[2] Bates, J. Virtual Reality, Art, and Entertainment. *Presence: Teleoperators and Virtual Environments* 1 (1), Winter 1992, 133-138.

[3] Berk, M. van den. Visuele spraaksynthese. Master's thesis, University of Twente, 1998.

[4] Beutnagel, M., A. Conkie, J. Schroeter, Y. Stylianou & A. Syrdal. The AT&T Next-Gen TTS System. In: *ICSLP–98*, Sydney, Australia, November 1998.

[5] Clermont, Th., M. Pomplun, E. Prestin and H. Rieser. Eye-movement research and the investigation of dialogue structure, Proceedings of TWLT13: *Formal Semantics and Pragmatics of Dialogue (Twendial'98)*, J. Hulstijn and A. Nijholt (eds.), 1998, 61-75.

[6] Diamond, D. & T. Berliner (eds.). The Stage Directors Handbook: Opportunities for Directors and Choreographers. Theatre Communications Group, New York, ISBN 1-55936-150-6, 1999.

[7] Dirksen, A. and Menert, L. Fluent Dutch text-to-speech. Technical manual, Fluency Speech Technology/OTS Utrecht, 1997.

[8] Dutoit, T. High-quality text-to-speech synthesis: An overview. *Electrical and Electronics Engineering* 17 (1997), 25-36.

[9] Friedman, B. (ed.). *Human Values and the Design of Computer Technology*. CSLI Publications, Cambridge University Press, 1997.

[10] Friedman, B. & P.H. Kahn. Human agency and responsible computing: Implications for computer system design. In: Friedman [9], 221- 235.

[11] Gil, F.M. et al. 3D Real-time graphic environments for theatrical and TV scenographic simulations. In: Proceedings *Nimes '98*, LLIA Nos 134-135-136, 1998, 163-167.

[12] Hulstijn, J. & A. van Hessen. Utterance Generation for Transaction Dialogues. Proceedings 5th *International Conf. Spoken Language Processing* (ICSLP), Vol. 4, Sydney, Australia, 1998, 1143-1146.

[13] Koenen, R. MPEG-4: Multimedia for our time. http://drogo.cselt.stet.it/mpeg/koenen/mpeg-4.html, 1999. See also: MPEG-4: Overview of the MPEG-4 Standard. ISO/IEC JTC1/SC29/WG11, N2459, October 1998, Atlantic City.

[14] Laurel, B. *Computers as Theatre*. Addison-Wesley 1991; 2nd edition 1993.

[15] Lie, D., J. Hulstijn, A. Nijholt, R. op den Akker. A Transformational Approach to NL Understanding in Dialogue Systems. Proceedings *NLP and Industrial Applications*, Moncton, New Brunswick, August 1998, 163-168.

[16] Lombard, M. & T. Ditton. At the heart of it all: The concept of presence. *Journal of Mediated Communication* 3, Nr.2, September 1997.

[17] Mateas, M. An Oz-centric review of interactive drama nad believable agents. CMU-CS-97-156, Carnegie Mellon University, Pittsburgh, June 1997.

[18] Nass, C., B. Reeves & G. Leshner. Technology and roles: A tale of two TVs. *Journal of Communication* 46 (2), 121-128.

[19] Norman, T.J. & N.R. Jennings. Constructing a virtual training laboratory using intelligent agents. Manuscript, University of London, 1999.

[20] Ostermann, J., M. Beutnagel, A. Fischer & Y. Wang. Integration of talking heads and text-to-speech synthesizers for visual TTS. In: *ICSLP-98*, Sydney, Australia, November 1998.

[21] Reany, M. The Theatre of Virtual Reality: Designing Scenery in an Imaginary World. *Theatre Design and Technology*, Vol. XXIX, No.2, 1992, pp. 29-32.

[22] Reeve, C. Presence in Virtual Theatre. *BT Presence Workshop*, BT Labs, 10-11 June 1998.

[23] Rickel, J. & W.L. Johnson. Task-oriented dialogs with animated agents in virtual reality. In: Proceedings of the First Workshop on *Embodied Conversational Characters*, Tahoe City, CA, October 1998.

[24] Rickel, J. & W.L. Johnson. Animated agents for procedural training in virtual reality: Perception, cognition, and motor control. To appear in *Applied Artificial Intelligence*.

[25] Roehl, B. Draft Proposal for the VRML Streaming Working Group. http://www.vrml.org/WorkingGroups/vrml-streams/proposal.html, 1998.

[26] Schooten, B. van, O. Donk & J. Zwiers. Modeling interaction in virtual environments using process algebra. In: Proceedings *Interactions in Virtual Worlds (IVW'99).* Twente Workshop on Language Technology 15, University of Twente, May 1999.

[27] Shneiderman, B. & P. Maes. Direct manipulation vs interface agents. Exerpts from debates at IUI 97 and CHI 97. *Interaction*, November-December 1007, 42-61.

[28] Sproull, L., M. Subramani, S. Kiesler, J. Walker & K. Waters. When the interface is a face. In [9], 163-190.

[29] Thie, S. & J. van Wijk. Experimental evaluation of social virtual presence in a decision making task. In: Proceedings *BT Workshop on Presence*, May, 1998.

[30] Vertegaal, R. *Look who's talking to whom: mediating joint attention in multiparty communication and collaboration.* Ph.D. Thesis, University of Twente, Enschede, 1998.

[31] Witmer, B.G. & M.J. Singer. Measuring presence in virtual environment: A presence questionnaire. *Presence: Teleoperators and Virtual Environments* 7 (3), June 1998, 225-240.

[32] Wyard, P.J. & G. E. Churcher. Spoken Language Interaction with a Virtual World in the MUeSLI Multimodal 3D Retail System. In: Proceedings *Interactions in Virtual Worlds (IVW'99)*. Twente Workshop on Language Technology 15, University of Twente, May 1999.

[33] Zahorik, P. & R.L. Jenison. Presence as being-in-the-world. *Presence: Teleoperators and Virtual Environments* 7 (1), February 1998, 78-89.

# Verbal and Written Interaction in Virtual Worlds
## Some application examples

Pierre Nugues
Institut des Sciences de la Matière et du Rayonnement
6, boulevard du Maréchal Juin
F-14050 Caen, France
pnugues@greyc.ismra.fr
http://www.greyc.ismra.fr/~pierre

## ABSTRACT

This text first summarizes what can be the respective advantages of language interaction in a virtual worlds and 3D images in language interactions and dialogue.

It then describes three examples of verbal and written interaction systems in virtual reality, starting with Ulysse, a conversational agent that can help a user navigate in virtual worlds. Ulysse has been designed to be embedded in the representation of a participant of a virtual conference. Ulysse responds positively to motion orders and navigate the user's viewpoint on his/her behalf in the virtual world. On tests we carried out, we discovered that users, novices as well as experienced ones have difficulties moving in a 3D environment. Agents such as Ulysse enable a user to carry out navigation motions that would have been impossible with classical devices.

The second example is a prototype to recreate car accidents in a virtual world from written accident reports. Reports have been supplied by an insurance company and describe most often collisions between two vehicles. We could animate scenes, coordinate entities in the virtual world and thus replay some of the accidents from their natural language descriptions. Such animation would have been difficult using classical interaction devices. The text describes techniques we have used to implement our prototype and the results we have obtained so far. It explains how an information extraction system can benefit from such a tool.

Finally, the text describes a virtual workbench to study motion verbs. From the whole Ulysse system, we have stripped off a skeleton architecture that we have ported to VRML, Java, and Prolog. This skeleton allows the design of language applications in virtual worlds.

**Keywords:** Virtual reality, Conversational agents, Spoken navigation, Scene generation from texts.

## 1 COMPUTER INTERFACES AND VIRTUAL REALITY

Computer interfaces have now widely stabilized into a set of paradigms where variations are merely cosmetic. Following the Macintosh's finder, Windows' desktop or X-Window's avatars have all converged into visualization and interaction means that include windows, icons, menus, and pointers – the so-called WIMP model.

This model may have reached a plateau and be unable to scale up to new computing entities such as the Internet or to be adapted to new tasks such as cooperative work or simulation.

From ideas to recreate desktop tools on the screen of a computer under the form of symbols – icons –, some researchers thought that virtual reality was a better paradigm to design metaphors. This pushes desktop symbolization closer to reality and presents a way to escape some trends of the GUI routine. In applications such as computer tools for collaborative work others researchers saw virtual reality as means to situate users, to make them aware of the context: their coworkers and other working teams notably, and to enable easier communications [1].

In many respects, virtual reality is appealing because it brings more realistic images and more interaction to the computer desktop. Although cognitive values of visualization and visual think have been extensively described and researched [2] [3] [4], power of images and interaction is probably best captured by the Chinese proverb:

> *I hear and I forget, I see and I remember, I do and I understand.*

And virtual reality addresses the two last points better than any other interface. From this viewpoint, virtual reality would appear as the extreme trend of the interactive desktop metaphor and an ultimate interface.

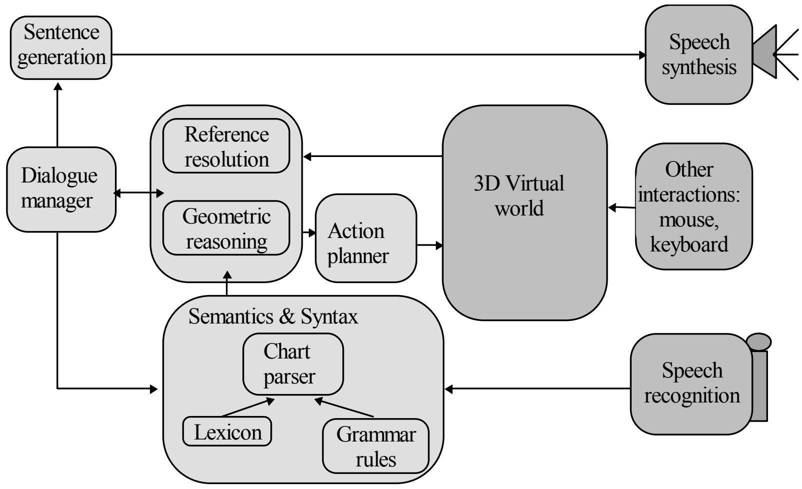



Figure 1: Ulysse's architecture.

## 2 COMPUTER INTERFACES AND LANGUAGE

In virtual reality environments however, navigation is often difficult and interaction is can be oppressive. It extends in that way drawbacks of existing interfaces as well as their advantages. Virtual reality requires extensive and sometimes tricky gestures. Opening a folder in the recreated office of a virtual world would probably require more movements than with the Macintosh Finder.

Information access, desktop control which are often tied to ease of navigation are key points of a good interface design. It's not sure that widely available virtual reality interfaces address this problem. In experiments we conducted earlier, we found that computer novices as well as experienced computer users – but unfamiliar to virtual reality – had much navigational difficulties in virtual worlds [5]. In addition, virtual reality embodies the principle of direct interaction that requires from objects to be visible, close, and in a relatively upright position. When not visible, objects are sometimes difficult to find and then to approach which adds a supplement of navigation chore.

A language interface would enable easier designation and navigation and hence help users complete their tasks faster. If this last statement has no definitive proof, a hint can be given by the analogous example of the Web development that shows that most popular portals are natural or constraint language interfaces (e.g. Altavista, Voilà, Excite, Lycos, Yahoo). Thanks to their indexing robots they prevent a user from clicking zillions of pages before finding relevant information. Although, the extent of language processing behind these sites might be discussed, it shows a clear user preference to designate things using Dutch (or French) rather than to navigate links.

Gentner and Nielson [6] in a prospective article underlined limits of the WIMP interfaces. They described the possible role of language in future interfaces. They noted that language lets us refer to objects that are not immediately visible, encapsulate complex groups of objects, and support reasoning. They predicted a slow pervasion of natural language techniques in interfaces that would allow a negotiation between the user and its interface thanks to limited linguistic capabilities.

In our recent projects, we have implemented natural language agents in an attempt to bring to virtual reality systems some advantages of linguistics capabilities. That's what we describe now.

## 2. ULYSSE

Ulysse was our first implementation of a linguistic device in a virtual world. From user studies that we undertook, we discovered that many users were not able to move properly in a virtual world. We designed and implemented a conversational agent to help their navigation [5] [7] [8] [9] [10].

Ulysse consists in a chart parser and a semantic analyzer to build a dependency representation of the word stream [11] and a case form. It also features a reference resolver to associate noun phrases to entities of the virtual world and a geometric reasoner to cope with prepositions, groups, spatial descriptions, and to enable a limited understanding of the structure of the virtual world (Figure 1).

Ulysse is embedded in the representation of the user in the world. Upon navigation commands from the user, Ulysse analyzes the word stream and navigates the user's viewpoint on his/her behalf in the virtual world. Ulysse's action engine is a planner that uses an algorithm derived from STRIPS [12] (Figure 2). Ulysse can been used with a keyboard interface or a speech recognition system such as IBM's VoiceType or ViaVoice.

## 3. ULYSSE TO NAVIGATE INTO A BRAIN

We implemented the first version of Ulysse using the DIVE virtual environment [13]. We ported it to a PC to navigate into a reconstructed brain [14]. We kept the Ulysse overall architecture, but we had to modify the image display. Our idea was to combine realistic images and dialogues to explore brain regions and their functions [15].

The Cyceron research Center in Caen provided us with magnetic resonance images of the brain. We extracted 3D surfaces from them using thresholding operations. We chose arbitrary points on the surface that we associated to an arbitrary color. The colors were then diffused on the surface.

We investigated possible interactions and we designed scenarios to manipulate and navigate into this virtual brain. The scenarios have been designed in cooperation with the art group *Das synthetische Mischgewebe*. These scenarios have been limited to consider main regions of the brain such as the hemispheres, frontal lobes, etc. We restrained navigation to carry out linear motions and rotations relative to a designated object.

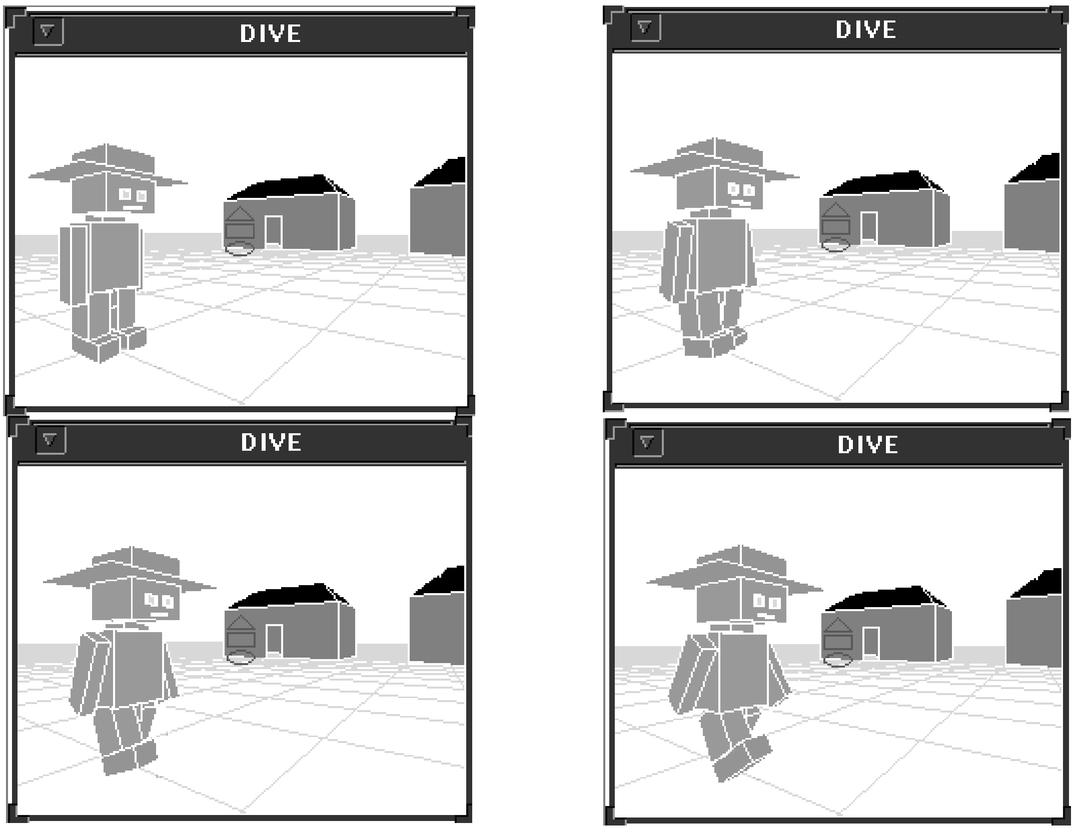



Figure 2: Ulysse's walking motion.

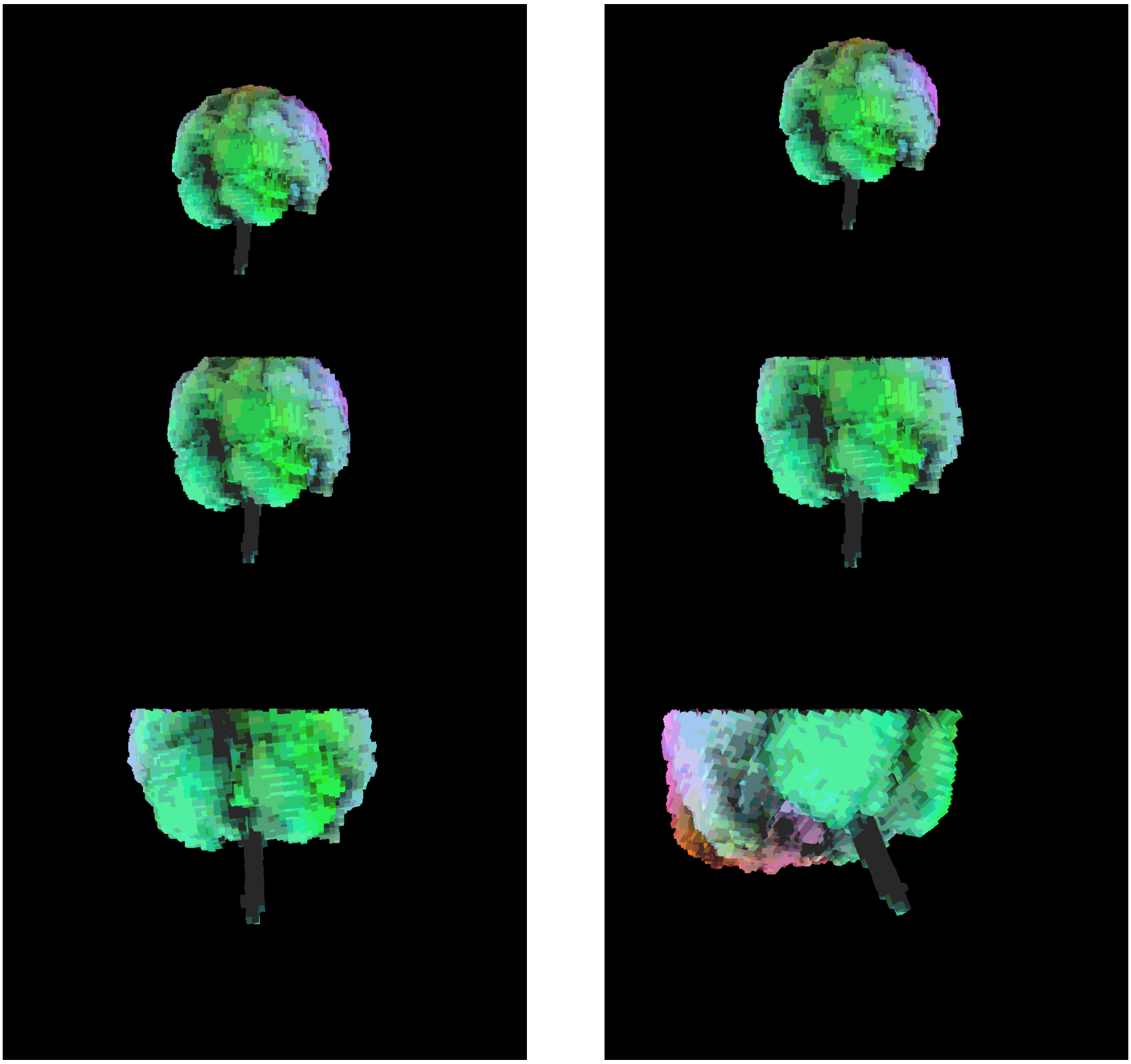


Figure 3: An animation example.

The action manager enables animation such as the sequence in Figure 3 that correspond to the utterance:

*Je voudrais voir le tronc.*

*I would like to see the medulla*

Although this prototype has only been used for art performances, it could find other applications. They include education and the interactive discovery of the brain anatomy and functions.

## 4. TACIT

The Tacit project [16] has been aimed at processing and understanding information from written reports of car accidents. Reports have been supplied by the MAIF insurance company and describe most often collisions between two vehicles.

Tacit combined two approaches. One was to build a deep model of semantics associated with the reports. And the other approach was to use information extraction methods to skim some features of the accidents.

We used the deep semantic approach to generate and animate scenes in a virtual world from corresponding texts. In addition to language processing techniques, we also made an extensive use of knowledge on driving conditions in France.

### 4.1 Modeling the Scene

While the information extraction approach was applied to original running reports, the deep semantic modeling had to consider simplified texts. Car accident reports include many understatements, negations, or cosmetic descriptions that cannot be interpreted – at the present stage of our work – without guessing the state of mind of the driver. Consider for instance:

> *Je roulais sur la partie droite de la chaussée quand un véhicule arrivant en face dans le virage a été complètement déporté. Serrant à droite au maximum, je n'ai pu éviter la voiture qui arrivait à grande vitesse.* (Report A8)

> *I was driving on the right-hand side of the road when a vehicle coming in front of me in the bend skidded completely. Moving to the right of the lane as far as I could; I couldn't avoid the car that was coming very fast.*

where the collision is not even mentioned.

We needed a simplification to make this text explicit. It yields:

> *Je roulais sur la chaussée, un véhicule arrivait en face dans le virage, je l'ai percuté dans le virage.*

> *I was driving on the lane, a vehicle was coming in front of me in the bend, I bumped it in the bend.*

Even with this simplification, the text is merely a guide to re-create the scene and not a complete description: the author of the text gives the essential details only. The first clause of the text A8 for instance assumes that the driver follows the course of the road at a correct speed.

In order to reproduce this piece of reasoning, we used a processing architecture consisting of linguistic analyzer and a road domain model [16] [17] [18] [19]. Units identify entities and link them together using space and temporal relations. Linguistic processing splits sentences into a sequence of events. The result is a set of entities and the relations that link them. A major difference with Ulysse's was that Tacit involved an elaborate temporal modeling of events and actions [20].

Here are results we obtained with text A8 [21]:

*I was driving on the lane*

- Entities:
  `v0: vehicle; s0: person; r0: road; e1: trajectory; im1, ip1, ie: intervals`
- Time relations:
  `contains(ip1, im1) & before(im1, ie)`
- Relations:
  `[ip1] driver(s0, v0)`
  `[ip1] within(v0, r0)`

*a vehicle was coming in front of me in the bend,*

- Entities:
  `v1: vehicle; s1: person; sr0: bend; e2: trajectory; im2, ip2, ie: intervals`
- Time relations:
  `contains(ip2, im2) & before(im2, ie) & simultaneous(im1, im2)`
- Relations:
  `[ip2] driver(s1, v1)`
  `[ip2] within(v1, rr0)`
  `[ip2] getting_closer(v1, v0)`
  `[ip2] facing(v1, v0)`

*I bumped it in the bend.*

- Entities:
  `v1: vehicle; s0: person; e3: bump; im3, ip3, ie: intervals`
- Time relations:
  `contains(ip3, im3) & before(im3, ie) & before(im2, im3)`

From the first sentence *I was driving on the lane*, we create a person `s0` and his vehicle `v0`, a road `r0`, an event `e1` corresponding to the movement of the driver. We also create time intervals (`im, ie, ip`) to locate the events in a chronological order and to determine whether they have finished or not.

We have relations specifying that the driver of the car `v0` is `s0`, and that, during the process interval `[ip1]`, the driver is on the road `within(v0, r0)`.

These results are integrated in the scene construction unit. They are combined with knowledge of the domain in order to complete the description. The reasoning processes builds a chronology of events and the vehicles' trajectories. The description is translated into the VRML geometric format. It features a modeling of cars, roads, and road equipment. Animation considers the event list with their temporal

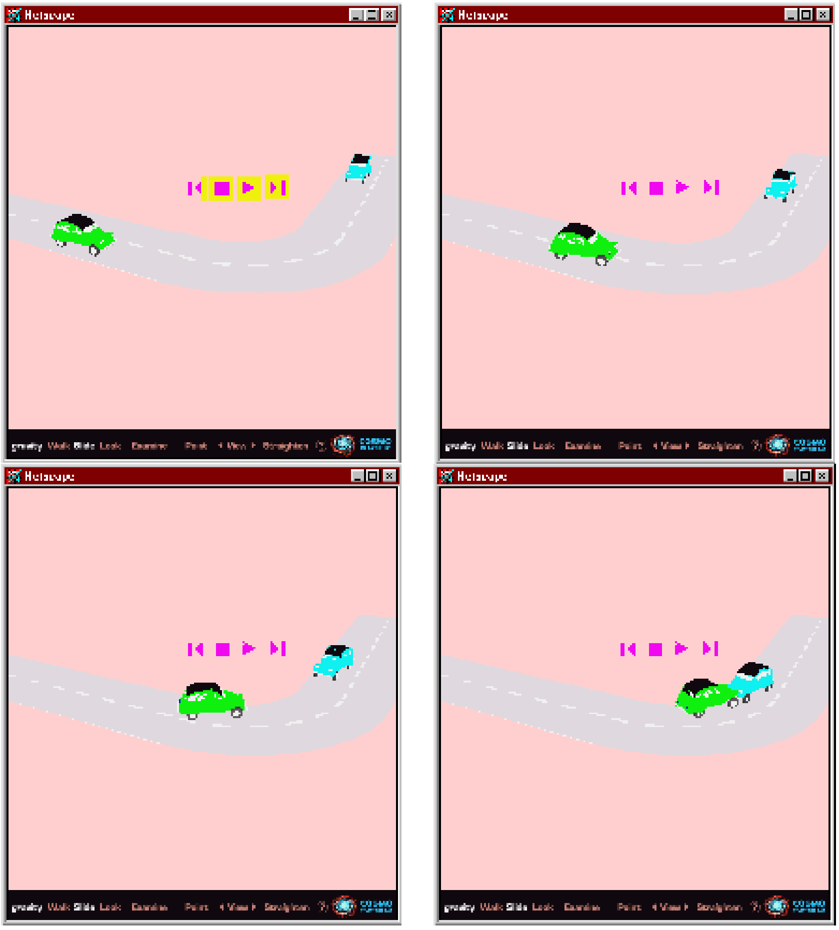


Figure 4: Scene synthesis corresponding to text A8.

length; the object list; their positions, speed and directions at the beginning and at the end of each process. It compiles them into VRML interpolators so that the scene can be animated. Figure 4 presents snapshots of the synthesized scene.

## 4.2 EMBEDDING A PLANNER

The temporal model we used in the previous example couldn't formalize well information relative to some driving events. For instance, it was impossible to encode the moving position of a vehicle relatively to the road and at the same time, relatively to the position of another moving vehicle. We selected a couple of actions such as overtaking, or the behavior of vehicles at a junction and we modified the processing architecture. We implemented planners to reproduce these more complex events.

Representing an overtaking scene, for instance, requires the planner to split the overall action into a set of simpler ones. It also requires that the overtaking vehicle constantly monitors the position of the vehicle that is overtaken. We had to use a reactive planner because actions couldn't always be pre-computed and depended on the current state of the geometrical database.

From the Ulysse model where the planner is embedded in the user's avatar, we incorporated planners in vehicles. The planner is embedded in the tracking vehicle which is called the "actor". This vehicle updates its position relative to the vehicle in front: the "object". The planner applies rules to compute the successive positions and orientations of the vehicles within the time frame in which the action occurs.

The initial problem *overtake* is fractionated into sub-problems. It must satisfy an initial condition: "go faster than the other vehicle". Then it must complete a

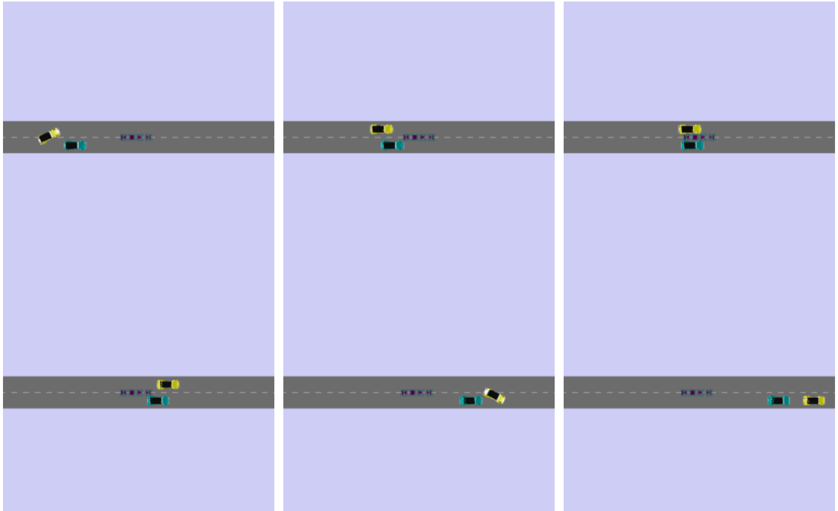


Figure 5: Overtaking.

sequence of actions: get closer to the leading vehicle; move to the left lane; go in front of the overtaken vehicle; finally return to right lane. Rules are triggered by comparing positions of the two vehicles. For instance, the actor changes lane when it reaches a given distance from the object. Knowing the positions and the entities of the scene, the system compiles the VRML interpolators that define the overall motion. Figure 5 shows snapshots of the synthesized overtaking.

Although the information extraction techniques can produce more substantial results, they are only able to extract templates from the reports. We believe that visualization and animation can make the text description easier to understand. They could help insurance analysts assess the likeliness of the report. We are presently considering them to check whether they could be useful to extract the sequence of events as the deep modeling did for simplified texts.

## 5. VRML Ulysse

Ulysse has recently undergone some changes. We rewrote it to adapt its architecture to Internet programming tools and languages, namely Java and the VRML programming interface. We also modified the parser that was quite slow eliminating the chart parser and replacing it by a more efficient algorithm. Although VRML Ulysse is not a complete port it provides the programmer with a skeleton that is relatively easy to adapt to other tasks [22].

VRML Ulysse has three main components: a language engine in Prolog, the VRML world, and a Java applet to form the input interface and to ensure communication with the VRML world and the language engine. Both are linked through the External Authoring Interface. The Java applet is derived from the Script class that provides facilities to send and receive events from a VRML 2 world (Figure 6).

Master students from the university of Caen started using it to implement logical and cinematic definitions of French motion verbs. They worked on *sauter* (jump) and *courir dans* (run into) for which they implemented a model in Prolog that they could visualize with VRML Ulysse.

## 6. Conclusion

We have described three prototypes of conversational agents embedded within a virtual worlds. These prototypes accept verbal commands or descriptions from written texts. Ulysse conversational agent enables users to navigate into relatively complex virtual worlds. It also accepts orders to manipulate objects such as a virtual brain. We believe that a spoken or verbal interface can improve the interaction quality between a user and virtual worlds.

We have also described an agent to parse certain written descriptions of simultaneous actions of world entities. We have outlined techniques to synthesize and synchronize events from these texts. Although texts need to be simplified and the number of actions is still limited, we could animate the entities according to the description and simulate some of the accidents.

Finally, we have sketched the architecture of a new version of the Ulysse system in VRML, Prolog, and Java which allows the design of language applications in virtual worlds. It has been used to implement the

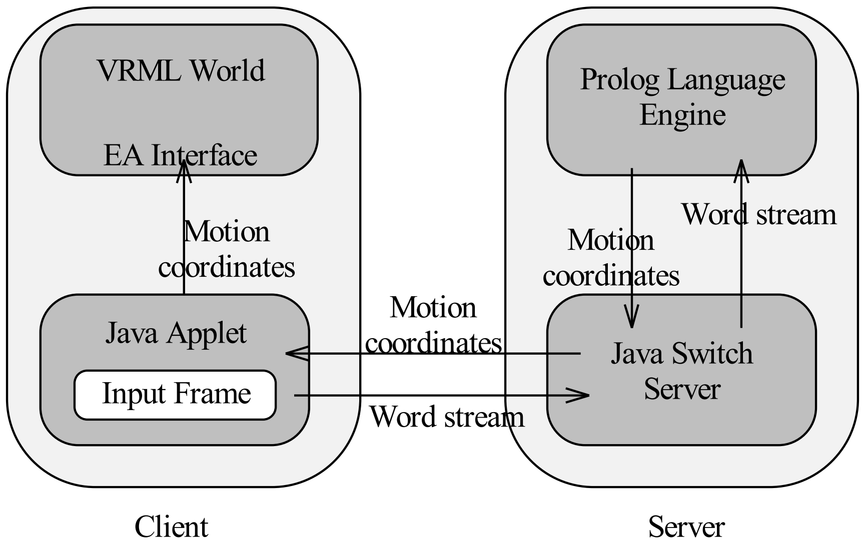


Figure 6: VRML Ulysse architecture.

cinematic definition of French motion verbs. While there are theories in this domain, few are proven due to the lack of experimental devices. Experimental tools are central to the improvement or design of theories. We hope such a system enables the experimentation of theories on motion verbs making them implementable and provable. The prototype is available from the Internet.

In conclusion, we believe that the virtual reality and computational linguistics communities could have a fruitful cooperation. In spite of their different technical culture and history they could create paradigms to explore future interfaces and new ways of computing.

## REFERENCES

[1] Benford, S. & Fahlen, L. "Viewpoints, Actionpoints and Spatial Frames for Collaborative User Interfaces". In *Proc. HCI'94, Glasgow*, August 1994.

[2] Kosslyn S, *Ghosts in the Mind's Machine*, New York, Norton, 1983

[3] Kosslyn S, *Image and Brain*, Cambridge, MA MIT Press, 1994.

[4] Tufte E., *Visual Explanations, Images and Quantities, Evidence and Narrative*, Graphic Press, Cheshire Conn., 1997.

[5] Godéreaux C., K. Diebel, P.O. El-Guedj, F. Revolta, & P. Nugues, An Interactive, Spoken Dialog Interface in Virtual Worlds, in: *Linguistic Concepts and Methods in CSCW*, John H. Connolly and Lyn Pemberton (Eds), Chapter 13, Springer, London, 1996, pp. 177-200.

[6] Gentner D. & Nielson J., The Anti-Mac Interface, *Communications of the ACM*, 39(8):70-82, 1996.

[7] Godéreaux C., P.-O. El Guedj, F. Revolta, & P. Nugues, Un agent conversationnel pour naviguer dans les mondes virtuels, *Humankybernetik*, 37(1):39-51, 1996.

[8] Nugues P., Godéreaux C., El Guedj PO, & Revolta F., A Conversational Agent to Navigate in Virtual Worlds, in: *Proceedings of the Eleventh Twente Workshop on Language Technology*, S. LuperFoy, A. Nijholt, and G. Veldhuijzen van Zanten eds., pp. 23-33, Universiteit Twente, Enschede, June 1996.

[9] Bersot O., El Guedj P.-O., Godéreaux C., & Nugues P., A Conversational Agent to Help navigation and Collaboration in Virtual Worlds, *Virtual Reality*, 3(1):71-82, 1998.

[10] Godéreaux C., PO. El Guedj, F. Revalta, & P. Nugues, Ulysse: An Interactive, Spoken Dialogue Interface to Navigate in Virtual Worlds, Lexical, syntactic, and semantic issues, in: *Virtual Worlds on the Internet*, J. Vince and R. Earnshaw (Eds), Chapter 4, IEEE Computer Society Press, Los Alamitos, 1999, pp. 53-70 & 308-312.

[11] El Guedj PO & Nugues P, Analyse syntaxique combinant deux formalismes au sein d'un Chart

hiérarchique, *Humankybernetik*, 38(3):135-144, 1997.

[12] Fikes R. & Nilsson N. J. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2:189-208, 1971.

[13] Andersson, M., Carlsson, C., Hagsand, O., & Ståhl, O. DIVE, *The Distributed Interactive Virtual Environment, Technical Reference*, (Kista: Swedish Institute of Computer Science), 1994.

[14] Ludwig M., A. Lenoir, & P. Nugues, A Conversational Agent to Navigate into MRI Brain Images, in: Proc. *Interfaces 97, 6th International Conference*, Montpellier, 28-30 mai 1997, pp. 207-210.

[15] Damasio H., *Human Brain Anatomy in Computerized Images*, Oxford University Press, 1995.

[16] Victorri B. at al. *Le projet TACIT, Traitement Automatique pour la Compréhension d'Informations Textuelles*, Rapport de fin de projet, Université de Caen, 1999, http://elsap1.unicaen.fr/rapport.pdf

[17] Enjalbert P. & Victorri B., Du language au modèle, *Traitement automatique des langues*, 35(1):37-64, 1994.

[18] F. Pied, C. Poirier, P. Enjalbert & B. Victorri, From language to model, *Proc of ECAI-96*, 1996.

[19] Poirier C., Mathet Y. Enjalbert P, La compositionnalité à l'épreuve des faits, à travers un projet de compréhension automatique de constats d'accident, *Traitement automatique des langues,* 39(1):99-129, 1998.

[20] Gosselin L., *Sémantique de la temporalité en français*, Duculot, Louvain-la-Neuve, 1996.

[21] Tabordet F., Pied F, & Nugues P, Scene Visualization and Animation from Texts in a Virtual Environment, *CC AI. The journal for the integrated study of artificial intelligence, cognitive science and applied epistemology. Special issue on visualization*, Guest editor: T.G. De Paepe,15(4):339-349, 1999.

[22] Baumard L., R. Bertholon, & M. Bittar, *Navigation dans un monde VRML*, Projet de fin d'études, ISMRA, Caen, 1998.

[23]

# EISCAT virtual reality training simulation

# A study on usability and effectiveness

Lakshmi Sastry, D R S Boyd
Advanced Interactive Systems Group
CLRC Rutherford Appleton Laboratory
Didcot, OXON, OX11 0QX

## Abstract

This paper describes the results from an user evaluation carried out to study the effectiveness of integrating a virtual reality (VR) based operational simulation as part of a traditional classroom based training course. The virtual environment (VE) consisted of the operational simulation of the EISCAT radar control system. The contribution of the VR based simulation was to enable the trainer to introduce the training content with a realistic impact and transform the learning into an experiential process which would not have been possible otherwise. The participants were young scientists attending a week-long training course to learn about the operational aspects of EISCAT as well as data analysis. The VR sessions were interspersed within classroom lectures. The virtual environment required the users to familiarize themselves with the environment and the equipment in it, monitor the simulation processes, identify random fault conditions and interact with the environment to reset these. Both trainees and trainers were asked to provide feedback on the effectiveness of the inclusion of the simulation in the course. Results indicate a positive influence of the VR based training in enhancing the learning experience, augmenting and reinforcing the traditional training material. In particular the principal trainer commented that the simulation was "a very useful means of illustrating the points" he was trying to make as well as the VE providing an almost in-situ context for highlighting technical issues. The simulation used the virtual hand metaphor for object selection and manipulation. The noise due to field interference and the resultant performance error in the electromagnetic input device significantly affected the ease of interaction. Despite this drawback, trainees expressed the desire to undertake more interactive tasks within the immersive environment, which, they commented, increased their sense of involvement and hence enhanced their learning.

## 1. Introduction

***Training rationale***: Characterization of any situation in terms of identifiable objects with well defined behaviours enables a problem solver to find the general rules that apply to that situation in terms of those objects and properties, apply those rules logically and draw conclusions [1]. Constructivist approach to pedagogy favours explorative self-discovery via problem solving as an effective component of education and training [2]. Zeltze [3] proposes that autonomy (the ability for a computer based environment to follow their own goals, behaviours and evolve dynamically during runtime), presence (the sense that the participant is sharing the data space of the computer based environment) and interaction (of the human with that environment he is sharing) as the three quantitative measures that characterize the effectiveness of a learning/teaching environment and rates VR based systems highly on all these three axes. Other qualitative studies on VE also indicate a positive correlation between the ability to *focus* selectively at relevant sources of information and the ability to process that information in isolation for better *task performance* within a VE and *immersion*, the perception of being surrounded in a virtual environment and *involvement*, the heightened response to the continuous stimuli that the environment provides [4]. Based on such studies, the National Science Foundation (NSF) Workshop on Global Change, VR and Learning [5] proposed the general principle that "VR improves learning, when it does, by providing learners with new, direct experience of phenomena they could not have experienced before either in direct interaction with the real world or using other technologies". Clancey [6], Crumpton and Harden [7] Loftin [8] and Byrne [9] also note the attribute of VR technology to attract and retain attention, and provide motivation. VR has the ability to support the representation of any data, however abstract their referents are, without the need to simplify the complex concept behind the data, and to support direct interaction with the objects that represent the data, in authentic and quasi-authentic ways. These attributes of VR make it an ideal technology to support experiential training in real-world processes, in particular where such training with operational equipment will be costly, of limited availability or involve hazardous environments or equipment[10-14]. From this point of view, the contribution of the VR based simulation, described in this paper, was to transform a classroom based training into one of an experiential real-world problem-solving process which would not have been possible otherwise.

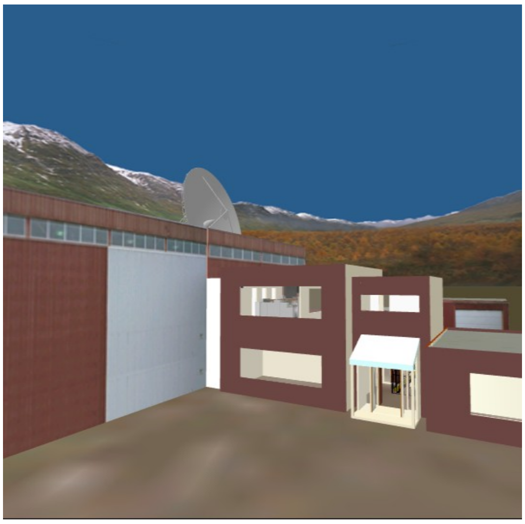


Figure 1. Overview of Eiscat site

***Evaluation Rationale***: As qualitative evidence mount, it is perhaps not surprising that there is a technology push to apply it to novel applications such as the EISCAT operational training scenario described here. It is important that whenever new Advanced Learning Technology (ALT) is used in educational applications, evaluations take place to understand the efficacy of including such methods. This paper describes qualitative results from an user evaluation conducted to understand, primarily, the usefulness and effectiveness of including an immersive VE as an integral part of a traditional training course. The virtual environment consisted of the operational simulation of the EISCAT radar control system(Figure 1). Typical interactions within the VE involved the starting and stopping of experiments, monitoring their progress and the detection and correction of various fault conditions. Given that the simulation was constrained to make use of available VR hardware and software development environments, the secondary goal was to gain an insight into the usability issues relating to the technology used and its application in the current context. For the evaluation, participants were encouraged to articulate their thinking as they navigated and interacted within the virtual environment. They also completed detailed questionnaires after each session.

Figure 2. A view of the control room

***Methodology Rationale***: While discussing the case for the use of immersive VE for education and training, Winn[2] raises guidance (the interaction between the learner and the tutor), feedback (semantically correct system response to user actions which is neither missed nor misunderstood) and collaboration (with peer group) as the three problematic issues within such environments. Our approach was to minimize the impact of these problems as much as possible in the way the training scenario was designed. We created as realistic a visual and functional simulation as possible. We interspersed the immersive sessions between the classroom lectures. We traded off some loss of the sense of presence by allowing the trainer and trainee to discuss technical aspects, while the trainee was immersed in the VE.

Section 2 describes the background to the training course and the VR based EISCAT operational simulation. Section 3 describes the background and the methodology for the evaluation. Results of the evaluation are discussed in Section 4. Section 5 contains a brief note on conclusions and current work being carried out to address issues elicited from user feedback.

## 2. EISCAT VR simulation

### Background

EISCAT, the European Incoherent SCATter radar facility consists of three geophysical research incoherent radar systems, two of which are located close to the city of Tromso in Norway. The radar systems are used by scientists to receive and analyse scattered radio signals from ionosphere. Data is gathered by the scientists by conducting experiments which are either predefined or newly designed and run by the members of the EISCAT consortium. When an experiment is conducted, analogue signals

Figure 3. UHF antenna

are received, digitized and stored. During that phase, a real-time analysis is carried out to provide on-line estimates of ionospheric parameters. The control system also provides a variety of system control and diagnostic information which allow the scientists to monitor the quality of the data they are collecting and make real-time decisions on the experimental facility.

As part of their support activity for the UK EISCAT community, the Space Science Department at the Rutherford Appleton Laboratory conducts annual week-long training course for young scientists to introduce them to the operational aspects, fault detection and correction during the running of the experiments as well as data analysis.

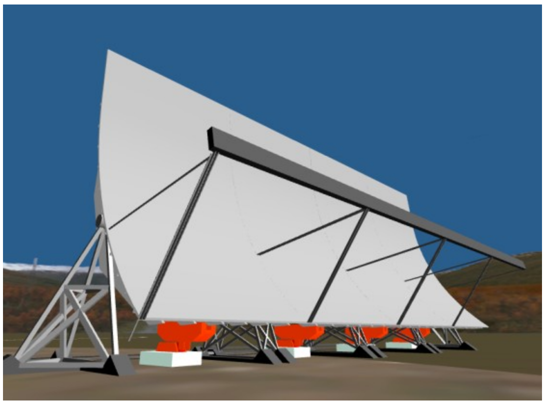


**Figure 4. VHF antenna**

## VR based simulation

***Software development platform***: The virtual reality based operational simulation was developed using the commercial VR toolkit dVS ™[15]. dVS consists of an object database and a set of concurrent interactive processes such as the virtual user body input, graphics output, collision detection etc. dVS provides a higher level authoring tool, dVISE, for rapid prototyping. dVS contains C libraries that provide the application programmer with access to virtual objects, their attributes as well as device level interfaces so that application specific functionality and behaviours could be implemented.

***The hardware setup***: The simulation was built and run as a single process on a Silicon Graphics Onyx Reality Engine system with three graphics pipes. During the VR training sessions, the visual output from the graphics pipes were channeled to a HMD (Head Mounted Display) and to two stereo projectors. The cinema style setup enabled one user to immersively navigate and interact within the virtual environment while enabling others to participate as audience in a semi-immersive, albeit passive, mode of interaction. This setup allowed the trainer and trainee to interact, discussing technical aspects in context but with a trade-off on the sense of presence experienced by the fully immersed trainee. For the immersed participant, the head and hand positions were tracked. A 3D spacemouse with five trigger buttons was used as input device. The output from one of the graphics pipes was also directly captured onto video together with superposed audio captured from one microphone placed within the VR Centre and another attached to the immersed user.

***The virtual simulation***: The virtual environment presented the salient features of the Tromso site, consisting of the main building housing the radar systems control room (Figure 2) and the two radar systems (Figure 3,4). It recreated the outside scenery of the mountains and the valley by stippling together textures from photographic images. The operational simulation was controlled by an application program which was both a parser for the radar command and control sequence and a handler of the user's input and interaction with the VE. It realistically emulated the operational behaviours and fault conditions of radar control systems.

***User interaction tasks***: Typical user interaction tasks in the VE consisted of navigation (primarily for the purpose of familiarization with the environment, identifying various equipment components and their function), object selection and manipulation tasks consisting of starting and stopping experiments and changing alphanumeric parameters via 3D floating menus(Figure 5). There was also a need to directly manipulate buttons on the racks of equipment when a specific fault condition occurred.

***User interaction metaphors***: Walking and flying "magic carpet" metaphors was used for navigation. The user moved in the direction of gaze. Forward and backward movements were mapped onto buttons on the physical input device. The user could accelerate by pressing the designated pick button on the input device together with either of the two movement buttons. Hence, there was a modal switch to the pick button functionality. The immersed user was represented by a virtual hand within the VE. This virtual hand was used to pick and manipulate objects. Pick events were generated by triggering the pick button on the input device. 3D floating virtual menu metaphor was used to input application commands


Figure 5. 3D menu of fault conditions

and other input such as numeric values (Figure 6). The floating menu could be located with respect to either the user's head or hand position or it could be placed with respect to the virtual environment (world frame of reference). The floating menu contained sliders, buttons and digital displays with object behaviours equivalent to their 2D counterparts in WIMP (Windows, icons, menus and pointer) based 2D desktop systems.

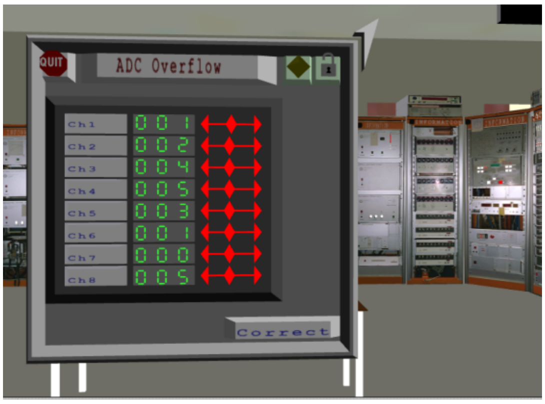



Figure 6. 3D menu for resetting values

## 3. User evaluation

***Background***: Kaufman et al.[16] stress that for virtual reality interfaces, it is paramount that whenever new interface metaphors are created, validation of its functionality, usability and effectiveness are undertaken via usability testing. Usability attributes of an interactive system are usually measured by taking one of four views of the man-machine interaction, namely, the interaction-oriented view, the user-oriented view, the product-oriented view and the formal view [17, 18]. The interactive qualities of virtual reality user interfaces are usually evaluated and quantified from the interaction and user oriented views[19-28]. In particular, it is common to get quantitative measures of ergonomic attributes. For such usability testing, applications are built specifically to validate the user interface device or metaphor.

In the current context, the motivation behind the user studies is two-fold, as noted earlier. The first reason arose out of the desire to explore whether the innovative incentive to *push* this new ALT into education and training is indeed beneficial in providing an effective learning experience for any general training scenario.

The other reason for the evaluation was to get qualitative feedback on usability issues. For this part, it would be more appropriate to consider the evaluation to be formative for the following reasons: Annual campaigns to conduct EISCAT experiments are scheduled centrally at the EISCAT Association level. Most of the scientists who attend the EISCAT training course at CLRC may participate in one such campaign at some point. Apart from that, there is no immediacy in the application of their newly acquired knowledge from the training course at CLRC. Reflecting this informal setting, the course contained no pedagogic evaluation component. As a result, in common with many innovative ALT projects, there was incomplete requirement specification, with a necessity to undertake development based on an under-informed design. An additional development constraint was the need to map specified interaction tasks to the interaction techniques supported by the underlying VR toolkit. Under these circumstances, it appeared that non-technical formative evaluation would be a method of gaining better insight into usability issues and then making an informed design choice[29]. Hence a formative evaluation with an interaction-centred view as well as an user-centred view appeared as a reasonable strategy to follow. The aim was to gain insight which would enable discerning adoption of novel interaction techniques for the interaction tasks that could be introduced during the next iterative development.

***The participants***: There were seven candidates, two female and five male, aged between 22-29 years. They were science graduates, each registered for Ph.D. with a UK university. None of them had used a virtual reality system or visited Tromso before. They were motivated and capable of mastering the user actions necessary for interacting within the virtual environment.

***The evaluation methodology:*** As noted in the Introduction, the immersive VR sessions were aimed to maximize guidance and feedback and augment the training content. The normal course was rescheduled to include three immersive VR sessions interspersed between classroom lectures, two of which were specifically designed as immersive problem-solving exercises for the trainees.

***Immersive VR sessions***: The first immersive VR session was a group session, added onto a one hour introductory lecture by the principal trainer, introducing the EISCAT control facility. One of the trainers piloted the target-oriented immersive navigation within the virtual environment as the lecturer introduced various equipment and their functionality. The trainer also used his customary training material of overhead projector slides.

The next two sessions were individual students' sessions which ran in parallel to hands on computing tutorials for data analysis. During the first of these sessions, each student was asked to gain familiarization with the immersive VR system devices as well as the VE. This was followed by a short test in which the student was asked to identify various equipment components. Each student took about 20 minutes to gain their familiarization. Simple interaction tasks included in this stage consisted of interacting with information buttons to enquire the name of components and starting and stopping experiments. Each of the these sessions lasted about 30 minutes. The second series individual sessions consisted of more rigorous exercises, requiring the students to monitor the radar control simulation to detect randomly introduced fault conditions. They had to articulate their diagnosis to the trainer and take remedial actions to reset the simulation. These sessions took place towards the end of the training course. The duration of these sessions tended to vary, lasting in general about 45 minutes, depending on the discussion between the trainer and the trainee as well as individual trainee's interest in exploring the VE further.

***Instructions***: Each participant was given detailed written and oral instructions regarding the VR interactions before each session. The principal trainer produced hand-outs explaining the particular aspect of EISCAT operation that the candidate was expected to consolidate at each session. These were based on the classroom lecture material covered prior to the immersive VR sessions. For instance, the trainees were taught, in the lectures, about a set of six fault conditions that could occur during the running of standard experiments which would seriously corrupt the data gathered. They were informed a random subset of these fault conditions would be randomly introduced into experiments during their final immersive VR sessions. The student had to identify the fault condition when it occurred and take remedial actions.

***Questionnaire***: Each participant was asked to complete sections of a detailed questionnaire at the end of each session. These contained a general section on factors relating to the user's physical comfort with immersive hardware and factors such as lag and VE lighting that influenced their state of movement and visual comfort. The other sections of the questionnaire were designed to gather subjective marking and qualitative comments on aspects of the virtual simulation relating to its fidelity and hence the ease or otherwise of identifying virtual components and fault conditions. These were designed to elicit the student's comments on the effectiveness of the immersive VE in reinforcing their training. The users were also asked to comment on the interaction techniques they used for navigation, object selection and manipulation. Appendix contains a subset of the list of questions used to derive the discussion results in the next section. A typical example of individual questions was of the form given below:

- Were the visual cues effective in identifying normal EISCAT operational modes

Never | | | | Always

- Please give any additional comments on the effectiveness of visual cues for EISCAT operational modes

Quite separately, the trainers were asked to complete a questionnaire and a debriefing session between the trainers and the developers was carried out after the course.

## 4. Results and discussion

The primary objective of the analysis of the data gathered both on videos and through the questionnaires was to gain qualitative understanding of the two issues, namely, the efficacy of the inclusion of the VE in training and the usability of the interaction techniques. The user sample was too small for a separate control group. There was also a lack of comparative measures due to the lack of availability of feedback from previous training courses and of different interaction techniques. However, the qualitative insights gained appear particularly valid in the context of this application. The users proved to be enthusiastic, discerning learners and users of the VR system, providing articulate feedback via oral and written comments. For instance, on their own accord, they had all highlighted specific instances where the interaction difficulty they were experiencing were due to the interaction technique or to their lack of absorbing the lecture material or even due to the density of the training material covered in a particular module!

***Navigation***: The set tasks undertaken during the first series of individual immersive VE sessions for the students required them to gain familiarization with the locations and names of the racks of computer equipment in the main control room. Each of these racks had an information button at its top left corner. Selecting a button would display its name in 3D space for a short time(Figure 7).

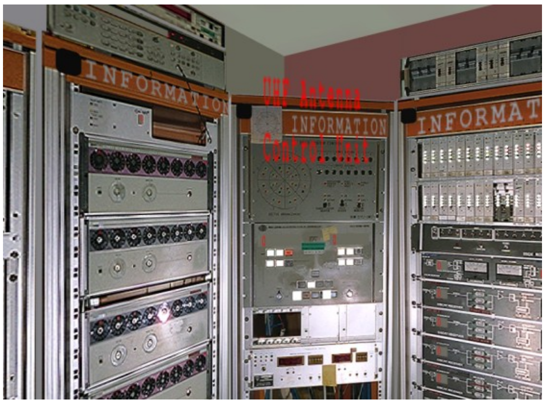



Figure 7. Equipment information display

All textual information in the VE were located with respect to fixed world coordinate system to maintain consistency. These information buttons and the associated texts were located above eye-level in order to avoid occluding other buttons and digital displays on the racks. Users commented that they had no problem reading the text, however the need to look up, fly close to select the information button with the virtual hand and then step back to read the text was found to be difficult due to the noise error from the input device's electromagnetic sensor[31]. Users tended to overshoot the target both during navigation towards and selection of the information buttons. A laser or wand metaphor for object selection would have removed the need for the navigation and significantly improved the interaction. Users also found that the resolution of the head mounted display made a significant difference to their close examination of the detailed digital displays on the racks.

One of the navigation tasks every trainee undertook was to move from the control room via the door, down a staircase, along a long narrow corridor to locate the water-closet and retrace the steps to get out of the building to inspect the antennae. One of the trainees found the need to look up while climbing the staircase as extremely counter intuitive and two more found it disconcerting. It appeared that it took significantly more practice to achieve this goal. There was consensus that collision-enabled restraint on user's movement close to solid objects would be helpful with an additional constraint on vertical motion of the body while moving within the main building. However, once outside the buildings, the magic carpet navigation, with its support for accelerated motion, was deemed most appropriate uniformly. It enabled users to get underneath the antenna, position themselves at ground level and gain a good impression of the true scale of the antenna while also enabling them to soar above and gain a bird's eye view of the entire EISCAT site.

It would be reasonable to conclude that supporting the user to switch between the magic carpet and point-fly navigation techniques would enhance navigation experience within this VE. An orbital mode of navigation around the antenna could also enhance user experience.

***Object selection and manipulation***: User tasks required them to interact using the virtual hand to select and pick objects to convey commands such as the starting and stopping of experiments. 3D widgets in floating menus [Figure 5,6] and virtual buttons on various control equipment (such as those on the middle rack in figure 7) were used for issuing such application commands and resetting numerical input parameters. All these interaction objects were tailored to provide visual feedback when the user's virtual hand intersected them. However, the jitter in the input device was at far too high a frequency for the user to register the visual highlight and trigger the selection event with ease. This rendered the taking of any time dependent performance metrics inaccurate. User frustration was reflected in feedback to questions 3.2.2 (Appendix) and 3.2.4 (second set of statistics). One of the trainees found the object manipulation tasks causing her unacceptable arm and neck muscle fatigue and another felt the jitter causing eye strain. For virtual objects such as the buttons on racks, increasing the virtual area of selection improved ease of interaction but this facility of customizing the size of individual 3D widgets at leaf node level was not available for the built-in 3D menu system. Making these widgets uniformly large broke the spatial sense of proportion within the control room and obscured view.

It would seem reasonable that a laser ray or cone emanating from the virtual hand, or a virtual wand held in the virtual hand intersecting with the object of choice would have enabled the users to trigger selection events with more ease and reach for objects beyond arm's length without any need for navigation. The reduction of the sensor's sensitivity to noise or an alternative way to increase the tolerance to this noise error at the software interface level needs to be resolved.

***Effectiveness of the VE***: There was unanimous agreement among the trainees that the VR based simulation enabled them to relate the lecturer's system descriptions in terms of a more realistic spatial context of the control room and the site. Typical comments on this aspect showed that the VE "helped to convey the full interplay of operation and control information available to the experimenter which could not have been achieved with over head projector sheets alone" and that it "gave an expectation of how EISCAT looked and functioned". Subjective marking for questions 3.1.1-3.1.3 support these comments. There was consensus that full immersion helped the trainees to gain a sense of presence and achieve better task performance. For these reasons, the inclusion of the VE in future courses was strongly recommended (questions 4.6 - 4.11 b in Appendix).

The realistic simulation helped the trainees to spot the fault conditions, with failures to do so "occurring only due to their not absorbing the material in technical hand-outs". More importantly the VE appeared to provide the context for the trainees to seek clarifications from the trainer in the context of their action and the system's response. A typical example was the error involving channel attenuation parameter which was misunderstood by every one of the trainees who failed to note that a high value for this parameter would be unacceptable. Unfortunately the two errors associated with this parameter each had degrees of severity, necessitating the users to recognize these in terms of the attenuation values and issue different sets of commands. It was not surprising that the maximum conversation between each trainee and the trainer took place in the context of these errors. In view of the problems associated with the selection mechanism, the simulation implemented a simplified mapping of triggering a single reset event for all the input attenuation and frequency numerical values. Thus the interaction avoided the need for the user to finely control and manipulate the scale widgets while at the same time displaying the appropriate values that should be set. Most of the trainees commented that given the complexity of the concept, setting these values, however difficult the interaction was, would have reinforced their understanding as they would have directly observed the effect as they reset the parameter for each of the eight channels. Another possible direct manipulation task that could have been associated with these two fault conditions was the changing of filters on one of the equipment racks as they would have to do in the real-world process. The trainers specifically did not consider it as a task worth undertaking within the VE. Despite the difficulty of interacting above eye-level, trainees still expressed a desire to carry out this task as a way of "getting more active in the virtual environment"[3,4].

The feedback from the principal trainer indicated that the simulation was "a very useful means of illustrating the points" he was trying to make and that he was "actually very pleased with the way the students were able to identify and respond to the problems" which occurred in the second set of student sessions. He also made an interesting observation that the students finished the course fresher than any from the previous courses and that the positive feedback from the students made the additional commitment from him to participate in the one to one immersive sessions worth his while.

# 5. Conclusion

An operational simulation was included in a classroom based training course. The contribution of the VR based simulation was to transform the traditional "chalk and talk" learning into an experiential process with strong elements of discovery via interaction with objects in the virtual environment and contextually based discussions with the tutor. The evaluation carried out to understand the usefulness and effectiveness of this inclusion clearly indicates the benefits both from the points of view of the trainer and the trainees. These benefits include the enhanced learning experience which reinforced the training content by putting it in an almost 'in-situ' context.

There was feedback from the formative evaluation component to highlight usability issues as well. This indicated that the gaze-directed magic carpet navigation metaphor was non-intuitive when it necessitated persistent looking up in confined spaces such as climbing staircases. However this metaphor was considered intuitive when flying through large open spaces as well as when searching for information at and below eye-level. The virtual hand metaphor for object selection similarly failed for selecting objects above eye-level. This difficulty was exacerbated by the noise and errors produced by the electromagnetic tracker of the hand-held input device.

It seems reasonable to conclude that a dual navigation mode consisting of the magic carpet and point-fly metaphors will enhance navigational usability within this VE. Constraining user movement to floor level and imposing collision-enabled restraint on such movements along corridors will also improve the experience of the navigation interaction. Orbital mode navigation could perhaps be considered for inspection around large objects such as the two antennae. It is reasonable to assume that the cognitive overload required for these navigational mode switching will be easily within the capacity of the target users of this simulation. It appears that a laser-beam or magic wand type interaction technique for object selection, together with an input device with more tolerance to metal and interfering fields are also needed to improve users' engagement and reduce neck and arm fatigue.

## Acknowledgement

The authors wish to thank Ian W. McCrea, Steve Crothers and Chris Davis of Space Science Department, CLRC, for their collaboration and advice throughout this project. Special thanks are due to the trainees for their enthusiastic participation and feedback. Thanks are due to Mark Curtis and Andrew Wells who modelled the initial geometries of the EISCAT virtual environment using 3D Studio™ and MultiGen™ and to Chris D. Osland for setting up the audio and video facilities for capturing immersive user sessions.

## References

[1] T. Winograd and F. Flores. Understanding computers and cognition: a new foundation for design. Reading, MA. Addison-Wesley, 1986.

[2] W. Winn. The impact of three-dimensional immersive virtual environments on modern pedagogy. NSF Workshop: whitepaper. Seattle, Washington and at the University of Loughborough, England, 1997.

[3] D. Zeltzer. Autonomy, interaction and presence. Presence, Teleoperators and Virtual Environments. Vol. 1. pp. 127-132, 1992.

[4] R.G. Witmer and M.J. Singer. Measuring presence in virtual environments: a presence questionnaire. Presence, Teleoperators and Virtual Environments, pp. 225-240, 1998

[5] T. A. Furness, W. Winn and R. Yu. The impact of three-dimensional immersive virtual environments on modern pedagogy. Global Change, VR and Learning. Report of workshops held at Seattle, Washington and at the University of Loughborough, England, 1997.

[6] W.J. Clancey. Situated action: a neuropsychological interpretation: Response to Vera and Simon. Cognitive Science, Vol. 17, pp. 87-116, 1993.

[7] L.L. Crumpton and E.L. Harden. Using virtual reality as a tool to enhance classroom instruction. Computers in Industrial Engineering, Vol. 33, No. 1-2, pp. 217-220, 1997.

[8] R.B. Loftin. Applying virtual reality in education: a prototypical virtual physics laboratory, IEEE Computer, pp. 67-74, 1993.

[9] C.M. Byrne. Water on tap: the use of virtual reality as an educational tool. Ph.D. Dissertation, Dept. Industrial Engineering, Univ. Washington, Seattle, WA., 1996.

[10] Virtual Reality in Training, Computer Graphics World, pp. 225-240, August 1998.

[11] S. Stansfield, D. Shawver, D. Rogers and R. Hightower. Mission visualization for planning and training, IEEE Computer Graphics and Applications, pp. 12-14, Sept. 1995.

[12] R.B. Loftin and P.J. Kenney. Training the Hubble space telescope flight team. IEEE Computer Graphics and Applications, pp. 31-37, Sept. 1995.

[13] K.L. Dean et. al. Virtual Explorer: creating interactive 3D virtual environments for education. Proceedings of SPIE, Vol. 3295, pp. 429-434, 1998.

[14] D. Bowman, L.F. Hodges and J. Bolter. The virtual venue: user computer interaction in information rich virtual environments. Presence, Teleoperators and Virtual Environments, Vol. 7, No. 5, pp. 478-493, 1998.

[15] dVS developer's guide. http://www.division.co.uk

[16] S. Kaufman. New interface metaphors for complex information space visualization: an ECG monitor object prototype. Proceedings of Medicine Meets VR. HITLab Report No. R-97-2, 1997.

[17] M. Rauterberg. How to measure and to quantify usability attributes of man-machine interfaces. Proceedings of IEEE International Workshop on Robot and Human Communication, pp. 262-267, 1996.

[18] M. Rauterberg. Usability evaluation: an empirical validation of different measures to quantify interface attributes. IFAC Man-Machine systems, Cambridge, MA. pp. 467-472, 1995.

[19] M.C. Salzman, C. Dede and R.B. Loftin. Usability and learning in educational virtual realities. Proceedings of Human Factors and Ergonomics Society 39th Annual Meeting, pp. 486-490, 1995.

[20] D. Bowman, D. Koller and L.F. Hodges. Travel in immersive virtual environments: an evaluation of view point motion control techniques. Proceedings of Virtual Reality Annual International Symposium (VRAIS), pp. 45-52, 1997.

[21] D. Bowman, D. Koller and L.F. Hodges. A methodology for the evaluation of travel techniques for immersive virtual environments. Virtual Reality: Research, Development and Applications, vol. 3, No. 2, pp. 120-131, 1998.

[22] M. Wells, B. Peterson, J. Aten. The virtual motion controller: a sufficient-motion walking simulator. HITLab Report No. P-96-2, 1997.

[23] K. Nemire. Evaluating an immersive virtual environment prototyping and simulation system. Proceedings of SPIE, Vol. 3012, pp. 408-416, 1997.

[24] C. P. Chu, T. H. Dani and R. Gadh. Evaluation of virtual reality interface for product shape designs. IIE Trans., Vol. 30, pp. 629-643, 1998.

[25] D. R. Silverman and V. A. Spiker. A usability assessment of virtual reality simulation for aerial gunner training. Proc. Human Factors and Ergonomics Society 41st Annual Meeting, pp. 1218-1222, 1997.

[26] M. Billinghurst, S. Weghorst and T. Furness. Wearable computers for three dimensional CSCW, HITLab Report No. R-97-10, 1997.

[27] D. A. Bowman and L.F. Hodges. An evaluation techniques for grabbing and manipulating remote objects in immersive virtual environments. Proc. ACM Symposium on Interactive 3D Graphics, pp. 35-43, 1997.

[28] I. Poupyrev, S. Weghorst, M. Billinghurst and T. Ichikawa. Egocentric object manipulation in virtual environments: empirical evaluation of interaction techniques, EUROGRAPHICS'98, Vol. 17, No. 3, 1998.

[29] C. Smith. Measuring the usability and effectiveness of advanced learning technologies. Proceedings of Telematics for Education and Training, 1994.

[30] P. Zahorik and R. L. Jenison. Presence as being-in-the-world. Presence, Teleoperators and Virtual Environments, Vol. 7, No. 1, pp. 78-89, 1998.

[31] M. A. Nixon, B. C. McCallum, W. R. Fright and N. B. Price. The effects of metals and interfering fields on electromagnetic trackers, Teleoperators and Virtual Environments, Vol. 7, No. 2, pp. 204-218, 1998.

# Appendix

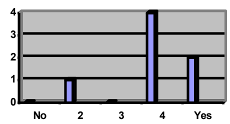
## EISCAT Training –
## VR Simulation Questionnaire

1. Personal details

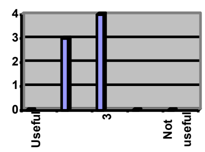
2. Ease of use of the VR system

**3. Effectiveness and usefulness of the VR simulation:**
***Session 1:***

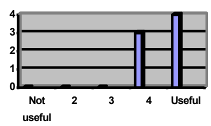
3.1.1 Did the concurrent presentation using the 3D model enhance presentation?




3.1.2 Was being personally immersed in a 3D model useful in reinforcing the lecture material?

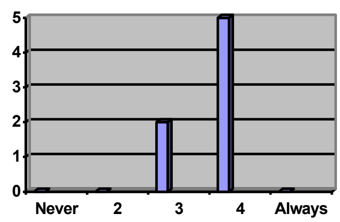



3.1.3 Would it be useful to have the opportunity to use this simulation as a "refresher" before your visit to Tromso?

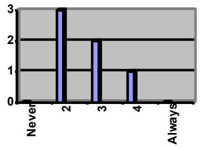



3.1.4 Any comments on any aspect not covered above:

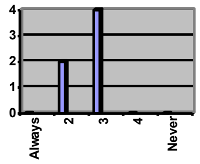
***Session 2:***
3.2.1 Were the visual cues effective in identifying normal operational modes?



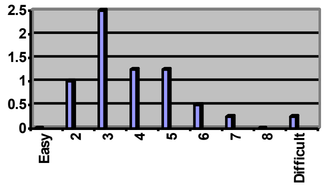
3.2.2 Was the 3D toolbox easy to use?



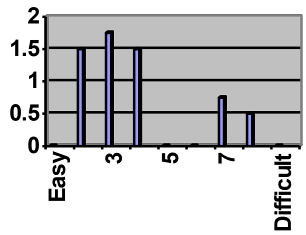

3.2.3 Were the purpose/functionality/use of the 3D widgets in the toolbox self-explanatory?




3.2.4 Were the error conditions easy to spot and correct (Please select all that are applicable)?
Spotting errors (Average for 4 errors):

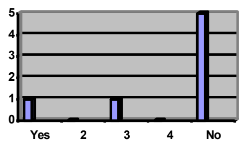



Average for correcting the 4 errors:



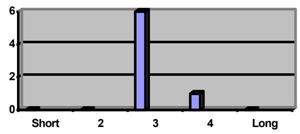

3.2.5 Please give any additional comments, if you have any, on specific reasons for the ease or difficulty in identifying error conditions within the virtual environment.

3.2.6 Would you have preferred to do this session using a monitor rather than the head mounted display?

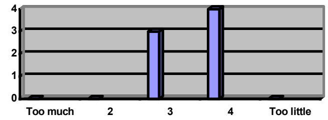



**4. General/Miscellaneous:**

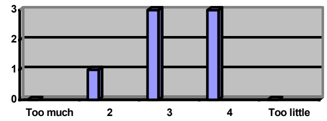
4.1 Was the allocated time enough for the hands-on VR sessions?



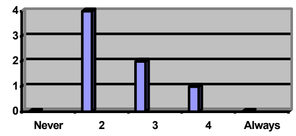

4.2 Was the material covered in each session enough?
Session 1:



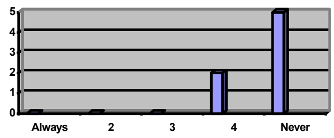

***Session 2:***



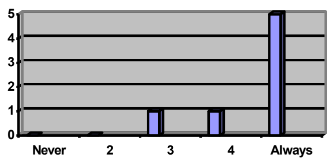

4.3 Did you find the recording of the sessions intrusive?



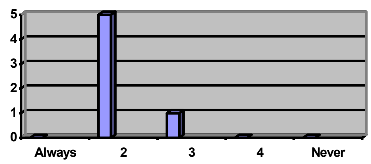

4.4 Did you find it difficult to concurrently talk through your thinking/decision making process?



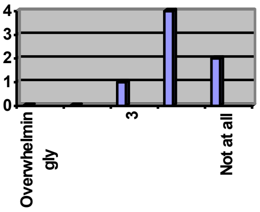

4.5 Did you feel psychologically comfortable with the set up?



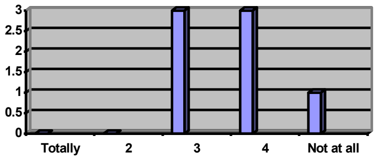

4.6 Did you feel comfortable carrying out the tasks immersively in the presence of one of your trainers?



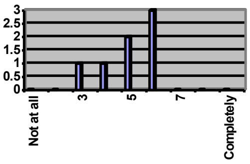
4.7a Did you feel a sense of "being there" during the immersive navigation?



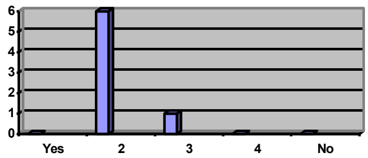

4.7b Did the sense of "being there" contributed to your absorption in the tasks?



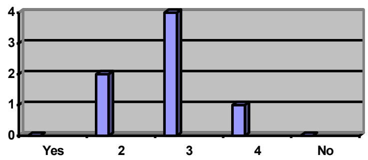

4.7c Did the sense of "being there" make you unaware of your real physical surroundings?



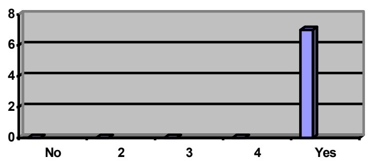

4.8 Do you feel that the VR simulation enhanced your training?



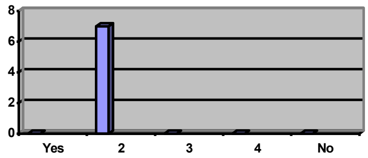

4.9 Would you have preferred more of the training material to be covered via the VR simulation?



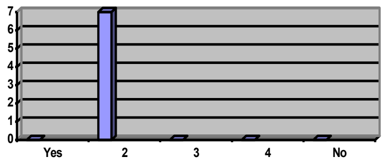

4.10 Were the instructions given for each VR evaluation session adequate?



4.11a In spite of any device level limitations has the experience of using immersive VR simulation sufficiently interesting, useful and enjoyable as to enhance the learning experience?




4.11b Would you recommend its continued inclusion in future courses, for instance?




4.12 Any comments on any aspect not covered above:

# "Males say 'blue,' females say 'aqua,' 'sapphire,' and 'dark navy.'" The Importance of Gender in Computer-Mediated Communication

Frank Schaap
MA student Social Anthropology
University of Amsterdam
Nova Zemblastraat 7-F
1013 RJ Amsterdam, The Netherlands
Fts@xs4all.nl, http://people.a2000.nl/fschaap/

## ABSTRACT

In this paper I will look at how social and cultural concepts of gender are performed and re-enacted in text-based virtual realities. Through in-depth interviews and the analysis of two years worth of online role playing logs, I will look at how cultural concepts of gender influence the construction of a character and what the consequences and repercussions of performing gender in a role-playing MOO are. The investigation will show that it is of great importance for the design and implementation of virtual realities to consider the everyday conceptions gender and of social reality in general.

**Keywords:** MUDs, MOOs, gender, performance, relevance of everyday notions for design and implementation of virtual realities

## 1 INTRODUCTION

This paper explores the importance of gender differences in computer mediated communication (CMC) for the design and implementation of virtual realities. It will do so by examining the social constructions of what is considered a convincing female or male character on a role-playing MOO. I will first discuss some methodological issues regarding anthropological research in cyberspace and my own research in particular. Then I will detail a few theoretical issues about gender. Next I will turn to the findings of my research and finally I will focus on the importance of gender in CMC.

## 2 ANTHROPOLOGICAL RESEARCH IN CYBERSPACE

The anthropologist, the student of the culture of Man, often travels far, far away, scouring the far reaches of the earth in search of hitherto undiscovered tribes and lands. In carefully chosen, scientific words the anthropologist tells his more or less implicitly heroic story of penetrating deeply into dangerous tropical forests, scaling class four mountains and living under harsh conditions, always, eventually, finding a people with fascinating customs. The anthropologist brings back a story of an other world where other people live very different lives. The ideal of ethnography, of "writing culture," is not only to recount, map, index and explain these Others' lives, although it does that in great detail, but ultimately to tell of how these people experience their world and how they have formulated other answers to the questions of life.

The preferred mode of research is called "participant observation." While theoretical debates have raged over the (im)possibility of both participating and observing at the same time, usually it means that the anthropologist lives with the people whose culture he studies for a prolonged period of time, learning as much as he can about their beliefs, habits and mores. "Simply" participating and observing is of course complemented by in-depth interviews, more wide ranging surveys, use of historical materials, etc. The main focus of the anthropologist however remains a small group of people, usually a village or a particular sub-group, and his understanding of their mode of living stems from the prolonged submersion in their culture and the in-depth, often very personal contact with a relatively small number of informants. Because anthropology has always been confronted with the "other," its major strength is the way its research tools and methodology allow the researcher to dynamically frame new and "other" realities.

In recent years anthropologists have also ventured into studying various Western communities and sub-groups "at home" [7], instead of focussing solely on people in exotic, far away locales such as the Brazilian rainforest [1] and the Trobriand islands in the Western Pacific [11]. With the rising popularity of the Internet anthropologists (and other social scientists) have started to study the social and cultural aspects of "life online" [12]. The ongoing research has not yet yielded a full-length monograph about a particular online community, however, it must be noted that Julian Dibbell's recent book *My Tiny Life* [3] offers a fascinating and thorough account of what life in a virtual text-based world can be like, though it lacks the theoretical and methodological framing characteristic of an anthropological monograph.

The setting of my anthropological research is a role-playing MOO[1] called *Cybersphere*. Role-playing in the surroundings of a MOO can be understood as a kind of impromptu, unscripted, textual playacting. A player makes up a character that s/he wants to play, deciding on a history for the character, its gender, particular character traits, its appearance, etc. The player in a sense is the character's author/actor, as its creator as well as the person who brings it to life by "playing," i.e. typing out/in the actions and reactions of hir[2] character when interacting with other characters (who in turn are played by other players).

There is a small but crucial difference between a role-playing MOO and a social MOO, such as, for instance, LambdaMOO. On a role-playing MOO you are supposed to play a character and while the actions and utterances of the character are written and performed by the player behind it, the player is not responsible for them, s/he is "merely" acting. The player must of course stay within the boundaries of "good" or accepted role-playing or the situation will become quite different and s/he *will* be held accountable for the actions of hir character. The character itself on the other hand can always be held responsible for its actions by other characters; there is, in theory at least, a clear division between interactions In Character (IC), between characters, and interactions Out Of Character (OOC), between the players of those characters. The character, in short, is, at least during the role-playing, a fictitious character, animated by the player behind it, but not necessarily representative of that player. On a social MOO on the other hand, since it is not explicitly a game, the player presents or performs a persona. The persona, whatever form it may take, is always assumed to represent the player in one way or another and not is not a completely fictitious character. The persona, in short, is a representation, however partially or warped, of the player behind it.

I have chosen a role-playing MOO as the setting for my research exactly because the players must make an effort to present a convincing character. A large part of the players' efforts go towards the "suspension of disbelief" and an important part of effecting that is making sure you present a coherent and convincing character that fits the theme of the game. *Cybersphere* is a cyberpunk world set 30 years into the future and although there are cyborgs, androids and advanced robots the vast majority of the players plays a human character; there are no elves, goblins or bulbous-eyed aliens with glowing fingertips to speak of. This was also an important factor for deciding on *Cybersphere* as my fieldwork location, since the primary interest of my research is the performance of gender and how players present a convincing female or male character. It would undoubtedly be fascinating to learn about the particularities of elvish or gnomish gender categories, but because, as an anthropologist, I am primarily interested in how humans shape their world, I decided to first study the performance of human gender in an online role-playing game.

The research methodology of participant observation has been successfully applied to anthropological research "at home" and as my study has shown there are no insurmountable problems with applying it to an online setting. Requesting a character and joining the other players in their world is probably even easier than booking a flight to Wherever and getting a hotel room or a place to sleep on the mud floor. There is however one important difference. When the anthropologist arrives in a small community the locals will either have already been briefed about his stay with them or they will know all about it upon his actual arrival. Online, however, nobody knows you're an anthropologist. Unless, of course, you tell them. Participating and observing therefore takes on a whole other dimension as long as the anthropologist remains anonymous. Participating actually means playing your character, suspending your disbelief and acting as your character would. Observing on the other hand can take place so unobtrusively as would never be possible in a real setting. When you don't act or

[1] MOO: MUD, Object Oriented: a real-time, multi-user, text-based virtual reality; for a more detailed explanation of MUDs see Curtis [3].

[2] I'm using "hir" here as a gender-neutral contraction of "his" and "her," instead of resorting to "her/his." I first encountered it in Sullivan and Bornstein's novel [15], but the use of a gender-neutral (possessive) pronoun seems to have spread quite fast and wide on the Net, where gender is a more malleable quality than in real life. The anonymous anthropologist introduced earlier is, as you may have noticed, male.

react to what goes on around your character, people will almost completely disregard the fact that you're in the same room as they (their characters) are. They will assume that your connection has gone bad but that your character for some reason stays logged on, or that you have gone away from the keyboard (AFK) to have lunch or something.

For my research then, I have played my character(s) for well over two years, primarily on *Cybersphere,* but also visiting other MOOs in other guises. Making use of the log function of the telnet-client I was able to keep verbatim logs of every encounter and every conversation I had while logged on. After an initial period of participating anonymously, I publicly announced that I was doing a research and I asked the players of *Cybersphere* for their help on a questionnaire I had prepared. After that I conducted personal and in-depth interviews. I chose to do "open" interviews, not keeping strictly to a prepared set of questions but giving the informant a large degree of freedom to influence the conversation, while still keeping a list with points of interest and prepared questions at hand in case the interview would slowly grind to a halt. This body of data was then complemented by a slightly different version of the questionnaire that I posted to a couple of relevant news-groups in order to gather comparable material from players outside of *Cybersphere* and the material I collected from different websites.

## 3 GENDER AS PERFORMANCE

As I mentioned before, the main focus of my research is the "performance of gender." In the next paragraph I will discuss the opinions and criteria of the players themselves on what constitutes a convincing character, focussing in particular on the criteria for what constitutes a convincing female or male character. After discussing the theory and the findings of the research I will turn to the importance of gendered presence in virtual realities. For the theoretical purposes of this paper I will draw on the work of Judith Butler [1] and Suzanne Kessler and Wendy McKenna [10].

Kessler and McKenna argue that gender is not the natural expression of a biological or physical quality that every individual possesses, but rather a social construction by which "[w]e make a *gender attribution,* that is we decide whether someone is male or female, every time we see a new person" (1978: 2; italics in original). While not denying the very materiality of the body, they argue that

> [W]hen we "do" gender in particular instances we are creating the reality of gender as a construct. It is apparent, though, that we do not only create gender as a construct, but we create the specific categories of "female" and "male." We must be doing more than gender; we must be doing *female* or *male* gender. (Kessler and McKenna 1978: 154-155; italics in original)

Butler then opens up the possibility for a detailed inquiry when she renders the construction of gender as a performance, a re-enactment of social and cultural imperatives for proper behavior.

> [G]ender is in no way a stable identity or locus of agency from which various acts proceed [sic]; rather, it is an identity tenuously constituted in time – an identity instituted through a *stylized repetition of acts.* Further, gender is instituted through the stylization of the body and, hence, must be understood as the mundane way in which bodily gestures, movements, and enactments of various kinds constitute the illusion of an abiding gendered self. (Butler 1990: 270; italics in original)

In the case of role-playing characters in a virtual world this rendering of gender suggests that by examining the "bodily gestures, movements, and enactments of various kinds" the way in which gender is constructed may be revealed. The fact that a character is the performance of the player, plus the fact that this performance takes place entirely in written exchanges offers the researcher an unique vantage point; the written exchanges must be expressive to be compelling and the text will reveal more or less conscious conceptions of (gender)identity and the understanding of what socially appropriate behavior and viewpoints are.

## 4 FINDINGS OF THE RESEARCH

Central to role-playing is the attempt to present your character as convincingly and fully as possible. The strategies for achieving this are varied and to a large extent depend on the individual player. There however seems to be a large degree of consensus on how cross-gender characters can be recognized. A cross-gender character is a character of a gender other than the player who plays it. One 28 year old male[3] player

[3] As a reminder: I merely assume that this player indeed is male in real life and that "he" was speaking the truth when we conversed out-of-character, i.e. as our everyday selves. Although one learns to read the cues whether someone is serious or not, one can never be entirely sure. On the bright side, in a real life situation I might have been meeting face to face with a man whom I didn't recognize as a female-to-male cross-dresser. If this isn't enough to convince you, consider the fact that there is at least one reported case in which the

noted:

> Playing the opposite gender is the ultimate in role-playing challenges. It is much like trying to play another race (elf, dwarf, etc) but is more difficult, in that nobody can actually say "Hey, you're not really an elf!" In other words, not only is it difficult to do, but you can be found out more easily than any other role-playing situation.

The players' opinions about cross-gender characters are interesting in that they reveal a lot about the implicit and explicit conceptions of gender. In a perfect world a character would be judged only by its performance, but in reality, as we shall see, the (gender)identity of the player behind it also plays a part in the other players' perception of it. The mundane ways in which gender is performed on or through a fictional, textual character of course differs from how it happens in real life. This has mostly to do with how the virtual body is constructed in virtual reality, a matter I will return to more in-depth in the next paragraph. The same player I just quoted offers the following details on how to play a character and which "mundane" aspects of a role-playing character to pay attention to:

> I mostly pay attention to a very detailed description, and make certain to have a highly developed personality. I try to keep the personality in mind, using certain key phrases, phrases and expressions that I don't use, as though they were by habit. Same with body language. I make sure I have several "hooks" which I can use to remind myself exactly who and what I am playing. In many cases, these "hooks" are very stereotypical.

For the purposes of this article I shall focus mainly on the characters' descriptions. The character's description turns out to be a very important aspect of conveying the desired impression of that character, since it usually is one of the first and most "tangible"[4] features one learns about another character. A player from *Cybersphere,* whom I shall call Tôsama,[5] explained:

> Initially, while the character is becoming 'known' to the other players, [the description] is everything; over time decreasing in importance, as others see whether the player's game play is congruent with the description.

The description, upon closer inspection, reveals itself as a highly codified piece of discourse that conveys its meaning in specific ways. Part of the socialization process of a newcomer to the MOO is learning how to read and write a character within that discourse. The rules for successfully reading and writing a character description are largely implicit and they are not set down in the helpfiles of the MOO. However, on occasion opinions are voiced on what is acceptable and what is not. Lillith, a young woman raised on movies, music and literature who lives in *Cybersphere,* wrote on one of the public message-boards:

> I've composed a very short list of things included in people's [descriptions] that, (and I point out immediately that this is strictly my opinion and by no means a statement of definite fact) due to their repetitiveness and/or lack of creativity and/or use of very tired cliche's, are the sort of things that make me grumble/yawn/sigh upon seeing them. They are as follows: "He/She has wisdom beyond their years." It obviously isn't beyond their years if they have it, is it? "He/She has eyes that seem to pierce straight through you, or, if you prefer, "He/She has eyes that seem to look right into your soul." I'd sure hate to be in a room with two people piercing through each other with their eyes. That'd get messy, I think. "He/She radiates an aura of pure evil." Sorry, I can't think of anything witty about this. "blah" is the only thing that seems to enter my head. "He/She is amazingly/ravishingly/arousingly/etc. beautiful/handsome." What if you're not my type? What if you're blonde, brunette, or whatever, and it (for some reason) is a major turn-off for me? People generally decide for themselves whether something is beautiful or ugly. Well, unless you watch too much TV. But that's another argument altogether. There are probably many others, but these really stick in my craw. I'm sorry if I'm directly offending anyone who might be reading this, but I think we're all capable of being a lot more original. Mainly, I just hate it when people decide for me

---

whole village plotted against the visiting anthropologist and supplied him with completely made-up information about the village's lineages.

[4] Unlike social MOOs where one can change one's description at will, characters on *Cybersphere* and other Role-Playing MOOs heavily rely on their descriptions to create and consolidate a particular identity and the descriptions of existing characters are usually not changed significantly over time. That is why I call the description "tangible."

[5] Here I am referring to the player by name of his character, a fairly normal procedure in a world where keeping track of certain people through the procession of their characters is hard enough as it is. Sticking to one name for both player and character and making sure you use the <OOC> command for distinguishing your Out Of Character interactions from your In Character (IC) conversations proves to be the simplest solution.

> in their [description] how I view them.

Before continuing the examination of the descriptions, I would like to point to the disclaimers that flank the statements in the message. As Susan Herring shows in an article [6] that analyses the differences in "…the communicative behaviors of participants…" of mailing-lists, the "…women who participated in the discussion… displayed features of attenuation – hedging, apologizing, asking questions rather than making assertions – and a personal orientation, revealing thoughts and feelings and interacting with and supporting others." My own examinations of the message-boards of the MOO collaborate Herring's findings on the way that women compose their messages, but also confirm Herring's findings that a large percentage of "…the messages posted by men made use of an adversarial style in which the poster distanced himself from, criticized, and/or ridiculed other participants, often while promoting his own importance." Herring notes consequently:

> While these styles represent in some sense the extremes of gendered behavior, there is evidence that they have symbolic significance above and beyond their frequency of use. Thus other users regularly infer the gender of message posters on the basis of features of these styles.

Returning to the analysis of the descriptions, my material suggests that players generally assess other characters' descriptions in either of two ways, which I will call here "good" and "bad" descriptions. "Bad" descriptions are filled with clichés (such as the ones Lillith mentions) and grossly overdone stereotypes and reveal comparatively little about the character itself. Mostly it indicates the attitude of the player towards the use s/he sees for hir character, which can be, as Graye suggested, "amusement, silliness, RP [role-playing], social activity in an OOC style." The stereotypes are usually dead give-aways, as one twenty year old female player described them:

> Female: tall, shapely, long legs, blonde hair, huge breasts, dressed like a skank, etc. Male: looking like some recent product of the GI Joe toy line... Musclebound beyond belief, huge crotch area, etc.

One description, that as an example is just too good to pass up, starts like this:

> Vampina
>
> A young woman with an unconcealably large pair of breasts. They jut out from her body for feet, covering the entire front of her torso. Her hips flare out nicely, and she walks surely.

While this description obviously takes it over the top, tongue-in-cheek in my opinion, together with the next much more serious attempt it presents some insight in how these descriptions are constructed.

> John
>
> A young man who is tall, wide in the shoulders, narrow in the waist and hips, with a rock-solid body. His hands are enourmous, formidable-looking weapons. His broad face seems to have been carved out of granite---and with some difficulty, with much breaking of chisels and breaking of hammers. His pale-blue eyes, clear as rain-water, regard the world with icy suspician. He wears an eagle tatoo on his left hand. His head has been shaved down to almost the scalp.

Descriptions like these surely conjure up an image, but they offer little detail that might be taken as indicative of the character's personality. As Tôsama points out, when you start getting acquainted with a new character, you don't have anything to go on but hir description. If it is an "off the rack" description it often fails to pique the interest of the other character, it doesn't offer enough salient details to draw the other player in, to make hir curious about *who* this new character is; often a bland description is taken to be a sign that either the character is not a very interesting personality or that the player has other intentions with the character than to role-play.

Highly stereotypical descriptions filled with clichés are usually considered as failed attempts to either play a character of the opposite gender or as compensation for something lacking in the player's real life, as the comments of an 18 year old male player illustrates:

> If I played a woman and was obsessed with how big my breasts were in my description the whole time, then it wouldn't be real convincing. Actually, though, I do know women who like to play attractive flirty women to live out what I think is an inner-desire of theirs who would most likely do the same thing as in the past description.

So even descriptions that rely on stereotypes and clichés don't necessarily mean that the player is playing a cross-gender character, maybe s/he is new to the game and has not yet fully incorporated the discourse or maybe s/he is a tad unimaginative or doesn't care enough about it. Although "bad" descriptions may be a first indication that one is dealing with a cross-gender character, and surely most players are at least a bit more aware of that possibility when encountering such a character, it turns out that in order to determine the other player's real life gender most players pay particular attention to certain details of a description or a character's behavior. "Good" descriptions are imaginative, evocative and usually finely crafted pieces of text. The way in which these "good" descriptions are primarily (and the "bad" descriptions secondarily) assessed is by locking on to those details that a player thinks are good indicators of the player's (gender)identity. Graye explained it like this:

> I pay particular attention to what is emphasized in another character's description; sometimes, it's stance, sometimes, it's intelligence, other times it's the face, or an overall impression the player wishes to convey with [the] character. In a character's gender, I look for emphasis on sexual parts of the body, syntax and word use when describing the character (ass vs rear, tits vs chest, muscular vs built, chiseled features vs no features at all), and frequently the clothing a character wears, and how IT is described. Clothing is often indicative of a player's attitude toward their character and whether or not the gender of the character is the same as the player's IRL gender.

How then would a player "read," for example, the description of the clothing of a character as a clue to the player's gender. A female player by the name of Shadowschild offers the following:

> Details of clothing, male players tend to put less detail into their clothing and appearance, female players tend to emphasize the details... i.e. males say "blue" females say "aqua", "sapphire", and "dark navy". Also, females tend to be more careful about what they say and how they say it. The majority of bad spellers and characters with bad grammar that I have run into have been male. [T]his also goes for the [descriptions]... females tend to check and double check the [descriptions] to make sure it looks right.

To illustrate the points Shadowschild makes her description follows:

> Shadowschild
>
> A young, attractive, almost elvish looking female, with long dark hair that is sharply contrasted by a lock of white on her left temple. She has dark, dark blue eyes, so dark as to almost be black, and they are lit from within by a mischievious twinkle.
>
> Her hair is held out of her eyes by a black, military-style beret bearing a stylized silver VF pin. A sapphire blue tube top covers her chest and arms leaving her belly and shoulders bare. A finely wrought silver bracelet graces her left wrist. Delicately etched across the front, on black enamel, is the insignia of the Violent Femmes. Her belly is soft and smooth... just the barest hint of muscle shows, despite her obvious fitness. Her flanks curve gently inward, the skin soft and white... Shadowschild appears quite comfortable in a faded old pair of blue jeans. She has on a pair of soft black boots that come only up to her ankle.

Attesting to her eye for detail is her use of a little known feature of the MOO which let's you add descriptions for individual body parts, such that when for instance you would look at her hair, it would read:

> Dark brown, with a strikingly placed streak of white on her left temple. Reaching all the way down to the middle of her back, it is very soft and smells of apples.

The scope of this article cannot possibly exhaust all the possible and sometimes highly individual details that players will pay attention to in order to gain information about the "validity" of a character and the "identity" of the player behind it. Players indicate that the first time they meet a new character they usually take the character and its description for granted, bar of course too obvious clues, and settle for the fact that the player is of the same gender as the character. Not having access to an actual body as a reference makes you a bit suspicious, but there's no need in being totally paranoid about it. After all, as Shadowschild noted in response to my question whether the details she described were necessary to convincingly perform a female character:

> Among other things. [The attention to detail] is one thing... by itself, I would not commit to saying yea or nay about a character... but it is one thing that, combined with others, might make me wonder.

The most interesting thing to note here is that the evaluation of the believability is applied to the description and performance of the character. If a female player would play an aggressive female character with a little detailed, sloppily written description, the suspicion would soon arise that the player might not really be female. Would a male player play a particularly caring, maybe effeminately and immaculately described male character, other players might wonder if there wasn't a female behind the mask after all. So the fact remains that whatever your real life gender, your character must be described and performed within a discourse that imposes quite strict rules on what constitutes a convincing female character and what a convincing male character.

Part of the motivation for this line of inquiry was the following question that Kessler and McKenna pose:

> Rather than asking people to notice or describe the typical and atypical behaviors of their own and other gender... information could be gathered on which, if any, nonverbal behaviors are "conditions of failure." In what nonverbal ways could a person behave such that her/his

gender is questioned? (1978: 157)

Rather in line with Kessler and McKenna's own findings that there ultimately is no single defining quality that distinguishes a man from a woman, my findings, as Shadowschild also noted, would suggest that neither is there one single quality or action that all by itself is a condition of failure to present either a male or female character. This all points to the fact that gender is not (a part of) the identity installed/instilled only once, when a baby is born, "Look, it's a boy/girl!" but rather that gender is, as Butler says, "…the stylized repetition of acts through time…" (1990: 271). Gender must be re-enacted, performed, re-affirmed with every social interaction because it is not a natural or biological quality, but a social construction. There is no single condition that will affirm one's gender, but there's also no singel condition of failure, because, as Kessler and McKenna note:

> If the conditions of failure could be described, then people could be any gender they wanted to be, at any time. (1978: 157)

And in a virtual environment, where the performance of gender includes the performance of a perfectly fitting virtual body, a player can even be questioned on hir real life gender when s/he plays a character that corresponds with hir own gender. The freedom to play with gender in a virtual environment seems, ironically so, constrained by the player's real body. Not because s/he is not capable to imagine an other body in a convincing way, but rather because others will relate to hir as defined by hir real life gender, unless s/he conceals it. George Eliot, a 21 year old female wizard on Cybersphere, who took her name after the nineteenth century author Mary Ann Evans' pen-name, played a male character for a while, not divulging her real life gender.

> George Eliot says, "I set out to do it... people react very differently when they know you're gender-bending."
> You ask, "in what way do they react differently?"
> George Eliot says, "It's hard to define, but the interaction is less authentic... they tend to talk to you as a member of your RL gender rather than as your character's gender."
> George Eliot says, "They're more reserved... there's still kinda a stigma attached to gender-bending to a certain extent, although it applies less to people going my direction than it does to guys playing women."
> You ask, "mmm, a stigma. what sort of stigma?"
> George Eliot laughs. "Couldn't define it because I don't feel it... what I noticed though is that after people found out I was a chick, a lot of their IC interaction with [my male character] stopped or changed... women who were hitting on him stopped, for instance. *grin*"

The question then remains, since I so ostensibly set out to describe the players' attempts to distinguish between a male playing a male character and a male playing a female character (and of course between a female playing a female character and a female playing a male character), why that is so important for players to know?

A part of that answer is simply that without an actual body present for reference, most people feel at least slightly at unease. Another part is that playing a character is often a very personal experience and the IC and OOC experiences are not that readily separable. The relation between two characters thus implies the relation between the two players behind them. Gender is such an integral part of how we interact with people that its absence, or rather the uncertainty about it, causes trouble with how we should relate and thus how we should communicate with that other person. If the gender is not simply available, or if it is "neutral," as is the "spivak" gender category on MOOs, then everything we do or say needs some extra thought, since the communication does not fall within regular, everyday patterns of speech, as a nineteen year old male player indicates:

> Playing a spivak on Cybersphere went down very well, I feel that I played it successfully (until I was bored of the character.) I had many people, annoyingly, asking what my real sex was, as they supposedly couldn't pick it from my role-play style. It was fun, but quite a bit of work to keep up.. it's that extra little piece of thinking before you say or do anything that can get tiresome..

If the player who chooses to play a spivak character has to put that extra thought into everything he says or does, then everyone who interacts with the character without knowing the player's gender will also have to put in that extra bit of thought. Uncertainty about or absence of gender thus influences communication as surely as the presence of gender does and in order to facilitate the communication players try to resolve the question of the other player's gender, so that they may speak easily and in/with their habitual speech patterns.

## 5 Importance of Gender in CMC

"In cyberspace you can be anyone you want to be," goes the popular saying. In the previous paragraph I have argued that it might not be so easy to convincingly project the desired person(ality) into cyberspace; it will after all still be judged by our common-sense notions about what constitutes a socially and culturally acceptable person. Not only do we apply our everyday ideas about identity to the “people” we meet in cyberspace, we also bring everyday expectations, habits and behaviors with us into cyberspace. In this paragraph then, I will turn to the implications of applying the “old rules” to a “new space” and argue why gender (and indeed our common-sense understanding of the world) should be a major concern when designing and implementing virtual realities. I will consider three topics here, to illustrate the point: the case of a virtual rape, the different reasons and expectations that people have for playing on a role-playing MOO, and how social notions of communication and organization are embedded in and reflected by the computer code that runs our computers and the MOOs we inhabit.

Julian Dibbell's article "A Rape in Cyberspace: How an Evil Clown, a Haitian Trickster Spirit, Two Wizards, and a Cast of Dozens Turned a Database into a Society" [4] has been quoted a lot over the years. One has only to feed a search-engine the words "julian dibbell rape" to watch the fall-out of his article appear. Undoubtedly this text will be added to the growing number of interlocked hyperlinks soon. The number of reactions is not so strange when we consider that the article is very well written and evocative. It is about a delicate matter that touches on, or maybe even cuts into that very "realness" that the virtual sometimes possesses. The article tells us of how one night in the virtual world LambdaMOO the now infamous Mr. Bungle raped several other characters. In a world that is made from text, that what you say happens, actually happens. So when Mr. Bungle fashioned a voodoo doll that allowed him to manipulate other characters and have them perform rather unsavory acts upon himself, others and their own bodies, he raped them.

As in real life, rape in cyberspace is not primarily a sexual act, but an act of aggression. Aggression that is mainly directed at women.[6] In a virtual environment this means either female presenting characters or characters of female players. The acts of aggression may be performed on a textual, virtual body, the aggression is directed at the player behind the assaulted character. A “virtual” rape thus transgresses the divide between real and virtual, the textual acts inscribing real pain on the victim.

Of course action is undertaken against those who perpetrate these crimes. On LambdaMOO it resulted, as Dibbell details in his article, in the unrevocable removal of the Mr. Bungle character from the MOO. Subsequently a ballot system was implemented on which all players could vote and for which all players could suggest topics to vote on.[7] LambdaMOO thus became the first (and only) democracy on the Net, leaving behind the enlightened rule of the wizards (those who have special programming privileges, more or less comparable to system-administrators).

On *Cybersphere,* a role-playing MOO and not a social MOO like Lambda, another way was devised to deal with the threat of virtual rape… after the first victim had been made. *Cybersphere* is a dark, post-apocalyptic cyberpunk world where gangs control the streets. Technically it would be quite possible to implement a democratic system on *Cybersphere,* but since it is a virtual world geared towards role-playing and providing the players with a highly immersive imaginary environment, this would probably seriously detract from the role-playing and would do the suspension of disbelief no good. This prompted an In Character solution that would fit in with how the world works.

> Shadowschild says, "*serious look*[8] Actually... did anyone ever tell you why VF[9] was founded?"
> You say, "*nods* Arielle gave me the once over... :-)"
> Shadowschild says, "She told you about the VR rape of a newbie female character/player?"
> You say, "*shakes his head* nope... not that... really?"
> Shadowschild says, "Really... guy got hold of a newbie and dragged her to a locked room. She didn't know how to make the MOO work and couldn't get away. That, combined with the 'predictable' attacks and insinuations by male characters on/at female characters prompted the creation of VF."
> You say, "*nods* jeez... that's one of the first things happening to me here.. eh, not getting raped, but getting pushed around and into a corner till the guy that was showing me around town the very first day could kiss me..."
> Shadowschild says, "That was way more

[6] I have not yet heard or read about a virtual rape committed by a woman.

[7] I am over-simplifying the rather intricate legal/voting system here, good sources on the mechanics of democoracy and virtual law are Dibbell’s book [3] and two articles by Jennifer Mnookin, [13] [13].

[8] The asterisks are often invoked to indicate an "emotion," *happy*, or even a pose, *nods and brushes some hair from his forehead*. This convention (some people use different symbols, such as <brackets> or -dashes- to accomplish the same) let's players quickly qualify a particular statement. These interjections function much like the well known smiley-faces, but offer a bit more freedom of expression.

[9] VF, Violent Femmes.

common a couple of years ago... and that is why VF was formed. Any VF member who sees that is obligated to stop it by any means possible."

And so the *Violent Femmes (VF)* were born. The VF is an all female gang, that never so much participated in the various skirmishes that raged across the MOO, but rather provided the female players, by way of providing protection and a safe haven for the characters, with some leverage and deterrence against evil minded players. What happened with the virtual rapist remains unclear. Shadowschild said that probably he (his character) had been hunted down and killed so many times that the player must have run out of in-game money and thus could not afford any more clones (which act as “extra lives”). If a character without a clone is killed, it is irrevocably removed from the game, just as Mr. Bungle was by wizardly intervention.

Turning to the reasons and expectations players have when playing on *Cybersphere* I find myself in a bit of a predicament. I have no specific data to present apart from the observations I made, which I can only present here as rather generalizing statements.[10] Still, I believe that this is an important point and so you will have to take my word for it (and by now we should all realize the relative merits of that in the age of body-less gendermorphing wireheads).

Fact is that, by and large, I have noticed female players seem to be more interested in role-playing “social situations,” friendships, affective relationships, social intrigues, etc. where the playing of the role, the identity and psychology of the character and the nature of the interaction are stressed most. Male players on the other hand, most often seem more intent on achieving more material goals, such as the collection of in-game money, rare items (preferably weapons) and the attaining of a position of power; of course role-play is still integral to the way male players must reach their goals, but often their role-play is much more utilitarian, more direct, phrased in less elaborate ways and of shorter duration. For female players, it would seem, the role-playing itself is the goal of the game, while for male players the role-playing is a means to an end.

It is probably not incidental that the differences in style of role-playing are reminiscent of the differences in the style of descriptions. It is also interesting to note that in her paper about gender differences in communication on mailing-lists, Herring illustrates her basic claim,

> [f]irst, that women and men have recognizably different styles in posting to the Internet, contrary to the claim that CMC neutralizes distinctions of gender; and second, that women and men have different communicative ethics -- that is, they value different kinds of online interactions as appropriate and desirable.

Regarding the broad picture painted here and collating the material presented, the conclusion that there are two distinguishable styles of communication presents itself. Herring duly notes:

> If our online communicative style reveals our gender, then gender differences, along with their social consequences, are likely to persist on computer-mediated networks.

The common-sense assumptions about the social reality we live in however, do not always present them as obviously and readily as in the performance of role-playing characters in a cyberpunk MOO. Everyday notions about social relations, hierarchy, appropriate modes of discourse and gender relations are much deeper embedded in the very fabric, in the very program code that makes the Net and MOOs possible. Because it is so deeply embedded often the effects and consequences are overlook and even if they are not overlooked, they are much harder to remain aware of.

In an impressive article entitled “Living Inside the (Operating) System: Community in Virtual Reality” John Unsworth [16] analyses how the social, cultural and economical circumstances of a particular historical moment (the late sixties till early eighties) first shaped the Unix operating system and later also the MOO program-code that make it possible to inhabit a virtual world. Unix and the program underlying MOOs are made of code, code that is not only a higher-order representation of the instructions for those bit-shifting machines, but a code that also encodes our notions about how the world works, a code that recreates our understandings of the functioning of the everyday world in a simulacrum which we then can use to share files, exchange information or inhabit as a virtual character.

Because of that particular historical moment in which Unix was “conceived” and written at AT&T’s Bell Labs it features a highly hierarchical file-system. Built to give multiple users shared and simultaneous access to expensive mainframe computers, the operating system that grants access must organize the access in such a way that no conflicts between the various users arise. The administrator, called “root,” owns the system and down through a treelike structure root grants or denies access and file-manipulating privileges. While at the literal level reflecting the corporate culture and the capitalist notion of ownership, other influences made Unix a uniquely

[10] I can’t present the data in part because I still have a large part of my material to analyse and in part because I haven’t asked my informants direct questions pertaining their reasons and expectations for playing on a Role-Playing MOO *in general,* which I believe to be an oversight.

social system that fostered communication. Accommodating multiple users in a cooperative environment Unix featured extensive network communication capabilities. Unix was distributed virtually free, but without user-support, which in turn spawned an informal self-supporting community of users, who post and reply to questions about the use of Unix on Usenet.

When in the early nineties the first MOO code was being developed, virtually the whole Internet relied on computers running some incarnation of the Unix (or Unix like) file-system. The MOO code is also constructed in a modular fashion, in which the individual objects (for which the Object Oriented stands in the acronym MOO) that make up the virtual world and its inhabitants, inherit qualities and privileges in a tree-like hierarchy similar to that of Unix. Like Unix the MOO code was developed at a large company, Xerox, and both projects were made possible by the incredible accumulation of wealth, that granted the engineers of these companies the freedom to "play" around in order to see if something useful came from it, and it was the same accumulation of wealth that made it possible that both projects were distributed (virtually) for free.

The very code of the operating system and the MOO program, based on the historically and socially located understanding of social reality, structures in the most elementary sense the ways in which the virtual environment is presented to the user/player, what the user/player's relationship is with other user/players and the files/objects present, how and by what means the user/player is allowed to communicate with other user/players, and even, in a sense, what the user/player may or may not imagine to be real in that virtual environment. Designing and implementing virtual realities is thus not merely a technical endeavor but also a social one, and if we want those virtual realities to offer a more equitable ground for communicating, playing and working than the real world in which we live, we better start taking our cues from the way we construct our really real reality before simply re-constructing it as a really virtual reality.

Shadowschild after all, made it quite clear that the attention to detail paid to women not only operates in an online role-playing environment, but that it is the same process that operates in the real world:

> [P]art of the image, at least for females, is wearing what is socially defined as 'feminine' clothing and accessories. Skirts, perfume (which I am allergic to!), earrings, makeup... [I] was a vet assistant... Now I am a copyeditor... but I have found it very difficult to make that change, because I have not been able to afford the expensive clothing and accesorries required... and because I refuse to wear a skirt. I have been told that I cannot... CANNOT apply for a job... not because I was not qualified.. not because I was crude, dirty or generally unpleasant to be around... but simply because i would not wear a skirt. I insisted on wearing slacks or pants.

The attention for detail, not the grossly overdone caricatures, but that everyday expectation that women in particular functions should wear a skirt, is exactly the same attention for detail that Shadowschild herself uses as a good indication whether a female character is a convincing female character. At the same time this kind of thinking gets to her in rather uncomfortable ways in real life circumstances. If this isn't the world we want to live in "in real life," then we should strongly consider if it is the world we want to recreate in an online virtual environment.

## 6 CONCLUSION

Let me start this conclusion by underscoring that I feel rather uneasy about the fact that this article might be polarizing the experiences and roles of men and women on the Net and thereby reinforcing, or re-inscribing, or maybe even creating exactly those differences that I seek to explain. Social research however might be likened to quantum physics. One can not "measure" the object/subject of research without irreversibly altering it. The result is Schrödinger's cat, either dead or alive, but never the possibility of Schrödingers cat. Both the social scientist and the quantum physicist turn out to be working hermeneutically, reading the result of their research as an indication of what might have been.

The preceding then, has shown that the anthropological method offers a fruitful approach for research into the social/cultural aspects of online life and in particular for the study of small communities. I have focused on gender in order to give the reader a glimpse of the intricacies of constructing and convincingly performing a character in a role-playing MOO. The performance of gender not only shows how the mundane everyday categories "female" and "male" are re-enacted, re-instituted in virtual reality, but also how these categories reach across the divide between the virtual and the real and influence eachother. Then, lastly, considering the implications and repercussions of how gender is/must be performed in virtual reality, I hope to have shown that in order to design and implement virtual realities that will accommodate both female and male players equally well, a thorough investigation of the implicit and explicit encoding of social reality into the very technology that enable those virtual realities is necessary.

## ACKNOWLEDGEMENTS

My thanks goes to all the citizens, past and present, of *Cybersphere,* who gave me so much (hell) over the past few years. A big thanks goes to Richard, who taught me the ropes of MU*ing on *Aquatia* and who let me teach myself even more by donning me with a wizbit. Mariska Stevens taught me more than I ever fathomed I could possibly know and supported me, always. And all the others, who *will* feature in the acknowledgements of my upcoming thesis… you know who you are.

## REFERENCES

[1] Butler, J "Performative Acts and Gender Constitution: An Essay in Phenomenology and Feminist Theory." In: Case, Sue-Ellen (ed.) *Performing Feminisms. Feminist Critical Theory and Theatre.* Baltimore, Maryland: John Hopkins University, 1990.

[2] Chagnon, N. A. *Yanomamo. The Fierce People.* New York: Holt, Rinehart and Winston, 1983 [1968].

[3] Curtis, Pavel *Mudding: Social Phenomena in Text-Based Virtual Realities.* Available: http://ftp.game.org/pub/mud/text/research/DIAC92.txt, n.d.

[4] Dibbell, J. *A Rape in Cyberspace or How an Evil Clown, a Haitian Trickster Spirit, Two Wizards, and a Cast of Dozens Turned a Database Into a Society.* Available: http://ftp.game.org/pub/mud/text/research/VillageVoice.txt, 1993.

[5] Dibbell, J. *My Tiny Life. Crime and Passion in a Virtual World.* New York: Henry Holt and Company, 1998.

[6] Herring, S. *Gender Differences in Computer-Mediated Communication: Bringing Familiar Baggage to the New Frontier, 1994*

[7] Jackson, A. (ed.) *Anthropology at Home.* (ASA Monographs 25) London: Tavistock Publications,

[8]

[9] 1987.

[10] Kessler, S. J. and W. McKenna *Gender. An Ethnomethodological Approach.* Chicago and London: University of Chicago Press, 1985 [1978].

[11] Malinowski, B. *Argonauts of the Western Pacific. An Account of Native Enterprise and Adventure in the Archipelagoes of Melanesian New Guinea.* London and New York: Routledge and Kegan Paul, 1987 [1922].

[12] Markham, A. N. *Life Online. Researching Real Experience in Virtual Space.* Walnut Creek: Altamira Press, 1998.

[13] Mnookin, J. L. *Bodies, Rest & Motion: Law and Identity in LambdaMOO.* Available: http://anxiety-closet.mit.edu/afs/athena.mit.edu/org/w/womens-studies/OldFiles/www/mnookin.html, n.d.

[14] Mnookin, J. L. *Virtual(ly) Law: The Emergence of Law in LambdaMOO.* Available: http://www.ascusc.org/jcmc/vol2/issue1/lambda.html, 1996

[15] Sullivan, C. and K. Bornstein *Nearly Roadkill. An Infobahn Erotic Adventure.* New York: High Risk Books, 1996.

[16] Unsworth, J. *Living Inside the (Operating) System: Community in Virtual Reality (draft).* Available: http://jefferson.village.virginia.edu/pmc/Virtual.Community.html, n.d.

# Modelling Interaction in Virtual Environments using Process Algebra

Boris van Schooten, Olaf Donk, Job Zwiers
University of Twente
P.O. Box 217, 7500 AE Enschede
{schooten|donk|zwiers}@cs.utwente.nl

## Abstract

Virtual environments (VEs) are often a complex mixture of novel and traditional user interface strategies, and also incorporate real-time dynamics and parallelism. We describe a modelling technique we are developing, which is based on the process algebra CSP. It is shown how VE systems and tasks can be modelled in CSP, and how a prototype system can be generated from the system specification by mapping a subset of CSP signals onto user interface functionality.

**Keywords:** Interaction Design, Process Algebra, Prototyping, Multimodal Interaction, Agent-Oriented Modelling, Virtual Environments, Task Analysis

## 1 Introduction

When designing Virtual Environments (VEs) it seems natural to design them as consisting of multiple 'agents', 'objects', or processes. For example, MUD and Virtual Reality (VR) systems are usually built out of a large number of concurrent (i.e. parallel and communicating) processes. In our own VE project, the Virtual Music Centre (VMC) [34], a high degree of concurrency exists also. The agents within the VE, the user navigating through it, as well as some more traditional 'desktop' user interface elements operate or may be operated concurrently. Plans also exist to enable multiple users to enter the VMC simultaneously, but no precise model yet exists of what information is shared and communicated between the users.

However, process concurrency is known to be a difficult computer science problem. Typical concurrency problems are timing problems and unaccounted-for unusual situations, and may surface as anything ranging from crashing bugs to ill behaviour that may be classified as usability problems. This is already seen with systems modelled after the desktop metaphor. In some ways, the desktop model is a form of VE, and may be considered a relatively established and mature model. Desktop systems often have concurrency bugs. For example, multiple windows accidentally access the same data concurrently, or the effects of a resize button or scrollbar is not notified properly, and the system crashes or the display becomes inconsistent. Concurrency-related usability problems also occur. For example, some windows do not allow the user to concurrently access other windows because of limitations of the software, and freeze the rest of the application. In some cases, this is not clear to the user, who may even think the application crashed.

It is attractive to enhance VEs with multimodal interaction. Here, we define a modality as a distinct channel through which information may be conveyed, either from the computer to the user (output modalities: ears, eyes) or the other way (input modalities: microphone, keyboard). In the literature, multimodality is interpreted in different ways by different authors, depending on their point of view: modalities may be classified not just according to physical distinction criteria, but also to cognitive or technological ones [5] [32] [27]. So, it is also possible to think of a line of text and a graphic as different modalities, because they have different cognitive properties. Bernsen [5] classifies continuous and isolated-word speech as different modalities. Isolated-word speech exists only because of technological constraints, and has distinct technological and cognitive properties. In the same vein, it is even conceivable to view working with different windows as multimodality. For example, a pull-down menu and a text entry field are two distinct technological concepts, and have different cognitive properties as well. Here, the concept of multimodality touches with the multi-agent concept as described above. This suggests that multimodality could naturally be modelled using concurrent processes, having a process for each modality.

## 1.1 Formal methods in HCI

As argued above, VE development seems to call for a modelling technique which enables designers to work easily and naturally with multiple concurrent processes. From a software engineering (SE) point of view, reducing the amount of concurrency may be more desirable. From a HCI point of view however, modelling systems as many parallel processes standing for interaction components may be closer to the conceptual or mental model the HCI designers have in mind, resulting in less problems when mapping the conceptual model to the implementation model [23]. It is also attractive to use a formal technique, because of its acclaimed properties [20]:

1. Conceptual clarity, aiding communication between the parties involved, in particular, the HCI and SE parties [14].
2. Enforcement of precision while designing, and stimulation of a specific way of thinking about the design. This may lead to timely consideration of issues that are otherwise overlooked.
3. Ability to manually or automatically verify usability-related properties. Various examples of such properties are found in the HCI literature. They can be separated into validity properties, such as reachability of states, deadlock-freeness, [27], complementarity, equivalence, and redundancy of modalities [10], and cognitive ones, such as task duration, cognitive load, and interface consistency [16].

Looking at the interactive systems literature, various attempts at formal specification have been made using various formal languages, ranging from those suitable for highly declarative and constraint-based descriptions (predicate logic, functional languages) to highly procedural ones (production systems, state automata, Petri nets, process algebras). See [20], [36], or [19] for examples of each. However, they remain little used in practice. This may in part be caused by factors other than their usefulness [7], but there are few clear results showing that they really work [15]. We will discuss some of the potential causes of problems with formal methods, and why formal methods may nevertheless be useful.

Sometimes it is claimed that an advantage of formal specification is its relative completeness with respect to reality [14], but exactly the opposite is the case. Aspects which may be important are necessarily omitted to make the specification tractable [20], or may be omitted unwittingly, even in simple cases [18]. Especially in (but not exclusively to) HCI, some of these aspects are even practically unspecifiable. Examples are the general context in which a system is used, and unanticipated user behaviour with a new system, evoked by unanticipated user knowledge and expectations. This means that, in general, some unspecified aspects remain that can only be examined by trial and error. In the case of HCI, this may also require repeated communication with the users. As a complement to developing by means of specifications, HCI often uses example-based techniques, such as mock-ups, scenarios, and prototypes. These are then considered by the developer(s) in detail, or shown to the end users [33]. However, example-based techniques may also lead to underemphasis of those aspects that can not be derived from a number of examples only, and should in turn be complemented with precise descriptions [3].

Next to this, the enforcement of precision is often seen as a drawback rather than an advantage. This may be caused by an inappropriate level of abstraction, but also by incompatibility of the specification with the developers' needs. It is important that the specification should be usable as the primary memory aid in various stages of the development process by various parties (display-based reasoning) [12]. Therefore, it should also conform to the nature of the design tasks, otherwise the specification may become a burden rather than an aid. In order to ensure that a method is useful as a thinking aid, we will have to look at what the different parties involved in the design of interactive systems need.

HCI specification techniques often document the user and the user's view. The user's task is usually documented using hierarchical task models, and the user's conceptual knowledge is usually documented using semantic nets supplemented with plain-text descriptions [28] [37]. The most popular formal HCI models are formal instances of such models, such as Goals Operators Methods Selection-rules (GOMS) [16] and Extended Task Action Grammar (ETAG) [14] (see [19] for an overview). Other models include flowchart models [22]. Note that, like many example-based methods, these formal models are mostly procedure-oriented rather than declarative or constraint-oriented, describing what happens step by step. Some of the more detailed models are also capable of modelling tasks that can be executed simultaneously, though usually, the tasks may not be concurrent, e.g. there is no way to describe how simultaneous tasks may depend on each other.

Other popular models of interactive systems are 'compositional' models, which are generally more suited to SE issues, as modular composition (manner of division of a system into subsystems) is an important SE principle. These models include Model-View-Controller (MVC) and Presentation-Abstraction-Control (PAC) (see [23] for a comparison, or [29] or [8] for an overview of variants). In these models, systems are composed of networks or hierarchies of concurrent agents, or 'interactors'. In PAC, separation is made within each interactor between the components internal data (A), presentation to and interaction with the user (P), and dynamics and communication to other PAC-modules (C). Furthermore, it prescribes communication dependencies to be hierarchical. The structure of MVC is less strict: there may be one or more models, standing for modules within the application's internals. There may be one or more view-controller component pairs communicating with the models. The controller may in turn be a model for more view-controller pairs. The view-controller pairs have a one-to-one relation to user interface objects. The view controls the presentation to the user, and the controller manages the user's manipulations.

The idea behind these models is closely related to the object-oriented paradigm, the goal of which is to enable maximum separation of concerns and re-use of standard components. The models by themselves are not formal, but can be formalised easily by describing the components in a formal language [29]. Usually they are used only to model the dependency structure of object-oriented programs, the components having a one-to-one relation to objects. However, even the SE benefits of prescribing such fine-grained composition, and emphasising composition rather than dynamics, are still uncertain, as they are for object-oriented methods in general [11] [26] [13] [1]. One of the problems that have been reported is the difficulty of following the control flow of an object-oriented program, even if the program has no processes runing in parallel. Explicit emphasis on dynamics, not inherent to interactor models, may greatly aid SE development.

Summarising, we can say that the aspects emphasised by the HCI and SE parties are different: one emphasises control flow, while the other emphasises data flow. These views may turn out to be hard to compare because of this [17], and it may result in different aspects to remain underspecified, formal precision notwithstanding. This may be a problem, since the views may lead to incompatible specifications as well, as is indicated in [38]. In this article, it is even claimed that usability is incompatible with re-usability. At the least, it would be useful to have the system and task models in the same formal notation, so they can be compared and verified more easily [35] [29].

### 1.2 Our approach

The emphasis of this article lies not on the possibilities of formal verification of usability and other properties, but on the possibilities of a formal technique to act as a reference point through various activities of the development process. It should combine systems design, prototyping, and task modelling.

We have chosen to base our formal technique on process algebra, in particular, Communicating Sequential Processes (CSP) [21]. CSP by itself is compositional, and can easily be combined with interactor models, as is shown in Markopoulos's work. Markopoulos combines ADC (a variant on PAC) with LOTOS (LOTOS [6] is an international language standard, and an extension of CSP) [29] [30]. In CSP, all dynamics are also described explicitly. It is close enough to an actual program to be easily executable, lending itself to prototyping approaches. Markopoulos even suggests that hierarchical task models may be expressed naturally in process algebra, though a more detailed account of this would be desirable. However, Markopoulos ran into some of the limitations of LOTOS for user interfaces: in particular, the UI prototyping capabilities of the LOTOS tools are limited.

Our own approach is to use CSP as a basis, but not limit ourselves to standard CSP toolkits, which are generally designed for hardware and software verification. Instead, we tailor our own formalism and additional tools for use with VE development. We are developing this technique by starting off with the simplest form of CSP, and exploring its possibilities and limitations by trying to model the VMC in some detail, so as to obtain a larger than toy-size example. The VMC is considered to be a sufficiently rich example, also containing a natural language dialogue agent and multimodality. In comparison with interactor models, the compositional granularity of our technique is slightly coarser: an interactor (as comparable to a PAC agent, an M component or a VC pair) corresponds to one CSP process. Unlike interactor models, this technique does not prescribe specific compositional dependencies, enabling experimentation with the technique and with interaction strategies.

## 2 CSP AS A MODELLING LANGUAGE

Specification in general has a number of relevant aspects that we will discuss here.

- A specification should be *modular*. This means that a specification of a large system should be composed out of smaller specifications. The smaller parts should be relatively independent, and should be meaningful specifications in their own right. Note that the task hierarchies and the software module compositions discussed in section 1 are both modular, though they are meaningful in different ways.

- In section 1 we have identified the need for a close mapping of a specification to a concrete and executable system for the design of interactive systems. However, a specification should also allow for *property oriented* (or *declarative*) specification styles. In many cases, the specification process starts out with a rather loose set of requirements that all should be satisfied. A property oriented specification style matches this situation, because it allows one to formalize each of these requirements in turn, and then to conjoin them.

Process algebras are related to *state-based* specifications. These have been used for a very long time and go back to finite state automata. Especially when limited to finite state specifications, there is an impressive tool support for automatically verifying system properties, for instance by tools like SPIN, Verilog, etc. Not all system requirements are specified easily or naturally in terms of states, however, and so it is important that state-based specifications can be combined with, for instance, behavioral specifications.

The theoretical basis of process algebras, like CSP, can be found in the theory of *labeled transition systems* (LTS). A LTS may in turn be described in terms of a set of logical constraints. Basically, a LTS is a state-based system. It is a graph with a set of nodes $Q$ that are called *states*, and with labelled edges which represent a *transition relation* $t$. We start off with a small example, taken from our VMC specification. Here, a user can walk around within the confinements of a building and interact with two other agents. A natural requirement for interaction is physical proximity of the user to the agent. We might have additional requirements, for example, the requirement that the user is looking at the agent.

This situation, although fairly simple, brings forward some typical issues already. We wish to work with a concept like "proximity requirements for communication" in an abstract way, without going into great detail how this is implemented, because:

- The precise definition of proximity might change over time, for example by incorporating a notion of viewing direction.

- The precise implementation might involve too much detail to specify practically.

- For the current version of the VMC, all that the agents react to is whether communication between the user and some agent is proximity-enabled or not.

The solution here is *not* to specify all minute detail, but rather to abstract from it: we specify only what is relevant for modelling the dynamics of most of the processes and agents in the VMC. In our example, the proximity status of the user may be modelled by the following three-state LTS (see figure 1):

1. In state `UserPos1`, the user is not in the neighbourhood of any agent,

2. In state `UserPos2`, the user is in the neighbourhood of the Karin agent,

3. In state `UserPos3`, the user is in the neighbourhood of an information board (abbreviated with "ib").

This rather minimal notion of position is all that is necessary to model the user's proximity to the agents. Note that, despite the simplicity, we have already specified something relevant: *the user cannot be both in the neighbourhood of Karin and the information board at the same time.* This property may be used for instance to prevent ambiguity when the user poses a question to an agent.

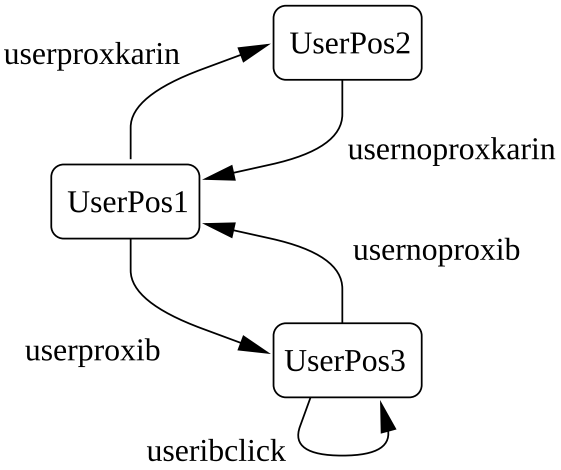



Figure 1: LTS specifying proximity relation

The state diagram also shows all possible transitions. The transitions are labeled by event names: each transition corresponds to an occurrence of the event given by its label. For instance, there is a transition from `UserPos1` to `UserPos2`, signifying an event `userproxkarin` where the user moves into the neighbourhood of the Karin agent. Note that the diagram also shows a transition labeled `useribclick`, which does not result in any state change. However, the fact that `useribclick` is only possible in `UserPos3` signifies that this event is only enabled in this state. So, a LTS can be viewed as a constraint specification, constraining which events are possible in what order.

A typical CSP system consists of a conjoinment of many such LTSes. In terms of logic, this conjoinment is a logical conjunction of a set of requirements, i.e. all these requirements should be satisfied. In terms of CSP, this is called the *parallel composition* of processes. This is in analogy with parallel execution on parallel machines, but this does not mean that the CSP process specifications have to be mapped to a parallel implementation, as long as the implementation satisfies the given constraints.

The LTS representation given here is graphical. Graphical representation is attractive, but has some disadvantages:

- For large numbers of states, a diagram showing all states does not provide much information,
- Editing and automatic processing of textual specifications is easier.

Therefore, we also use a textual representation of labeled transition systems, in the form of CSP processes. The diagram above is represented thus:

```
UserPos1 =     (userproxkarin -> UserPos2)
            [] (userproxib -> UserPos3),

UserPos2 = (usernoproxkarin -> UserPos1),

UserPos3 =     (usernoproxib -> UserPos1)
            [] (useribclick -> UserPos3)
```

In this simple specification, there is a direct correspondence between the diagram and the CSP text: each state corresponds to a so-called process definition of the form $X = Process$, where $X$ is some process name, like `UserPos1`. $Process$ can have the form $a \to P$, where $a$ is some transition label, and where $P$ is itself an expression defining a process. This denotes the process that first does action $a$ and then behaves like $P$. This construct is called the *action prefixing*. $Process$ may also have the form of a *choice* construct $P_1 \mathbin{[]} P_2$. For instance, `(userproxkarin -> UserPos2) [] (userproxib -> UserPos3)`, denotes a process that (initially) has the choice between a `userproxkarin` and a `userproxib` action. As soon as one of these two actions occurs, the choice has been made and the process behaves like either `UserPos2` or `UserPos3`.

In the example above, the action prefixing construct is only used in the form $a \to X$, where $X$ is some process name. In this case, the process definitions have a one-to-one correspondence to states in the LTS. However, any expression can follow the action. For instance, we could replace the fragment `(userproxkarin -> UserPos2)` by `(userproxkarin -> usernoproxkarin -> UserPos1)`. In this case, the LTS has more states than process definitions.

The *parallel composition* of processes $P_1$ and $P_2$ is denoted by $P_1 \parallel P_2$. For example let us consider (a sketch of) the Karin process:

```
Karin =
    (userproxkarin -> openwindows ->
                     initkarin -> Karin)
 [] (usernoproxkarin -> closewindows ->
                     exitkarin -> Karin)
```

This specification *in isolation* specifies two distinct sequences of actions, respectively following a `userproxkarin` or a `usernoproxkarin` action. Now, let us combine this with the `UserPos1` process into a system:

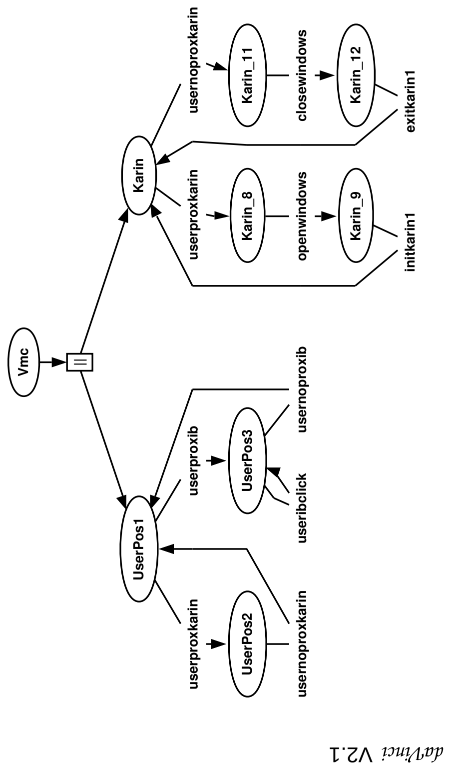



Figure 2: CSP diagram of Karin and user

```
Vmc where

Vmc  =  UserPos1  ||  Karin
```

Parallel specifications may also be represented in a graphical format, see figure 2. The figure was automatically generated from the specification with help of the daVinci graph layout program. The parallel composition is represented by a special `||` node, while the two subdiagrams representing `UserPos1` and `Karin` are regular LTS diagrams.

The resulting system is constrained by the constraints of both processes. Each process only places constraints on the actions that it actually uses. So, the processes only influence each other through those action labels that are shared by these two processes. Here, the shared action labels are `userproxkarin` and `usernoproxkarin`. In terms of CSP, shared action labels may also be called *communication channels*, and occurring actions may be called *signals*. Shared actions may only occur when the two processes execute it *jointly*. For instance, if one of the two processes is in a state where an `userproxkarin`

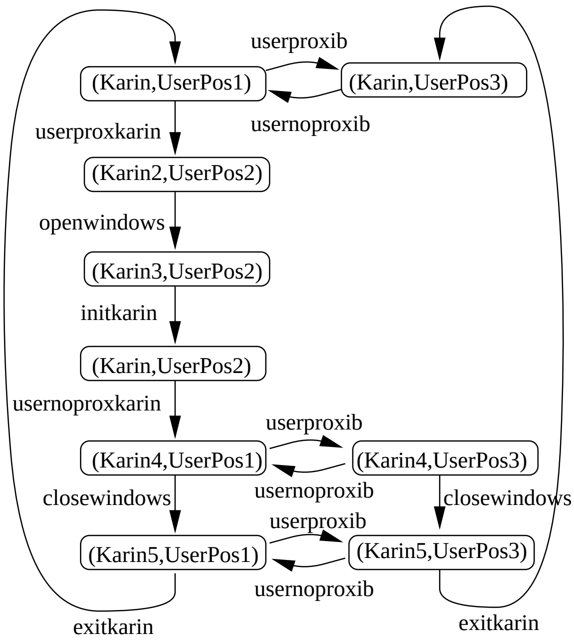



Figure 3: LTS of Karin and user

action is not enabled, then the action cannot occur at that moment. For instance, after a `userproxkarin` action, the `UserPos1` process, which has turned into a `UserPos2` process, is able to do a `usernoproxkarin` action. However, since the `Karin` process does *not* enable it, the `UserPos` process must *wait* at this moment. The `Karin` process is now able to do an `openwindows` action, followed by an `initkarin` action. These two action can happen *independently*. That is, the `Karin` process will execute these actions without synchronization with the `UserPos` processes. In all likelyhood though, there will be other processes in the system as a whole, that at this point *will* participate in the `openwindows` and `initkarin` actions. After these two actions have occurred, the `Karin` process is back at the initial state where a choice can be made between the `userproxkarin` and the `usernoproxkarin` actions. Here, the `UserPos` processes enforce that the `usernoproxkarin` action will take place, since the other action is disabled by the `UserPos` processes. The possible actions of the two systems combined may be modelled by a single big LTS, see figure 3. Note that, when there are many processes, such LTSes tend to become too large to understand, though they are generally small enough for exhaustive automatic verification.

There are two basic process terms in CSP that finish the execution of a process: **skip** and **stop**. The difference lies in what happens *after* the process has finished. Termination by means of a **stop** process denotes that all activity of the process ceases. Termination by means of **skip** denotes that a process that sequentially follows the current stage is started. This is denoted by *sequential composition* of the form $P_1$ ; $P2$, where $P_2$ is started as soon as $P_1$ terminates by executing **skip**. In practice, a sequence of actions like $(a_1 \to a_2 \to \cdots \to a_n \to \textbf{skip})$ is abbreviated as: $(a_1 \to a_2 \to \cdots \to a_n)$.

We introduce CSP operations for *renaming* and for *hiding* actions from a process. As an example, consider the following simple process that records whether a certain door is open or closed:

```
Door = (open -> close -> Door)
```

Compare this with a very similar process which specifies that the information board could be switched on or off:

```
OperateIB = (on -> off -> OperateIB)
```

In fact, basic patterns like these occur many times. An easy way of specifying them is to first specify the underlying *generic* pattern, thus:

```
TwoStates = (a -> b -> TwoStates)
```

Then, one uses the renaming operator of the form $P[d/c]$, denoting that action $c$ of process $P$ is renamed into $d$. In the example cases, we would define:

```
Door = TwoStates[open/a,close/b]
OperateIB = TwoStates[on/a,off/b]
```

Actually, many software components that one finds in for instance Java libararies for user interfaces can be specified by this sort of simple finite state processes. A TwoState process as above for instance corresponds directly to a (two-state) knob on a user interface.

Finally we discuss the CSP *hiding* operation of the form $P\backslash a$. The semantics of the operation is that all $a$ actions inside process $P$ are made invisible from outside. Hiding can be used to hide low-level details of a system that are deemed to be of less importance. In general, hiding can be used to create various *views* on a system.

For instance, actions like `userproxkarin` can be considered low-level actions from the viewpoint of the user interface, which we don't want to see. However, we *do* want to see the effect of these actions, such as the fact that Karin's windows and the information board's windows cannot be open at the same time. This may be achieved by a CSP term of the form $VMC\backslash userproxkarin$. A second example of a view is a focus on actions related to the information board only, which is achieved by hiding all actions not related to the board.

We summarize the CSP language constructs in the following table, which provides the grammar of the language:

*Processes*
$P ::= \textbf{stop} \;\mid\; \textbf{skip} \;\mid\; a \to P \;\mid\; X \;\mid$
$P_1 \;[]\; P_2 \;\mid\; P_1 \parallel P_2 \;\mid\; P_1 ; P_2 \;\mid\; P[b/a] \;\mid$
$P\backslash a \;\mid\; P \textbf{ where } X_1 = P_1, \ldots, X_n = P_n$

Sequential composition, parallel composition, and choice are all associative operations. This means that we may specify, for instance, $P_1 \parallel P_1 \parallel \cdots \parallel P_n$, without semantic ambiguity. Finally, we assume that parallel composition is an operator with lower priority than choice, which in turn has lower priority than sequential composition.

Most versions of CSP are more extensive than this. Extensions usually include value passing (data values may be passed from one process to another through a channel) and guarded actions (a guard is a Boolean function of data values, which, if false, prevents a transition to occur). Actually, these are just shorthand notations for plain CSP definitions. Note that data passing is generally not modelled when the data does not influence the dynamics of the system: the processes 'go through the motions' of passing data, but no data is passed.

## 2.1 Execution and prototyping

The current executable version provides hand execution, where the system can be stepped through by hand. By adding some directives to the definition of each CSP process, specification of a mapping between CSP signals and user interface calls or events may also be defined. This setup is similar to the scenario proposed by [2].

The coupling between CSP and the interface is as follows (see figure 4): a subset of the CSP processes correspond to user interface (UI) components. Within these processes, a subset of channels correspond to UI functions and events. The user can be seen as an observing CSP process, having as its alphabet the union set of all UI chan-

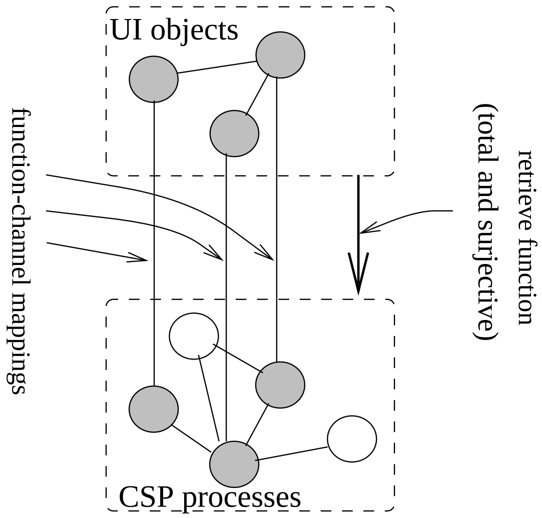



Figure 4: Architecture

nels. A number of standard UI component types are available, each having a set of standard functions and user-generated events.

Direct dependencies between interface components may also exist. For example, in the CSP descriptions that follow, windows are assumed to be independent, while in reality, they may obscure each other. If a relatively complete formal coverage is desired, we should show that these dependencies do not lead to undesirable situations. This can be done by describing a retrieve function $: UI \rightarrow CSP$ [39], which defines which concrete (UI) state maps to which abstract (CSP) state. This function should be total (i.e. all UI states have a defined CSP state), and preferably surjective (each state that is possible in CSP should also be possible in the UI). The surjectiveness is necessary to guarantee that reachability properties remain valid: otherwise, some states exist that are reachable within the CSP specification, but could never be reached through the UI.

In our example, we could assert totalness and surjectiveness by proving that any window may always be moved in such a way that all of its contents are visible and that moving a window can be done without generating unwanted signals, or that windows are in fixed positions while not obscuring each other. Note that this kind of specification is at a rather high level of detail. The examples of interactor models usually found simply assume that windows are reachable.

The mapping between channels and UI functionality may be defined by means of special directives, which are added to the declaration of the processes that correspond to UI components. The current notation is somewhat ad-hoc, but it illustrates the general principle well. A list of directives, placed between curly braces, may be declared before the body of each process declaration. Among the directives available are the following:

| | | |
|---|---|---|
| type | [<func>] | component type |
| input | {<chan>} | channels from system to user |
| output | {<chan>} | channels from user to system |

Within the <chan> directives, a list of channel names may be defined, each of which is followed by a body in which more directives may be placed. These include the following:

| | | |
|---|---|---|
| receive | [<func>] | function to call when signal occurs |
| init | [<func>] | function that sets up an interface listener which generates the signal |

Within the <func> directives, a UI function with optional parameters may be supplied.

The prototyping system is implemented by means of two communicating Unix processes: a C program implementing the CSP engine, and a Tcl process managing the user interface. The engine communicates the set of possible transitions that could occur given the current state (this set is called the nextset), then the Tcl process communicates its choice back to the engine, etc. The execution model is as follows: from all channels in the nextset that do not correspond to user output channels, one is randomly chosen. This amounts to 'flattening' parallel execution into random sequential execution, which is an established model [4]. If the channel corresponds to UI feedback, the UI is updated accordingly. If only user output channels remain, the user may generate signals by manipulating the interface.

## 3 Example models

In this section, we discuss some of the potential possibilities and problems of CSP as a modelling language by means of examples. The examples are centred around the VMC, which is a VE modelled after the music centre building in Enschede. The VMC project [34] is meant as a testbed for research in experimental interaction, such as natural language, virtual reality, and multimodal interaction. The project aims to be Web-based, i.e. to be accessible through standard Web-browsing tools. One of the agents in the VMC is a natural language dialogue system (originally called Schisma), which can be queried for information about performances, and which can handle reservations.

The examples given here include parts of the global system model, a technical design detail, and a task model.

## 3.1 System model

In the model of the VMC specified here, we have tried to include some of the most interesting agents found in the real VMC. This includes an information board, which displays information about today's performances. An attempt is also made to specify the Schisma dialogue agent, including its dialogue manager. The specification leaves out the lowest level of detail: not specified are mouse movement, the details of text input and parsing, the navigation interface, and the details of appearance of VE objects, window dressing, fonts, etc. The specification follows the actual implementation reasonably faithfully, though modelling implementation details is not considered an objective of the specification. The system consists of a number of agents, each of which has a presence in the world. The Schisma system (which is called Karin in the VE) is subdivided further into a dialogue engine, some auxiliary user interface objects, and some communication buffers. See figure 5 for a system overview.

The processes at the top level are agents within the VE. The `User` process also stands for the user's view and influence within the environment: the user can move around, generating proximity notification signals, to which the agents may react. In case more users are added, they would appear in the specification as `User` objects. Within the `User` process, the precise constraints on the user's influence are defined.

```
User {...} = UserPosN,

UserPosN =
      (userproxkarin      -> UserPosK)
   [] (userproxib         -> UserPosI)
   [] (userproxdoorfront -> UserPosDF)
   [] (userproxdoorinner -> UserPosDI),

UserPosK = (usernoproxkarin -> UserPosN),

UserPosI = (usernoproxib     -> UserPosN),

UserPosDF =
```

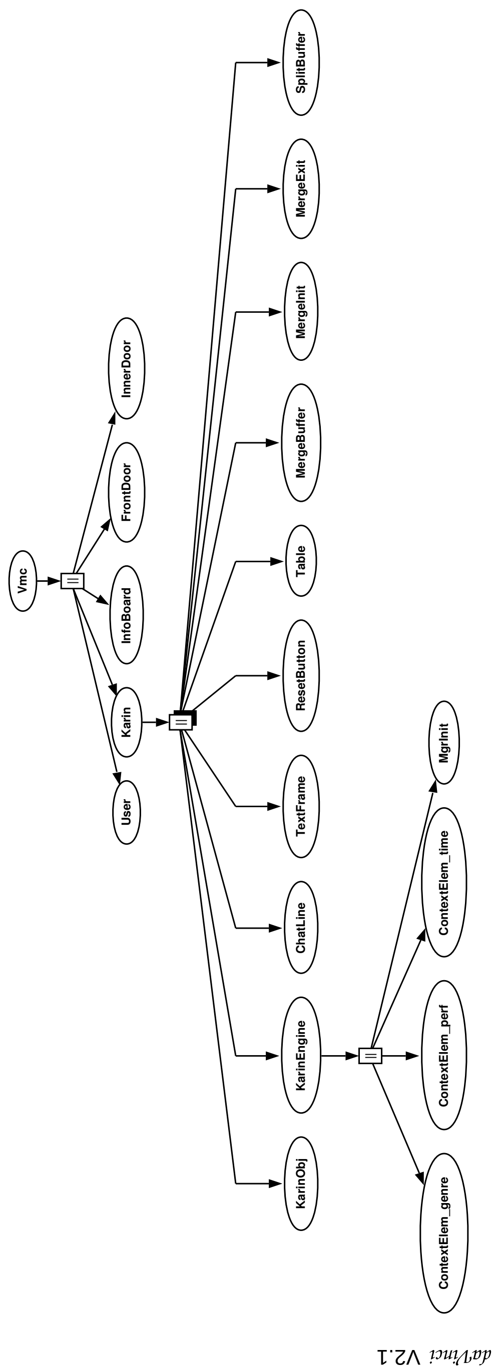



Figure 5: Overview of processes in the VMC Graphical representation of the VMC process declarations. The leaves of the tree contain the state automata (not shown). Note that the communication dependencies are not constrained by the tree structure, but may be arbitrary.

```
  (usernoproxdoorfront -> UserPosN),

UserPosDI =
  (usernoproxdoorinner -> UserPosN)
```

From the system's point of view, no more than a few discrete positions are necessary to model the user's location: agents may be notified of the user's entering or leaving their proximity, modelled as `userprox...` and `usernoprox...` signals. The user starts in front of the building, which corresponds to `UserPosN`. In this specific case, the agents are all at a sufficiently large distance from each other, so that the user can never be near two objects at the same time. `User` has only five positions, one for each agent s/he can be near, and one in which no agent is near.

Constraints such as these have to be specified if we require surjectiveness of the retrieve function: by specifying precisely where all agents may go, we can be sure the specification corresponds precisely to all possible situations. The question remains how such constraints could be specified in more general cases, for example in VEs in which the agents are able to move or a more involved notification scheme is used.

The Karin agent consists of Karin's manifestation in the world (`KarinObj`), Karin's dialogue manager (`KarinEngine`), a number of buffers for communicating with other processes, and a couple of windows used for communicating with the user. The user can also reinitialise Karin using a reset button, and Karin can display query results in a table. The four buffers are needed for multiway communication: there are several processes that communicate with `KarinEngine`, but do not communicate with each other. If these would all try to talk with `KarinEngine` over the same channels, *all* processes would be forced to synchronise their communication. The buffer implementation allows each process to write data independently, and even overwrite the data of another before `KarinEngine` could have read it: some data may be lost due to concurrent access. Note that the same problem was found in the real VMC while specifying, and is made explicit by the specification. If this data loss is not considered acceptable, the buffers could be made to block multiple consecutive writes, and deadlock conditions resulting from blocked buffers could be detected and eliminated.

The process `KarinObj` defines the presence of Karin and Karin's reaction to the proximity of the user.

```
KarinObj {
  type [object(350,150,"karin.gif")]
  output{
    userproxkarin   {init[setprox()]   }
    usernoproxkarin {init[setnoprox()] }
  }
}
 =    (userproxkarin -> openwindows
              -> initkarin1 -> KarinObj)
   [] (usernoproxkarin -> closewindows
              -> exitkarin1 -> KarinObj)
```

The UI directives define that `KarinObj` is a VE object, and set up interface listeners that generate the proximity signals `userproxkarin` and `usernoproxkarin` when the user approaches or leaves. The process reacts to these signals by respectively setting up or closing down the dialogue manager and the windows. Note the '1' in `initkarin1` and `exitkarin1`: this means that the process wants to signal `initkarin` and `exitkarin` to the dialogue manager, but the signal is buffered (in this case, through `MergeInit` and `MergeExit`). Other processes should signal their init and exit signals through other channels, named `...karin2` etc.

The processes `ChatLine`, `TextFrame` and `Table` are windows, which have two main states, `...Open` and `...Closed`. Only in the `...Open` state may the window display information and accept user input. As an example the `Table`-process is given:

```
Table {
  type [window]
  input{
    usertableopen  { receive [open()]  }
    usertableclose { receive [close()] }
    usertablerefresh {
      receive [showtext("Table filled")]
    }
  }
  output{ userclick {
      init [createbutton("click")]
  } }
} = TableClosed,

TableClosed =
     (opentable -> usertableopen ->
                              TableOpen)
  [] (closetable -> TableClosed),

TableOpen =
     (opentable -> usertablerefresh ->
                              TableOpen)
```

```
  [] (closetable -> usertableclose ->
                            TableClosed)
  [] (userclick -> textout2_perftime ->
                              TableOpen)
```

The table lists performances, which can be clicked on to select a performance. The information that should be displayed in the table is always supplied through the `opentable` channel. A user click results in a buffered signal `textout2_perftime` to the dialogue engine, communicating the information that specifies that performance.

The bulk of Karin's behaviour is specified in `KarinEngine`. The engine consists of the dialogue manager, starting at `MgrInit`, and some variables for keeping track of dialogue context. The dialogue manager is a simplified model, though it contains the main principles: keeping track of dialogue context by means of data obtained from previous utterances, and the manager itself is essentially state-based.

```
KarinEngine =    MgrInit
              || ContextElem_genre
              || ContextElem_perf
              || ContextElem_time,

MgrInit =  (initkarin -> textin_hello ->
                              MgrClear),

MgrClear = (delperf -> deltime ->
                 delgenre ->  MgrWait),

MgrExit =
   (exitkarin -> closetable -> MgrInit),

MgrWait =
   (textout_genre -> addgenre ->
              delperf -> MgrAnswTable)
 [](textout_perf -> addperf -> MgrPerf)
 [](textout_time -> addtime -> MgrTime)
 [](textout_perftime -> addperf ->
            addtime -> MgrAnswInfo)
 [](textout_book -> closetable ->
                              MgrBook)
 [](textout_yes -> textin_error ->
                              MgrWait)
 [](textout_no -> textin_error ->
                              MgrWait)
 [] MgrExit,

MgrPerf =   (gottime -> MgrAnswInfo)
         [] (nogottime -> MgrAnswTable),

MgrTime =   (gotperf -> MgrAnswInfo)
         [] (nogotperf -> MgrAnswTable),

MgrAnswInfo =
           (textin_tellinfo -> MgrWait),

MgrAnswTable = (textin_telltable ->
                   opentable -> MgrWait),
```

After some initiating actions, the dialogue manager arrives at `MgrWait`. This is the state in which Karin accepts input from the user, and reacts to it by providing information. There are two kinds of information: detailed information about one specific performance, or a list of performances. In this simplified model, it is assumed that a performance is fully specified by supplying both a performance name (`textout_perf`) and a time (`textout_time`). A list of performances can also be obtained by supplying a genre (`textout_genre`). The rest of the states are for determining what to answer. In case a query results in a list, the table is opened to display it. When the user wishes to book for a performance (`textout_book`), the manager goes into a separate booking mode `MgrBook` (not shown), which is analogous to `MgrWait`. The manager returns back to `MgrWait` when a performance has been specified and confirmed, or the booking is cancelled, or the user makes an irrelevant utterance.

Specification of the dialogue manager in simple CSP is somewhat verbose, but it is feasible to describe the complete dialogue manager this way, preferably aided by some shorthand notations for managing the context variables. This dialogue manager is finite-state (a finite-state automaton, having a set of pre-programmed states and transitions) rather than plan-based (taking actions on the basis of a hierarchy of goals and plans, created by a 'planning' engine) [24]. However, it may also be possible to describe plan-based dialogue managers in CSP. This could be done by modelling plans as processes, and defining each plan as a parallel composition of sub-plans, which is analogous to the task model found in section 3.3. Plans may also be defined recursively, just like tasks in a CSP task model [29].

In our opinion, this kind of specification has been useful for describing the system up to an interesting level of detail, and for uncovering concurrency issues: apart from the buffering problem, some other small bugs in the real VMC were found while specifying. For example, the user could select from an empty table, resulting in a nonsense reaction, and the reset button has a strange effect when clicked while Karin is still processing.

A VMC "prototype" can also be generated easily, with help of the (still limited) UI component library. See figure 6 and figure 7 for a comparison of the prototype with the real system. The navigation screen shows an overhead 2D map, rather than first-person 3D. The user, indicated by the label "userobj", can be moved around. If the user comes close to an agent, a separate window is opened, which shows a detail view of the agent. Instead of querying Karin by means of text input, the user can select his/her query by selecting from a column of buttons, each standing for one type of information that can be entered as text in the real VMC. Even though it is still very simple, the prototype could for example be used to examine visibility and layout issues, and get an idea of the system's "feel".

## 3.2 Detailed description of multimodal output

This section illustrates how more detailed and technical design issues could be discussed using the language. In the new version of the VMC, Karin's output is not limited to text. Speech and mouth movement are added, which have to be synchronised. To do this, the text is first converted to phonemes and speech. The speech is played as an audio sample, and each successive phoneme is converted to mouth animation while the sample is playing. The system is shown in figure 8.

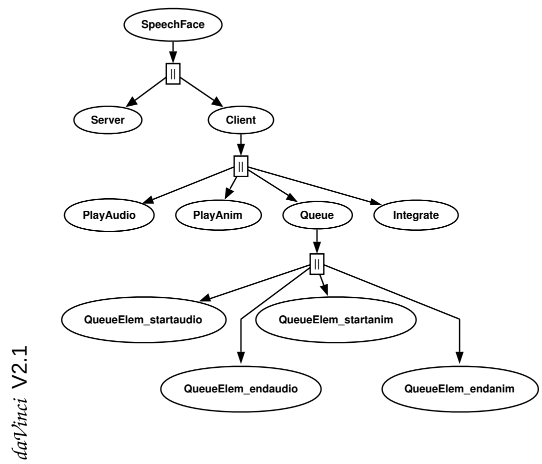



Figure 8: Speech-face system

The system is internet-browser-based, which means there are some serious technical constraints, as will become clear below. The text-to-speech program is running on a separate machine:

```
Server =     (sentence -> texttospeech
              -> Server)
          [] (loadaudio -> Server)
          [] (loadphonemes -> Server)
```

First, a sentence is sent to the server (`sentence`), which is then converted to speech and phonemes (`texttospeech`), which can then be downloaded by the client separately (`loadaudio` and `loadphonemes`).

The standard animation player component can be described as follows:

```
PlayAnim = (playanim
  -> startanim -> queuestartanim
  -> endanim -> queueendanim
  -> PlayAnim)
```

When giving the command to start playing (`playanim`), the animation starts (`startanim`, an internal event). This is then notified to a separate `Queue` process through a signal `queuestartanim`. `Queue` then enables a signal `notifystartanim` to be read once by other processes. The end of the animation (`endanim`) is similarly notified through `queueendanim`.

The audio player component might have had exactly the same structure as `PlayAnim`. However, a bug was discovered which causes the player to behave badly when a new sample is loaded while the old one is still playing. In other words, the audio player somehow participates in loading samples, and any attempt to load new samples while it is still playing results in undesirable behaviour (in the following specification, deadlock).

```
PlayAudio = (loadaudio
  -> (    PlayAudio
       [] (playaudio
           -> startaudio
           -> queuestartaudio
           -> endaudio
           -> queueendaudio
           -> PlayAudio)
     ) )
```

The technique currently used to synchronise these processes is as follows:

```
Integrate = (sentence -> loadphonemes
```


Figure 6: The real VMC

This screengrab shows the user talking to Karin. The user has just entered a date in the text entry field (the white bar on the right), and Karin replies (top right) with help of a table (bottom).

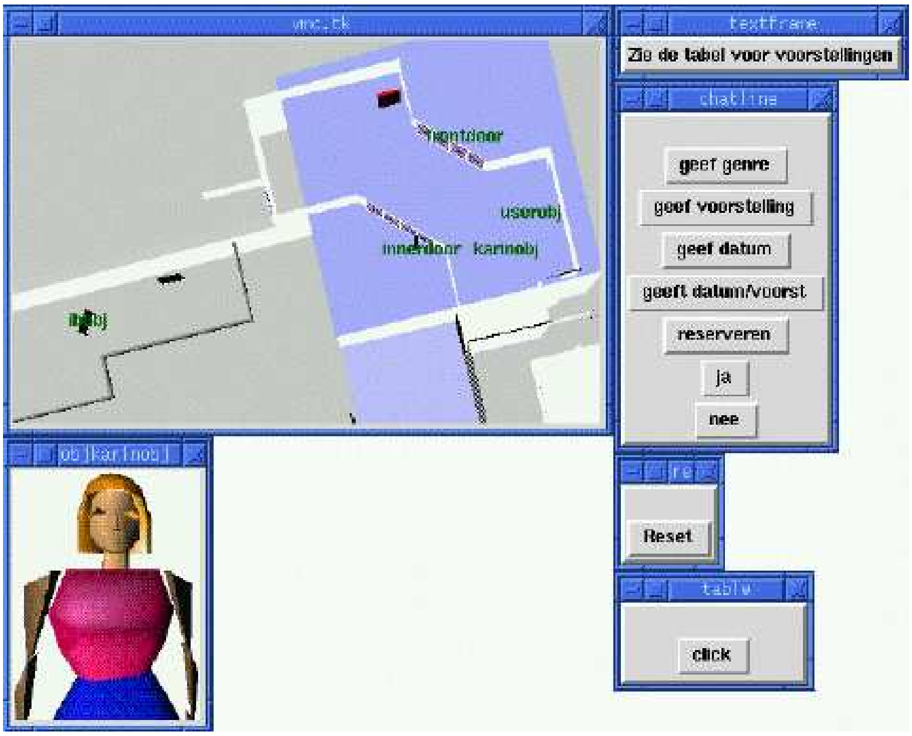



Figure 7: Generated VMC prototype

This screengrab shows the same situation as figure 6. At the top left is the navigation window, top right is Karin's reply, middle right is the text entry window, bottom left is Karin's detail view.

```
  -> loadaudio -> playaudio
                             -> IntVisWait),

IntVisWait = (notifystartaudio
                             -> IntVisPlay),

IntVisPlay =
     (playanim -> notifyendanim
                             -> IntVisPlay)
  [] (endvisemelist -> IntVisTerm),

IntVisTerm = (notifyendaudio
                              -> Integrate)
```

However, the time between `playanim` and `notifyendanim` turned out to be longer than the specified viseme duration, resulting in a cumulative buildup of delay. The first proposed solution was to split the samples into smaller parts. However, the delay between `sentence` and `playaudio` (the downloading turns out to be slow) causes unacceptable delay between the samples. It is not possible to load the next sample while the current one is playing either, because of the bug in `PlayAudio`. A limited solution would be to react on `notifyendaudio` immediately to stop the mouth early. The solution currently implemented is to measure time immediately after `notifystartaudio`, and to calibrate the duration of successive visemes according to the time that really elapsed as opposed to the viseme duration specified.

## 3.3 Task analysis

CSP can be used for specifying task hierarchies similar to basic forms of HTA reasonably naturally: each task and subtask may be specified as a process, and each basic operation may be specified as a channel. Loops and interleaving tasks could also be described easily. The task model can be tested with the system by parallel composition with the system. CSP also has some interesting possibilities for modelling more detailed cognitive aspects, such as task performance and memory load, as in task modelling languages like GOMS [16] and ETAG [14]. We list some of the possibilities here.

- In GOMS and ETAG, short-term memory is modelled as a goal stack, containing a list of all goals and subgoals currently being considered by the user. These may also contain parameters needed to perform the task. In CSP, this may be modelled by the processes currently running. In extensions of CSP with value passing, the parameters may be modelled by the values bound by each process.
- In GOMS and ETAG, choice between alternative possible strategies is done by means of selection rules. These rules specify whether which action should be taken by means of a Boolean function of the user's memory, or, possibly, of the system's feedback. Choice in basic CSP is simply by the choice operator, which could model both types of choice. CSP extensions with value passing generally allow choice to be made using an arbitrary Boolean expression as well.
- ETAG also specifies a domain model, describing the objects in the task domain, and their relations. In a CSP model, this roughly corresponds to the relation between channels and processes, and between different processes in the system model. However, no arbitrary inter-object relations may be described.
- Extensions of GOMS, such as EPIC [25], also enable highly detailed cognitive performance prediction, using a model of concurrent cognitive subsystems or critical path analysis for tasks that can be done simultaneously. This is especially interesting for modelling the effect of multimodality on user tasks. In CSP, tasks that may be interleaved can be described quite naturally by means of parallel composition. Cognitive subsystems may also be modelled by describing them as separate CSP processes. Real-time extensions of CSP [9] also enable specification of durations, which could be used for critical path analysis.

Here, we give an example of a task model. Note that the task is only one example of the tasks that a user may be trying to achieve with the VMC system. In this task, the user wishes to go to the theatre today, and is trying to select a suitable play. It is assumed that the information needed to select the play can only be obtained from the information board or by asking Karin for details about the performance. The task hierarchy is as follows:

```
Goal = Query ; Book,

Query = QueryIB [] QueryKarin,

QueryIB = GotoIB ; GetInfoIB,
```

```
QueryKarin =
 GotoKarin ; GetPerfList ; QueryEachPerf,

QueryEachPerf =
  SpecifyPerf ; QueryEPRead ;
  (    (seemore ->    QueryEachPerf)
    [] (seenenough -> skip) ),

Book = (    (GotoKarin ; PerformBook)
         [] (PerformBook) )
       ; Confirm

PerformBook  =
     (SpecifyBook ; SpecifyPerf)
  [] (SpecifyPerf ; SpecifyBook),
```

The goal consists of two subgoals: obtain the information (Query) and book for the play (Book). At this level of specification, there are several choices available to the user: in Query, s/he can query the information board (QueryIB) or Karin (QueryKarin); in PerformBook, s/he can first specify that s/he wants to book, and then specify which play to book, or do it the other way round. When querying details in QueryEachPerf, the user repeats the subtasks until s/he has seen enough, which is determined by the channels seemore and seenenough, which are internal to the user. When trying to book, it is possible that the user is still at the information board and has to walk to Karin first. The feedback from the system can be used to determine which one is applicable. Note that two tasks occur at different places, namely SpecifyPerf and GotoKarin. The following bottom-level tasks exist for the information board:

```
GotoIB =    GotoIB2
         [] (usernoproxkarin -> GotoIB2),
GotoIB2 = (userproxib -> skip),

GetInfoIB = (useribclick
              -> useribshowinfo -> skip),
```

Note that the navigation task GotoIB is versatile: the user will arrive at the information board regardless of where s/he is, by reacting to system feedback. The query information is obtained by receiving the signal useribshowinfo. If the user were not interested in this information, s/he could simply choose to ignore the signal. The bottom-level tasks for Karin are:

```
GotoKarin  =
     GotoKarin2
  [] (usernoproxib -> GotoKarin2),
GotoKarin2 = (userproxkarin -> skip),

GetPerfList = (usertyped_time
              -> usertableopen -> skip),

SpecifyPerf =
     (userclick  -> skip)
  [] (usertyped_perftime -> skip),

QueryEPRead =
  (usershowtext_tellinfo -> skip),


SpecifyBook  = (usertyped_book -> skip),

Confirm = (usershowtext_confirm
          -> usertyped_yes
          -> usershowtext_done -> skip),
```

GotoKarin is analogous to GotoIB; the other tasks are mostly obvious. Note that, when querying Karin for specific performances (QueryEachPerf), the user can obtain specific details either by clicking on the performances table (userclick) or by keying in a specification of the play (usertyped_perftime). In case the table is not open (this happens when the user has chosen to query the information board and is now trying to book), the system feedback will simply not allow the former option.

A weakness of the current model is that the user can only observe system feedback by seeing that certain output channels are enabled, or by catching input signals as they are generated. In some cases, for example when a task model requires re-reading an answer given by Karin, channels for re-reading the display should be added to the system model. For example, for each window process, different states should be added that distinguish the different texts being displayed at any particular moment, and an extra channel should be added for each state that enables the user to read that particular text. While designing the system model, such channels were never considered, because they are not meaningful from a systems designer's point of view: the system is not concerned with what happens with information after the command is given to display it. It is not even possible to detect when or if the user reads any message displayed. These extra signals could be modelled by means of variables (see below).

### 3.4 FUTURE RESEARCH

By means of examples, we have indicated that CSP is interesting as a central specification language for various aspects of the development process. From here onwards, several strands of research may be identified:

1. Investigating the possibilities of specifying hierarchical task models and plan-based dialogue models using process algebra. Apparently there is little research on this subject, though some examples of task models are emerging in the literature [29] [31]. However, a serious large-scale or systematic coverage seems to be lacking.

2. Extending the programming environment to include a more complete coverage of user interface functionality. The possibility of invoking arbitrary software modules from CSP, such as parsers and speech generators, is also attractive. Taken to its fullest extent, this means that CSP could be used as a central 'glue' language, glueing together separate modules, while keeping the architectural dependencies explicit, making software maintenance and experimentation easier.

   This could be done by an annotation scheme similar to the 'UI directive' scheme explained here. Apart from a less ad-hoc notation, this scheme could use a redesign: restrictions to the communication scope of the channels defined in the directives should be defined in a meaningful way, and a value passing scheme should be added.

3. Developing or adopting language extensions, shorthand notations, and verification tools that are particularly useful for VE development. The most basic ones are variables and value passing, enabling a shorter notation for some parts of the specification. System feedback through persistent modalities, as explained in the example of re-reading window contents in section 3.3, could also be modelled by means of variables. For example, a variable could stand for the state of a window, and is directly accessible by the user.

   Further extensions are interesting for VE development in particular. The ability to create multiple instances of one process is interesting for modelling multi-user systems. Real-time extensions are interesting for analysing cognitive performance in multimodal systems.

4. An interesting possibility, not yet mentioned, is automatic user data analysis in terms of the CSP signals. The sequence and duration of the signals that occur during interaction could easily be logged and analysed. This could also be useful for obtaining data for cognitive performance prediction, as is done in GOMS models.

## REFERENCES


[1] Ritu Agarwal, Atish P. Sinha, and Mohan Tanniru. The role of prior experience and task characteristics in object-oriented modeling: an empirical study. *International journal of human-computer studies*, 45:639–667, 1996.

[2] Heather Alexander. *Formal methods in human-computer interaction*, chapter 9: Structuring dialogues using CSP. Cambridge university press, 1990.

[3] M. E. Atwood, B. Burns, A. Girgensohn, A. Lee, T. Turner, and B. Zimmermann. Prototyping considered dangerous. In *Proceedings of INTERACT'95: Fifth IFIP Conference on Human-Computer Interaction*, pages 179–184, 1995.

[4] J. C. M. Baeten and W. P. Weijland. *Process Algebra*, volume 18 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 1990.

[5] Niels Ole Bernsen. towards a tool for predicting speech functionality. *Free Speech Journal*, 1, 1996.

[6] T. Bolognesi and E. Brinksma. Introduction to the ISO specification language LOTOS. *Computer Networks and ISDN Systems*, 14:25–59, 1987.

[7] Jonathan Bowen and Mike Hinchey. Seven more myths of formal methods. *IEEE Software*, 12(4):34–41, 1995.

[8] Gaëlle Calvary, Joëlle Coutaz, and Laurence Nigay. From single-user architectural design to PAC*: a generic software architecture model for CSCW. In *CHI '97*, 1997.

[9] Zhou Chaochen, C. A. R. Hoare, and A. P. Ravn. A calculus of durations. *Information processing letter*, 40(5), 1991.

[10] Joëlle Coutaz, Laurence Nigay, Daniel Salber, Ann Blandford, Jon May, and Richard M. Young. Four easy pieces for assessing the usability of multimodal interaction: the CARE properties. In *Proceedings of INTERACT'95: Fifth IFIP Conference on Human-Computer Interaction*, pages 115–120, 1995.

[11] J. Daly, A. Brooks, J. Miller, M. Roper, and M. Wood. The effect of inheritance on the maintainability of object-oriented software: an empirical study. In *Proceedings of the 1995 international conference on software maintenance*, 1995.

[12] Simon P. Davies. Expertise and display-based strategies in computer programming. In *People and computers VIII: proceedings of the HCI '93 conference*, 1993.

[13] Simon P. Davies, David J. Gilmore, and Thomas R. G. Green. Factors influencing the classification of object-oriented code: Supporting program reuse and comprehension. In *Symbiosis of human and artifact, advances in human factors/ergonomics 20A*, 1995.

[14] G. de Haan and N. Muradin. A case study on applying Extended Task-Action Grammar (ETAG) to the design of a human-computer interface. *Zeitschrift für Psychologie*, 200(2):135–156, 1992.

[15] Norman Fenton, Shari Lawrence Pfleeger, and Rober L. Glass. Science and substance: a challenge to software engineers. *IEEE Software*, 11(4):86–95, July 1994.

[16] Wayne D. Gray, Bonnie E. John, Rory Stuart, Deborah Lawrence, and Michael E. Atwood. GOMS meets the phone company: Analytic modeling applied to real-world problems. In *Proceedings of IFIP INTERACT'90: Human-Computer Interaction*, Foundations: Cognitive Ergonomics, pages 29–34, 1990.

[17] T.R.G. Green. Programming languages as information structures. In *Psychology of programming (computers and people series)*, pages 118–137, 1990.

[18] C. A. Gurr. Supporting formal reasoning for safety-critical systems. *High Integrity Systems*, 1(4):385–396, 1994.

[19] G. de Haan, G. C. van der Veer, and J. C. van Vliet. Formal modelling techniques in human-computer interaction. *Acta Psychologica*, 78(1-3):27–67, 1991.

[20] M. Harrison and H. Thimbleby, editors. *Formal methods in human-computer interaction*, chapter Preface. Cambridge university press, 1990.

[21] C. A. R. Hoare. *Communicating Sequential Processes.* Prentice hall, New York, 1985.

[22] K. S. Hone and C. Baber. Using a simulation method to predict the transaction time effects of applying alternative levels of constraint to user utterances within speech interactive dialogues. In *ESCA workshop on spoken dialog systems: theories and applications*, pages 209–212, 1995.

[23] Andrew Hussey and David Carrington. Using Object-Z to compare the MVC and PAC architectures. In C. R. Roast and J. I. Siddiqi, editors, *Proceedings of the BCS-FACS Workshop on Formal Aspects of the Human Computer Interface*, 1996.

[24] Arne Jönsson. *Dialogue management for natural language interfaces.* PhD thesis, Linköping University, department of computer and information science, 1993.

[25] David E. Kieras, Scott D. Wood, and David E. Meyer. Predictive engineering models based on the EPIC architecture for a multimodal high-performance human-computer interaction task. *ACM Transactions on Computer-Human Interaction*, 4(3):230–275, 1997.

[26] D. Kung, J. Gao, P. Hsia, F. Wen, Y. Toyoshima, and C. Chen. Change impact identification in object oriented software maintenance. In *Proceedings of the 1994 international conference on software maintenance*, 1994.

[27] Ian Lewin. Formal design, verification and simulation of multi-modal dialogues. In *TWLT13: formal semantics and pragmatics of dialogue*, 1998.

[28] Kee Yong Lim and John Long. *The MUSE method for usability engineering.* Cambridge University Press, 1994.

[29] Panos Markopoulos. *A compositional model for the formal specification of user interface*

*software*. PhD thesis, University of London, 1997.

[30] Panos Markopoulos, Peter Johnson, and Jon Rowson. Formal architectural abstractions for interactive software. *International journal of human-computer studies*, 49:675–715, 1998.

[31] M. Mezzanotte and F. Paternó. Verification of properties of human-computer dialogues with an infinite number of states. In C. R. Roast and J. I. Siddiqi, editors, *Proceedings of the BCS-FACS Workshop on Formal Aspects of the Human Computer Interface*, 1996.

[32] Katashi Nagao and Akikazu Takeuchi. Speech dialogue with facial displays: Multimodal human-computer conversation. In *Proceedings of ACL-94*, 1994.

[33] Jakob Nielsen. *Usability engineering*. Academic Press, 1993.

[34] A. Nijholt, Arjan van Hessen, and J. Hulstijn. Speech and language interaction in a (virtual) cultural theatre. In *Natural language processing and industrial applications (NLP+IA 98) - Special accent on language learning*, pages 176–182, 1998.

[35] P. Palanque and R. Bastide. A design lifecycle for the formal design of interactive systems. In C. R. Roast and J. I. Siddiqi, editors, *Proceedings of the BCS-FACS Workshop on Formal Aspects of the Human Computer Interface*, 1996.

[36] P. Palanque and F. Paternò, editors. *Formal methods in Human-computer interaction*. Springer Verlag, 1998.

[37] G. C. van der Veer, B. F. Lenting, and B. A. J. Bergevoet. GTA: Groupware task analysis - modeling complexity. *Acta Psychologica*, 91:297–322, 1996.

[38] Hans Wegener. The myth of the separable dialogue: software engineering vs. user models. In *Proceedings of INTERACT'95: Fifth IFIP Conference on Human-Computer Interaction*, pages 169–172, 1995.

[39] J. Zwiers, J. Coenen, and W. P. de Roever. A note on compositional refinement. In *Proceedings of the 5th refinement workshop*, pages 342–366, 1992.

# Presence: Interacting in Virtual Reality?

Martijn J. Schuemie, Charles A.P.G. van der Mast
Delft University of Technology
Faculty Information Technology and Systems
Department Information Systems
Section Design and Realization of Information Systems
P.O. Box 356 - 2600 AJ Delft - The Netherlands
{m.j.schuemie, charles}@its.tudelft.nl

## ABSTRACT

An important concept in Virtual Reality (VR) is that of presence; the subjective perception that a mediated experience seems very much like it is not mediated. Many authors have assumed a strong relation between presence and the level of interactivity, and this article will look into this assumption. Several variables of interactivity show a clear relation to presence, but others are more complex. Knowing which elements of a Virtual Environment (VE) can attribute to presence is necessary for some applications such as treatment of phobia, necessitating further research into this area.

**Keywords:** interactivity, presence, virtual reality, human-computer interaction, phobia

## 1 INTRODUCTION

When using Virtual Reality people often experience a feeling of actually being in the computer generated environment, a feeling described as 'presence'. Presence can more explicitly be defined as a mediated experience that seems very much like it is not mediated [5]. When a user experiences a high level of presence, it is even possible for the user to develop fear in response to simulated anxiety-provoking stimuli. Experiments have for instance shown a significant positive relation between presence and fear of heights [16].

This makes it possible for VR to be used in the treatment of phobia, where patients have to be exposed to the stimuli they fear. VR has already been shown to be effective in the treatment of fear of heights, fear of flying, arachnophobia, claustrophobia and agoraphobia, the fear of being in places from which escape might be difficult or embarrassing. At the Delft University of Technology, in close cooperation with the University of Amsterdam, a generic system for the treatment of phobia is being developed, taking into account the HCI-problems unique to such a system. One of these problems is the question how presence can be achieved. When designing VR there is a strong tradeoff between elements such as image quality and update speed caused by technological limitations [24], so it is important to choose the right elements to be incorporated into the VR to increase the level of presence and make the treatment more effective.

What causes presence remains somewhat of a mystery [8], although most authors have either implicitly assumed or explicitly suggested that a major or even the primary cause of presence *is the ability to interact with a mediated environment* [5]. In this paper we will look at some of these explicit suggestions and look at the extent to which these can be validated. First, we will take a closer look at presence and the various methods for measuring the level of presence. Then interactivity will be broken down into several variables.

After reviewing relevant research and literature regarding the relation between these variables of interaction and presence will be investigated, resulting in several conclusions in the last paragraph.

## 2 PRESENCE

There exist many theories regarding the nature of presence. In this paragraph one general theory will be reviewed, as well as several methods for measuring presence.

### 2.1 GENERAL THEORY OF PRESENCE

When interacting with a VE two mental models will be activated and shaped [25]:

1. The model of the Real World (RW)

2. The model of the Virtual World (VW)

Presence refers to the sense of "being in" a world, a state where these two models begin to overlap. It therefore only makes sense to speak about the degree of presence in one environment (the virtual environment) *relative to* another (the real environment) [21]. In other words: presence refers to the distinction made by the user between the RW and the VW.

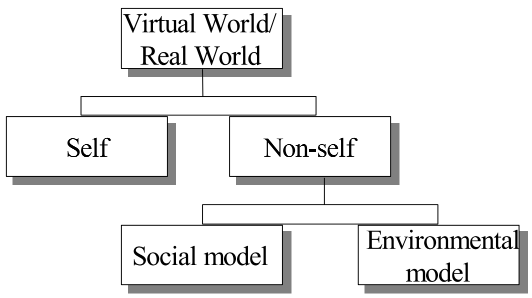



Figure 1: Mental models in a VR

Both models can be divided into the 'Self', which is a model of the individual him or herself, and the 'Non-self', a model of the environment as the individual experiences it. The non-self can even be further divided into a social model and an environment model. For each lower level model a specific type of presence can be defined [7]:

- Personal presence is related to the ‘Self’. It is a measure of the extent to which one feels like one is in a virtual world.
- Social presence relates to the social model as part of the ‘Non-self’. It is sometimes defined as the extent to which a medium is perceived as sociable, warm, sensitive, personal or intimate when it is used to interact with other people [18]. However, it is also possible that social presence is achieved using synthetic beings, for example using a creature that keeps coming back to you, asking you to pick it up and throw it away [4]
- Environmental presence refers to the environment model and indicates the extent to which the environment itself appears to know that you are there and to react to you.

## 2.2 MEASURING PRESENCE

As described in the previous paragraph, presence is very much a subjective concept. Measuring presence is therefore a difficult task, although several approaches have been suggested [20][2]:

*1. Participants reported sense of presence*

This is the most widely applied method, although very dependent on the participants' own subjective perception. Often questionnaires are used, but also attempts have been made to measure presence by letting the participant operate a continuous slider indicating the perceived level of presence at that time [12].

*2. Observation of participants behavior*

For instance, measuring reflex behavior to virtual objects directed at the participant. Also, during treatment of phobias, subjects displayed physiological responses to the simulated stimuli such as sweating, loss of balance and weak knees, suggesting a high level of presence.

*3. Performance of tasks in real and virtual environments*

This approach is based on the assumption that people perform better, i.e. more similar to the real world, in VR with higher presence. Therefore, presence can be measured by measuring the performance of the users, for instance the time taken to complete a certain task.

*4. Discrimination between real and virtual events*

When virtual and real experiences are cognitively processed in a similar fashion by the human brain, this indicates a high level of presence in the VE.
For instance, presence could be measured by determining the differences between memories of real and of virtual events, such as differences in accuracy and in detail [11].

*5. Incorporation of external stimuli*

If the participant interprets an external event, such as a loud noise, in the context of the virtual environment then he/she must be present in that virtual environment.

All these approaches are still being investigated and a universal measurement for presence is not available. Comparing results from different research projects concerning presence will therefore inherently be somewhat inaccurate.

## 3. INTERACTIVITY

We will consider an interactive system to be one in which the user can influence the form and/or content of the mediated presentation or experience [19]. The degree to which a medium can be said to be interactive depends on a number of subsidiary variables. Five primary ones will be mentioned here:

1. The number of inputs from the user that the medium accepts and to which it responds [3].
2. The number (and type) of characteristics of the mediated presentation or experience that can be modified by the user [7].
3. The range or amount of change possible in each characteristic of the mediated presentation or experience. It has even been suggested that an unrealistic amount of responsiveness can increase the level of presence [7], such as rain starting upon entering a room!
4. The degree of correspondence between the type of user input and the type of medium response. It is a "widely accepted working hypothesis" that "using our familiar sensorimotor skills to manipulate virtual objects directly by means of whole-hand input devices...contributes to our sense of presence much more than writing programs, twisting knobs, or pushing a mouse to accomplish the same task" [26].
5. The speed with which the medium responds to user inputs. When forced to choose between responsiveness to motion and resolution of images, VR developers are choosing responsiveness as the more important factor [7].

These variables coincide with three variables proposed by Steurer [23]:

1. Speed, corresponding to variable 5 mentioned above
2. Range, which covers variable 1, 2 and 3
3. Mapping, coinciding with variable 4

In this article the relation between this last set of variables and presence will be explored because a more detailed approach is not possible with the data available.

## 4 INTERACTIVITY AND PRESENCE

As stated earlier, interactivity of a medium is viewed as on of the key factors in facilitating the feeling of presence [17]. This could be explained with the use of an ecological perspective [6]: organisms such as humans perceive in their environment elements that have a meaning for the organism. This meaning can also be termed 'affordance', because it relates what the element affords. For example, the ground affords walking, a chasm affords falling and hurting, an apple might afford eating and a tiger affords being eaten. In other words, the possible interaction with the element for the organism is what is primarily perceived by the organism. Indeed, ethnographic study of interactions in multi-user VR-systems has shown perception and action to be closely linked [14]. Perception of the real world is closely linked to the action possible (i.e. interactivity) and it is hypothesized that interactivity in VR will lead to a type of perception very similar to that of the real world, causing a sense of presence.

This does however not imply that all variables of interactivity defined in the previous paragraph have the same impact on presence. In fact, research has shown that increasing such a variable sometimes causes no increase in presence. The exact mechanism determining the amount of presence experienced is unclear, although some relations are getting visible.

### 4.1 SPEED

The variable of response-speed seems to have the most straightforward effect on presence: Research showed that the performance of people completing a task in VR degraded significantly with imposed response lags of 40 ms or more[22]. It has been shown that presence reported by the user drops drastically below a level of 15-20 frames per second [1]. This would mean that a certain update speed is a minimal requirement for presence, but it should be noted that above the mentioned threshold very little increase in presence is to be expected.

### 4.2 RANGE

The possible range of interaction is also related to the level of presence. It is clear that the use of 6 degrees-of-freedom head-tracking can greatly increase the level of presence [9], although people often need to be encouraged to use this input device. When they first don the head-mounted display (HMD) they frequently treat it as they would a computer screen and just stand rigidly looking ahead. Experiments where subject were required to move their head and even their entire body to perform a task showed that body movement had a positive effect on presence [13]. The use of tactile augmentation, where the user can actually touch a virtual object and feel it, has also been reported to increase presence [10]. This technique uses a real object with a tracker attached to it so a virtual object can be simulated in its exact position.

However, increasing the number and range of inputs not always increases the level of presence. In an attempt to find a relation between level of social presence and the number of social-presence-cues an increase in these cues such as the possibilities for making gestures to each other and choosing a personal 3D representation did not result in increased presence [25]. Social presence is based on the fact that if other beings recognize you as being in the virtual world with them, and interact with you, that offers further evidence that you exist in the virtual world [7]. It seems that this intersubjectivity reduces the need for visual appearance and 'material' structure of the environment [14]. We could even make the hypothesis that, once social presence has been established in a way that the users are aware of each others existence in the VR, the fidelity of the interaction is of minor importance. This again suggests that there is a threshold above which increases in the variable, in this case the range of social interaction that is possible, has little effect on presence.

## 4.3 MAPPING

The mapping between the type of user input and the type of medium response also has an influence on the level of presence. Research has at least shown that performance in a search task was greatly increased by using head tracking instead of a hand-tracking device to change the viewpoint [15]. But other experiments revealed there can also be no difference in presence for different types of user input [1]. Subjects using a joystick were compared to those using a space mouse. Both devices offered 3-degrees of freedom, so the only difference was mapping between input and response. The fact that no difference in presence was found indicates that the exact contribution of this mapping is not yet fully understood.

## 5 CONCLUSIONS, FUTURE RESEARCH

The research mentioned in this article clearly determines interactivity of virtual environments as an important cause of presence. Some aspects of interactivity, such as the speed of the responses of the environment, show a clear contribution to presence up to a point. Further research is required however to understand the influence of mapping between input and response, as well as the range of user inputs. As suggested it may be possible that social presence can be achieved using a minimal range of interactivity, although research should investigate what these minimal requirements are.

In HCI there are many methods for designing user-interfaces. However, for VR applications such as treatment of phobia, where presence is an essential part of the interface, a new method is needed. A conclusion of this article is that when designing the interaction between human and computer the effects of the design-decisions on presence should also be taken into consideration. Often the interaction should be less efficient to increase the level of presence, for instance requiring the user to move about and interact with many (synthetic) beings, using a type of input device very natural to use but less effective than other devices.

As a last point, it should be pointed out that even better methods for measuring presence are required [25] to facilitate a precise view of the relation between interaction and presence. Much research in this area is still needed.

## REFERENCES


[1] Barfield, W., Baird, K.M., Bjorneseth, O.J., Presence in virtual environments as a function of type of input device and display update rate, *Displays*, 19, 91-98, 1998

[2] Barfield, W., Sheridan, T., Zeltzer, D., Slater, M., Presence and performance within virtual environments, *Virtual Environments and Advanced Interface Design*, Oxford University Press, 1995

[3] Biocca, F., Delaney, B., *Immersive virtual reality technology, Communications in the age of Virtual Reality*, pp. 57-127, Hillsdale, NJ: Lawrence Erlbaum Associates

[4] Delaney, B., *Virtual Reality Symposium*, Michigan: Saginaw Valley State University, April, 1992

[5] Ditton, T., Lombard, M., At the Heart of it all: The concept of presence, *Journal of Computer-Mediated Communication*, 3(2), september, 1997

[6] Gibson, J.J, *Ecological Approach to Visual Perception*, Boston : Houghton Mifflin, 1979

[7] Heeter, C., The subjective experience of presence, P*resence*,1(2), 262-271, 1992

[8] Held, R.M., Durlach, N.I., Telepresence, *Presence*, 1(1), 109-112, 1992

[9] Hendrix, C.M., *Exploratory Studies on the Sense of Presence in Virtual Environments as a Function of Visual and Auditory Display Parameters*, Master thesis, University of Washington, 1994

[10] Hoffman, H., Groen, J., Rousseau, S., Hollander, A., Winn, W., Wells, M., Furness, T., Tactile Augmentation: Enhancing presence in virtual reality with tactile feedback from real objects, *Paper presented at the meeting of the American Psychology Society*, 1996, San Francisco, CA

[11] Hullfish, K.,C. *Virtual Reality Monitoring: How Real is Virtual Reality?*, Unpublished master's thesis, University of Washington, Seattle, Washington, 1996

[12] IJsselsteijn, W.A., de Ridder, H., Measuring Temporal Variations in Presence, *paper presented at Presence in Shared Virtual Environments Workshop*, University College London, 10 - 11 June 1998

[13] Mariani, J., McCarthy, J., Slater, M., Steed, A., The Influence of Body Movement on Subjective Presence in Virtual Environments, *Human Factors*, Vol.40, no. 3, pp. 469-477, 1998

[14] O'Brien, J., Büscher, M., Rodden, T., Trevor, J., 'Red is behind you': the experience of presence in shared virtual environments, *paper presented at Presence in Shared Virtual Environments Workshop*, University College London, 10 - 11 June 1998

[15] Pausch, R., Shackleford, M.A., Proffitt, D., A user study comparing head-mounted and stationary displays, *Proceedings of the IEEE 1993 Symposium on Research Frontiers in Virtual Reality*, pp. 41-45, 1993

[16] Regenbrecht, H.T., Schubert, T.W., Friedmann, F., Measuring the Sense of Presence and its Relation to Fear or Heights in Virtual Environments, *International Journal of Human-Computer Interaction*, 10(3), 233-249, 1998

[17] Sheridan, T., Musings on Telepresence and Virtual Presence, Telepresence. Presence: *Teleoperators and virtual environments*, 1:1, 120-126, 1992

[18] Short, J., Williams, E., Christie, B., *The social psychology of telecommunications*, London, Wiley, 1976

[19] Slater, M , Davidson, A., Liberation from flatland: 3D interaction based on the desktop bat, *Eurographics '91*, pp 209-221, Elsevier Science Publishers B.V., 1991

[20] Slater, M., Usoh, M., *An experimental exploration of presence*, Department of Computer Science, Queen Mary and Westfield College, University of London, Report, 1992

[21] Slater, M., Usoh, M., Steed, A., Depth of presence in immersive virtual environments, *Presence: Teleoperators and Virtual Environments,* 1994

[22] So, R.H.Y. and Griffin, M.J., Experimental studies of the use of phase lead filters to compensate lags in head-coupled visual displays, *IEEE Transactions on Systems, Person, and Cybernetics*, Vol. 26, No. 7, July 1996

[23] Steuer, J., Defining Virtual Reality: Dimensions Determining Telepresence, *Journal of Communication,* 42(4), 73-93, 1992

[24] Stuart, R., *The Design of Virtual Environments,* McGraw-Hill, 1996

[25] Thie, S., Wijk, J., A General Theory on Presence: Experimental Evaluation of Social Virtual Presence in a Decision Making Task, *paper presented at Presence in Shared Virtual Environments Workshop*, University College London, 10 - 11 June 1998

[26] Zeltzer, D., Autonomy, interaction and presence, *Presence*, 1(1), 127-132, 1992

[27]

# Towards an Understanding 3D Virtual Reality Architectural Design System

Jarke J. van Wijk
Eindhoven University of Technology
Fact. Mathematics and Computer Science
vanwijk@win.tue.nl

Bauke de Vries
Eindhoven University of Technology
Fact. Building and Architecture
b.d.vries@bwk.tue.nl

Cornelius W.A.M. van Overveld
Eindhoven University of Technology
Fact. Mathematics and Computer Science
wsinkvo@win.tue.nl

Abstract

Current systems for Computer Aided Architectural Design are of limited use for the early phase of a design. We give a specification of a new system, in which we strive for a more natural and intuitive interaction. A key element is geometric analysis. From the input of the user, implicit geometric and structural relations are derived, which are used during subsequent interaction.

## 1 Introduction

Last 30 years many computer graphics systems for architectural design have been developed. The complex models and beautiful images that have been produced give the impression that Computer Aided Architectural Design (CAAD) is well-established and succesful. However, in the day-to-day practice of architects its application is limited [3]. Two main reasons can be discerned: First, most systems are far too complex to use by designers, second, the support that these systems offer for the early phases of the design is highly limited. As a result, Computer Aided Architectural Design is mainly used for the final phases of the design, as a sophisticated alternative for the drawing-board.

We think that the usability of a design system in the sketch phase can be improved if as much as possible information is extracted from the input of the user and reused during the design. In this paper we propose a partial solution: geometric analysis. The design is continuously scanned for geometric pecularities, such as coincident faces, orthogonal lines, etcetera, which relations are maintained as much as possible.

In section 2 the background of this work is described: CAAD, VR for architecture, the VR-DIS project, and a global data model for architectural design. Geometric analysis is described in section 3. Starting from an example, we introduce *implicit relations* as a means to improve the interaction. The embedding in an interaction cycle is described in section 4. From the gestural input of the user and implicit relations multiple possible modifications of the geometric model are derived, from which the user selects the most appropriate. Finally, in section 5 future work is described.

## 2 Background

### 2.1 CAAD

Most CAAD packages developed from 2D drawing programs. The emphasis for the application of these packages is on the final design stage when accurate models of the design are created and finally floor plans, cross sections, facades, etc. are plotted. Analyzing these systems two main approaches can be recognized for creating 3D models:

1. The drawing is primarily established in 2D (e.g. a floor plan) and then extruded into the third dimension. The original 2D design is lost and cannot be edited anymore as such.

2. The drawing is primarily created in 2D with drawing entities that already have a 3D representation. The 3D representation is shown separately and can only be edited via the 2D representation. Often to update the 3D presentation of the design the CAD application has to be triggered explicitly.

CAD users and developers try to find ways to use CAD systems not only in the final stage of the design process, but as a design support tool during all design process cycles. More precisely, architectural designers demand design tools that will support their creativity.

Since about five years CAD packages shifted or extended to the early design stage focussing on conceptual models represented by 3D geometries. A common approach in the more recent generation of design tools for manipulation of 3D geometries is by defining building element types with a set of parameters and instantiate them in a design with specific parameter values. In some CAD tools this is a one-direction process, i.e., the geometries cannot be changed afterwards by updating the attribute values. The advantage though is that one can edit the geometry at a lower level (e.g. edges and vertices) creating a desired shape which could not have been achieved by changing the building element attribute values. Other CAD systems (e.g. ArchiCAD) maintain the relationship between parameters and geometry. These parameters are abstracted from a category of building elements defining all variables that can be modified within certain ranges (e.g. length, color, etc). Yet another way to provide a flexible 3D-design tool is by describing a specific shape as a stack of operations (e.g. 3D Studio). After creation of the geometry primitive, every operation with its accompanying parameters is stacked on top of the last one. Every parameter can be changed afterwards as well as the order of the operations. Though the approach is very powerful, it requires a very good understanding of the operation structure. Design strategies for creation of form that are still in the research state are the use of shape grammars and rule-based design. Instead of abstracting from building element geometries, here the geometry creation process is formalized.

## 2.2 VR for architecture

The application of Virtual Reality until now is mainly restricted to realistic visualization of shapes. VR software (such as World Tool Kit and Device) is optimized for generating pictures of a building model consisting of objects with a shape representation and a texture mapping while navigating through the model. Moreover, objects can exibit some behavior like gravity and collision detection. With these capabilities a very realistic impression can be created of a building which only exists in the mind of an architect. The architect himself/herself or the principal can use VR to judge a design on its esthetical and its functional qualities. The display of the design in VR is dynamic, but the design itself is static. To take VR one step further the designer must be able to interact with the design by creating, modifying and deleting design objects and by evaluating the design to certain aspects. Current day-to-day pratice for generating a VR model of an architectural design consists of the following sequence:

1. Create a 3D model (e.g. by Autocad);

2. Attach material properties and/or textures to surfaces (e.g. by 3D Studio);

3. Add light sources and execute lighting rendering (e.g. by Lightscape);

4. Import the 3D model with appropriate surface rendering in the VR environment (e.g. World Up);

From the sequence only step 3 may be omitted if accurate lighting is not considered necessary.

After once having created a VR model, making alterations to the design is a rather tedious process since it requires re-execution of the sequence.

### 2.3 VR-DIS

At the Eindhoven University of Technology a research program called VR-DIS is established to pursue an innovative multidisciplinary design system. We envision a design environment which supports the designer in his/her creativity and in evaluating the design. The design is presented in a interactive environment using VR technology. Tools for shape manipulation are offered depending on the kind of operations (e.g. creating openings). Design information (e.g. mass, cost) is represented in a complex data structure which is manageable in the VR environment as well. Already in the conceptual design stage architectural, physical, structural, and engineering knowledge (from the past) can be consulted and adapted, to improve building efficiency and quality. For collaborative design, building participants can meet in the 'design space', discuss the design and make alterations. Last but not least the user will be invited to take part and give insight about user behaviour and user satisfaction. In the VR-DIS research program there is an emphasis on conceptual design. There are two main reasons to focus on this design stage:

1. Important design decisions are made in the early design stage based on tentative, incomplete and inexact knowledge. This notion has been recognized by many researchers. Our approach towards this problem is in trying to find general proven design and engineering concepts and make them available to the designer who is faced with a new design problem.

2. VR technology performs best in the early design stage. Comparing traditional Windows technology and VR technology against a range of functional aspects of architectural design systems indicates that Windows technology performs best in the final design stage using a 2D representation, whereas VR technology performs best in the early design stage using a 3D representation [6].

The VR-DIS research program results in a platform for CAAD that is implemented incrementally. In this paper we present and discuss our current ideas and directions for further research with respect to geometric interaction of the architect with the system.

### 2.4 Model

The VR-CAAD system we develop will allow for the creation and manipulation of geometric entities at different levels of abstraction. We distinguish three layers in the system, namely:

1. *Product model description.* On this level the geometry of a building element is described using a semantically rich syntax. This means that the geometry and its relationships are described in terms of the corresponding building elements, such as walls, floors, etc. Product models consisting of type definitions and their instances are usually implemented as design databases. Typically the available data can be processed by domain specific applications (e.g. Finite Element Analyses).

2. *Geometric objects.* On this level geometry is described with primitives such as cuboids, rectangles, cylinders, etcetera. Parametrized geometries of building elements are built on top of these primitives, they are the basic entities used to create any kind of (architectural) shape.

3. *Geometric elements.* On the lowest level the geometry is described as vertices (points), edges (lines), and faces (polygons). Polygons are easy to process by computer graphics hardware to obtain maximal frame rate performance. User interface interaction like selecting, dragging, etc. is typically executed through identification and recalculation of points, lines, and faces.

The challenge for the development of the VR-CAAD system is that the designer wants to communicate on the product model layer, while on the other hand he/she wants to interact on one of the lower levels. In the following sections we discuss how this can be done.

## 3 Geometric analysis

On each of the previously defined layers the information can be modelled as a set of objects, where each object has a type, parameters, and relationships with other objects. For instance, a building element can be a wall with a certain thickness that separates two spaces; a cuboid is a geometric element with certain dimensions that belongs to a particular group of elements; a vertex has coordinates and occurs in edges and faces.

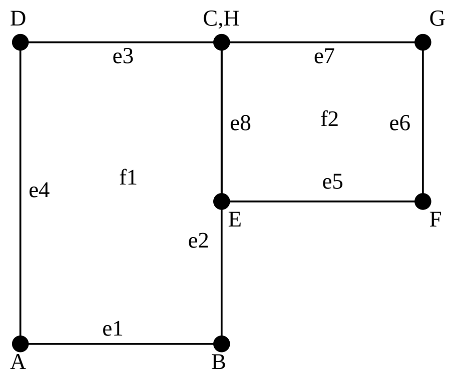



Figure 1: Two rectangles

As a simple 2D example, figure 1 shows a very simple floorplan, consisting of two rectangular spaces. The corresponding data structure for the two lowest levels is shown in figure 2. On each level a complete geometric model can be defined with a minimal set of information, for instance just by enumerating the geometric objects and their position, dimensions, and orientation. During interaction all levels have to be kept consistent with each other, which is often not trivial.

Suppose that the architect modifies his model via manipulation of vertices, lines, and faces. A key question is:

> *Given a certain manipulation, how can we select the most meaningful realisation of this manipulation?*

For example, suppose that the user moves point C in figure 1 in vertical direction. This can be interpreted in various ways:

1. The whole design must be translated;
2. Vertices D, C, G, and H must be translated;
3. Vertices D and C must be translated.

The most probable interpretation here is option two. The user has moved a point, which indicates that he wants a local operation, hence option one is probably not desired. Option two now keeps most of the structure intact. However, we are not certain, and there are situations where the choice is less clear. How can we make the right choice?

We expect that geometric analysis can provide a partial, though significant answer. In a geometric model many subtle relationships can be discerned. If we look at the example, we see that the rectangles touch each other, that vertex C and H coincide, that vertex E is located on edge e2, that e2 and e8 partially coincide, and that e3 and e7 are in line with each other. We call such relationships *implicit relations.* They emerge when a user builds up and modifies his model.

We assume that such relations are not coincidential, but reflect the intention of the architect. For instance, in the product model relationships between building spaces or building elements are typically described in terms derived from natural language such as, next to, on top of, etc.

Many different types of implicit relations can be defined, in which two or more elements of the same or different types participate. As an example, some relations between two rectangles are shown in 3.

We can now reformulate our original question as:

> *Given a certain manipulation, which implicit relations must be maintained, disabled, and added?*

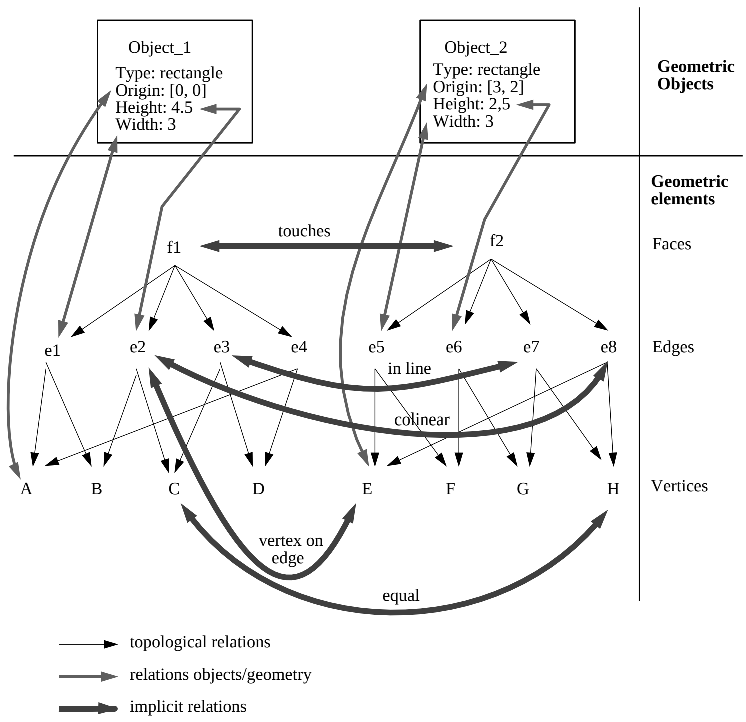



Figure 2: Data structure

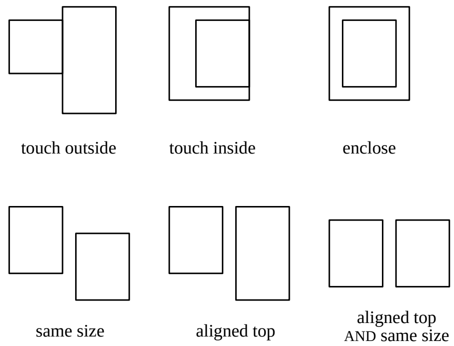



Figure 3: Relations between rectangles

Implicit relations that are important for the user must be recognized and used as invariants, possible at the expense of less important implicit relations that are disabled. Furthermore, new patterns in the mutual relations between objects must be recognized and added to the implicit relations. Note that implicit relations are redundant. Both the representation in terms of geometric objects, as well as the representation in terms of vertices, lines, and faces are already complete in themselves.

In the ideal case this whole process is automated. The system continuously extracts as much information from the input of the architect as possible, and reuses this information during subsequent interactions. Or, formulated differently, we aim at a system that understands what you mean. Having said that, we realize that this goal is hard, if not impossible, to reach.

The major problem is the detection and weighing of the implicit relations. Which ones are meaningful, which are not? Given multiple solutions, how to find an optimal one? These questions often cannot be answered without active participation of the user. Intelligent systems have often a habit of making the wrong guess. In the next section we discuss how interaction can benefit from implicit relations without being bothered by misinterpretation on the side of the system.

Another problem is to find solutions numerically that satisfy a certain set of implicit relations. We expect that local propagation is a viable route for this. Starting from the object that is manipulated, its neighbours are considered, if not all relations can be handled satisfactorily, their neighbours in turn, and so on.

## 4 Interaction design

The system that we intend to build should be as platform independent as possible. To explore the limits a large set-up (for instance a CAVE [2]) would be ideal, however, for application in practice by architects, it is more realistic to assume not much more than a PC, possibly extended with stereo graphics. Projection based systems, such as a Responsive Workbench [4] could be an attractive compromise. Anyhow, the method we propose here is independent of a particular hardware environment.

In any environment, the design of the interaction is critical. We aim at a system where the user inputs and modifies his design mainly via gestures, to approximate the natural way an architect designs a building via sketching as close as possible. Overviews of inspiring work in this area can be found in [1] and [7]. These gestures are interpreted by the system. The type of action and the object acted upon are derived from the shape and scope of the gesture. Also here understanding plays a major role: The system has to determine the most probable interpretation.

The main actions are creation, transformation (translation, rotation, and scaling) and deletion. New objects are created by sketching on surfaces. If the scope is taken into account a few strokes suffice to define the type and the dimensions of new objects. A Γ-shape drawn on a wall defines either a window or a door, dependent on the position of the lowest point; a straight stroke on a floor defines a wall, a small circle on a floor a column.

Transformation is probably the most important action. A natural way to support design is by adapting parts of other existing designs into a new situation by means of (geometric) transformations. Also, during the design the architect wants to consider many variations of dimensions and compositions of the building objects. A well-known way to implement this is to let the user select an object upon which a control-box with handles appears. The user can select a control-point and drag it. Another method is to let the user manipulate individual vertices, edges, and surfaces directly. In principle this is more efficient, because the user can perform such manipulations with a single gesture, without the separate selection step of the standard method, but this has to be confirmed by experiments.

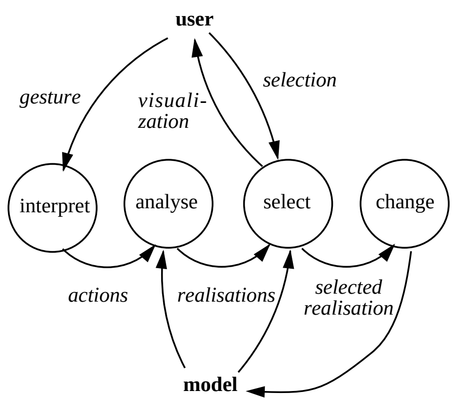



Figure 4: Interaction cycle

Deletion can be specified rather easily: The user erases a part of the design by making a Z-stroke on its surface.

The result of a gesture is one or more low-level actions, such as "Create a door in this wall at roughly this position" or "Move this vertex by a certain distance in a certain direction". Multiple alternative actions can result when the gesture is ambiguous. In the next processing steps these actions are translated into changes of the model. Figure 4 shows in a data-flow diagram how we envision this. The actions are analyzed based on the implicit relations, which leads to several different realizations.

We foresee two ways to proceed. The first is to extract more information from the gestures of the user. A slow gesture could indicate a global change, while a fast movement could mean a local change . With a fast movement the user breaks up a connection. Also, two-handed input can be used. The user fixes with one hand a part of the design, while translating another part with his other hand.

The second, and most important way to resolve ambiguities is to let the user decide. We do not believe in infallible intelligent systems. In the end, the user is the only one who can decide what's the right interpretation. To support him, multiple alternatives are shown, from which the user picks the right one. This can be implemented in various ways. Alternatives can be shown sequentially, upon which the user confirms this alternative or steps to the next. Alternatives could be shown simultaneously in separate windows. Such a solution could work in a desk-top setting, but requires much screen space, and requires the user to shift his focus. A challenging solution is to show all alternatives simultaneously in the same view, for instance as line drawings with different colors. The user can then pick the right one by specifying the color. Speech input seems a natural choice here.

After the selection of a particular realisation, the system updates the model: the geometry is changed and implicit relations are added or disabled.

## 5 Discussion

We have presented an overview of our ideas for a Computer Aided Architecture Design system. The central theme is to provide the designer with a natural and understanding system, such that he can focus on his design task. To realise this, we propose to extract and reuse as much information as possible from the input of the designer:

- The shape and scope of the gestural input of the user is analyzed;
- Via geometric analysis a set of implicit relations between geometric elements is maintained and applied during the interaction.

Our next step will be the implementation of the system. This will be a large task, during which many details have to be elaborated, and new problems and directions for extensions will show up. Also, we realize that it is hard to predict which choices are most natural and understandable in practice. We therefor will develop the system cyclically and iteratively. We start with a limited shape domain, possibly even 2D, to evaluate the basic algorithms and test their usability. In next phases we will gradually expand the shape domain and other functionality.

During this next phase we will try to find answers to several open questions. For instance, we are not sure if it will be possible to hide the implicit relations completely from the user. If not, the next question is how to visualize and let the user interact with them. Another large area for future research concerns the shape domain, especially the integration of free-form surfaces within this concept.

## References


[1] Billinghurst, M. Put that where? Voice and Gesture at the Graphics Interface. *Computer Graphics* **32**, November 1998, p.60-63.

[2] Cruz-Neira,C., D.J. Sandin, T.A. DeFanti, R.V. Kenyon, and J.C. Hart. The CAVE: Audio Visual Experience Automatic Virtual Environment. *Communications of the ACM 35,* 6, juni 1992, pp. 65-72.

[3] Dorsey, J., and L. McMillan. Computer Graphics and Architecture: State of the Art and Outlook for the Future. *Computer Graphics* **32**, February 1998, p. 45-48.

[4] Krüger, W. et al. The Responsive Workbench: A Virtual Work Environment. *IEEE Computer 28,* 7, juli 1995, 42-48.

[5] http://www.ds.arch.tue.nl/Research/program

[6] Vries, B. de, and H.H. Achten. What offers Virtual Reality to the Designer? In: *Proceedings of the Int. Conference on Integrated Design and Process Technology,* Berlin, Germany, 1998.

[7] R. Zeleznik. Sketching in 3D. *Computer Graphics* **32**, November 1998.

# Spoken Language Interaction with a Virtual World in the MUeSLI Multimodal 3D Retail System

Peter J. Wyard, Gavin E. Churcher
BT Laboratories,
Martlesham Heath, Ipswich, Suffolk,
IP5 3RE, UK.
{peter.wyard| gavin.churcher}@bt.com

## ABSTRACT

This paper describes a 3D multimodal spoken dialogue system in the home furnishings retail domain. Input is via spoken language, touch or the virtual world, and output is via a talking head (virtual assistant), 2D palette and the virtual world. The different input and output modalities are tightly integrated on a single dialogue turn. After giving an introduction to the MUeSLI system, this paper focuses on the architecture and methods used to integrate the virtual world with the spoken language processing, and some features and quality issues of the virtual world in such a system. Finally we discuss two research issues: the user's perception of the virtual assistant, and the effect of the virtual world on the style of interaction.

**Keywords:** Spoken Language, Spoken Dialogue, Multimodal Dialogue, Multimodal Integration, Multimdal Architecture, 3D Retail, Virtual World Interaction

## 1 INTRODUCTION

The aim of this paper is to describe the multimodal 3D retail system which we have recently completed at BT Labs, with a particular focus on the user interaction with the virtual world which is a key part of the interface. The MUeSLI system has spoken language interaction as a major feature, and the paper will describe how spoken language has been integrated with the virtual world. It will do this both from the user interface point of view, and in terms of software architecture. Other publications cover the overall MUeSLI programme (from focus groups to user trials of the automatic system) and how it is tackling the issues of multimodal integration [1], a detailed account of our approach to multimodal Wizard of Oz trials [2], and the detailed functionality and architecture of the overall system [3].

### 1.1 INTRODUCTION TO THE MUESLI PROGRAMME

Before proceeding to describe the system, it is helpful to give a resume of the MUeSLI programme. MUeSLI stands for MUltimodal Spoken Language Interfaces. It has long been our assumption that natural spoken language will add greatly to the effectiveness of human-computer interfaces, and much work is ongoing at BT Labs to develop unimodal speech-in, speech-out systems targeted at traditional telephony services. However, it is well known that humans use a combination of communication modalities when they converse (body language, facial expression, gesture, etc.), which adds considerably to the power of spoken language alone. We assume that the same will be true of human-computer interfaces, although in a different way. Multimodality should yield both *synergy*, where the combined effect of the modalities is greater than their sum, and also *error compensation*, where for example speech recognition errors can be recovered from using information from the other input modalities.

The time appears to be right for the development of multimodal interfaces in two respects. Firstly, multimodal interfaces rely on multimedia services being available to interface to. In the last couple of years there has been a very rapid rise in multimedia, largely fuelled by the rise of the World Wide Web (WWW), and multimedia is a fast growing sector for major telcos such as BT. Secondly, a number of component technologies are approaching maturity (speech recognition, speech synthesis, touch-screens, data gloves, eye-tracking etc.) with others following on behind (e.g. gesture recognition).

The MUeSLI project at BT Labs has two fundamental aims:

- To research the design of innovative and effective multimodal interfaces
- To investigate the robustness of such interfaces with real users

In order to support these aims, we have built a multimodal advanced spoken language interface to a useful multimedia application, in order to act as a research vehicle. (Note that in addition we have a programme of more formal research experiments to address particular issues which arise, such as user preferences for different synthetic personae, or the effect of multiple modalities on the nature of the dialogue). We are focusing on human-machine dialogue systems, in distinction to other multimodal work on human-human communication in shared spaces.

Neither the technology nor the human factors issues associated with multimodal interfaces are very well understood. We therefore considered it best to proceed in a staged approach, from paper exercises to a fully automatic system, as follows:

1. Storyboarding possible system interactions with a focus group, using pictures of the system's output screen. This is a low-cost, quick and flexible approach which allows a wide range of ideas to be explored, but its disadvantage is that it is very unlike the real system, and may not give accurate user predictions as to what they would like.
2. Exploring different system designs with a focus group, using a set of pre-canned (but real) output screens. (We hope to publish the results in future).
3. Performing Wizard of Oz simulations of successive versions of the system with a user [2].
4. Constructing a complete trial system.
5. Performing an incremental series of user trials.

The staged approach has two main benefits. Firstly, it enables us to maintain a user focus, and adapt the system we build to what users find helpful, rather than being purely technology driven. Secondly, the system remains more flexible longer into the programme, and is therefore more useful as a research vehicle for answering our basic questions about multimodality.

We have now completed stages 1 to 4, but no user trials have yet been performed.

## 1.2 Description of the System

At the start of the programme it was very important to choose a suitable application environment. The application chosen was an existing 3D retail system (internal to BT), whose visual nature provided ample scope for multimodal interaction, and whose database and functionality were rich enough to provide a decent challenge to the language component, without creating too large a vocabulary for the recogniser. It also gave the opportunity to investigate the combination of speech input with a virtual world interface.

The 3D retail system is intended to be deployed either as an internet service, or as an in-store kiosk application. To the original mouse-driven system we added a touchscreen to allow touch input, advanced spoken language capabilities, a synthetic persona (talking head) to act as a virtual assistant, extra functionality and some system intelligence. In fact, the primary interaction metaphor has been changed from controlling a screen to conversing with the Virtual Assistant. See Figure 1 for the multimodal system interface, with its 2D fabric palette, 3D virtual world (VW) and virtual assistant.

Fabrics may be selected according to a wide range of criteria, and placed on furniture, curtains and walls in the virtual living room or displayed on the palette. The user can refer to objects in the virtual room and to fabrics on the palette by speech or touch, and talk about them with the assistant. The user can combine speech and touch inputs in a single dialogue turn. He can ask for prices of fabrics or complete selections, for cheaper fabrics, etc. He can ask for advice on which fabrics would suit with a given fabric. He can move around the room and inspect the contents from different distances and angles. He can move furniture within the room. A natural dialogue is possible with the assistant, catering for various forms of reference (anaphor, deixis, etc.) and ellipsis. The assistant is able to take the initiative at certain points in the dialogue, currently to offer fabric matching advice and to suggest movements within the VW.

A snippet of typical dialogue will give the flavour:

User: Show me some plain green fabrics in linen.

System: There are more than ten fabrics. Here is a selection. [displays 6 fabrics on palette].

User: Put this one on the curtains [points to Plain Slub Velvet fabric].

System: Plain Slub Velvet is not a drapes fabric.

User: Put it on the sofa. (anaphor).

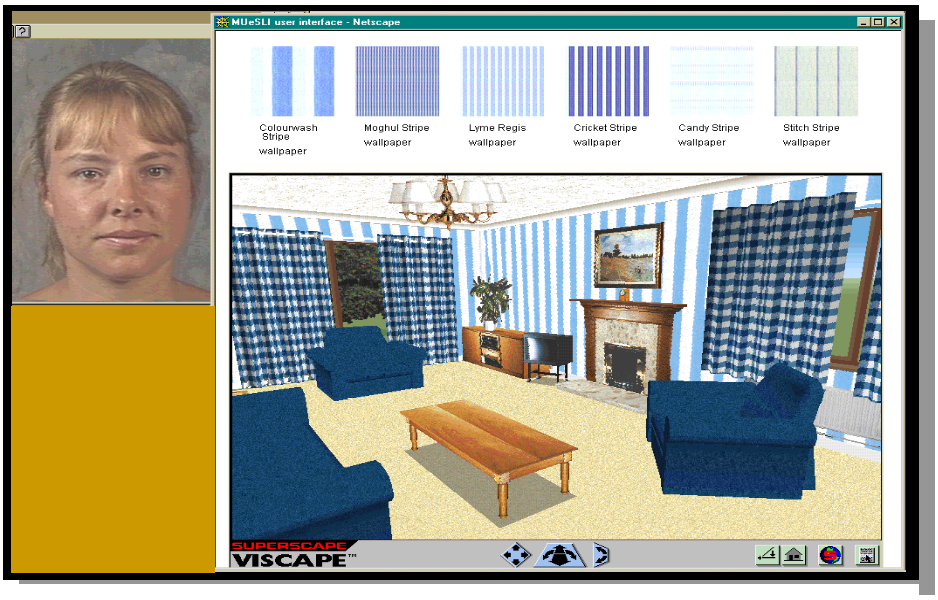



Figure 1: User Interface

System: [transfers Plain Slub Velvet to the sofa].
[pause]

System: Would you like to view the room from the door? (system initiative)

User: Move this chair over here. [points to chair and position in turn]. (user ignores system)

System: [moves chair to new position].

User: How much is this selection?

System: This design costs three thousand, five hundred and fifty pounds.

User: What cheaper upholstery fabrics do you have?

System: Here is a selection of cheaper upholstery fabrics.

## 2. MUESLI SYSTEM ARCHITECTURE

### 2.1 OVERVIEW OF ARCHITECTURE

The MUeSLI system comprises two main parts: the Natural Language (NL) Server and the user interface. Each part comprises a number of heterogeneous components, as shown in figure 2. Whilst the NL server would normally run on a dedicated server, the user interface is designed so that, with the correct client plug-ins, it can be run and deployed over the World Wide Web (WWW). The Interface Control component and palette of swatches are contained within a Java Applet that can run in a standard WWW browser, such as Netscape Navigator. The Virtual World is a plug-in in the same browser window. We chose to implement the Interface Control component and palette of fabric swatches in Java since it is readily equipped to handle network-based communication between modularised components, where the components could be on different machines. Java also gives us portability of platform, and is able to be downloaded over the Internet as an Applet in a WWW browser window.

Communication between the server and each of the interface components is via TCP/IP. The server is able to receive information from each of the components, for example, when a user has touched an object in the Virtual World, and also to send commands, such as to update which fabrics are shown on the palette. Commands between the server and either the palette and/or the Virtual World is via the Interface Control component, which can take high level commands and translate them into the commands required to control the Virtual World. We are currently using a Superscape Viscape Virtual World which has been designed using Superscape's proprietary tool, VRT. The Virtual World is rendered using the Superscape Universal (5.61) plug-in, available for both Internet Explorer and Netscape Navigator WWW browsers.

The server can send text to be spoken by the Virtual Assistant, a 3D talking head developed by BT Laboratories which is able to lip-sync and make small facial movements whilst playing Text-to-Speech from the Laureate TTS system [6]. The server is also able to receive input from the speech recognition client, a Java application which utilises the Java Speech API to control the speech recogniser. Currently, we are using

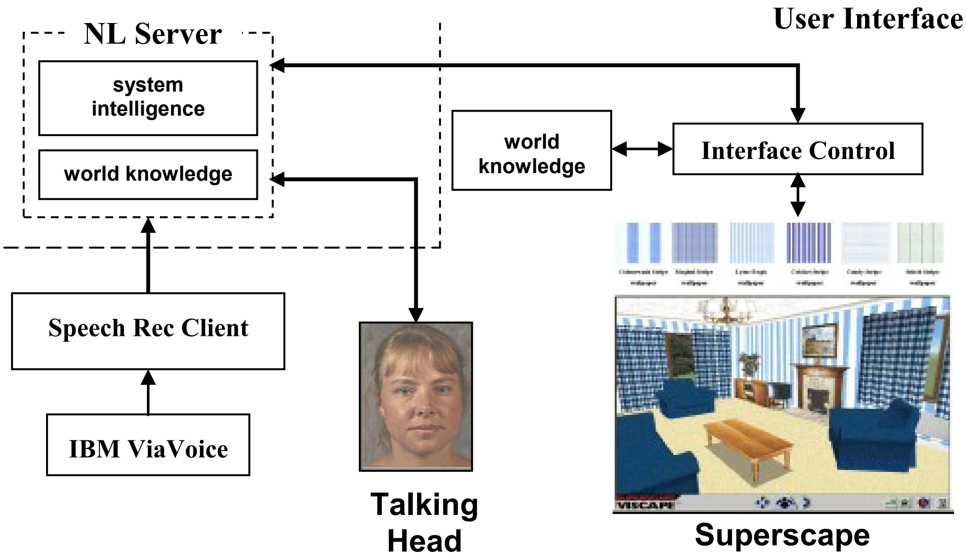



Figure 2: The MUeSLI System Architecture

IBM ViaVoice Executive which allows flexible speaker-dependent continuous speech recognition. The Speech API allows us to either specify a finite-state grammar for the recogniser or a 'dictation' mode. Initial impressions are that using the extensive finite-state grammar gives reasonably adequate speech recognition accuracy without training speaker models, whilst 'dictation' mode is not sufficiently accurate given the high number of words in the vocabulary, upwards of 64,000 words, and would particularly benefit from speaker training. One advantage of using the Java Speech API is that we are able to use different speech recognisers with only minimal changes to the system and the interface to the server, providing the speech recogniser itself supports the API.

## 2.2 NATURAL LANGUAGE SERVER

The NL server is written in Quintus Prolog and contains all of the system intelligence required to control the interface and interpret what the user has said and touched in the Virtual World. This enables the Interface Control in the user interface to remain computationally inexpensive and fairly easy to develop the software. Individual technologies, such as the Viscape plug-in and the speech recogniser, are computationally intensive at the current time, but we expect this load to diminish as the technology improves and the processing power of computers for multimedia increases.

The NL server's role is to interpret the user's interaction in the context of the current dialogue, and respond appropriately. It receives inputs from the speech recognition client in the form of Prolog terms indicating one of the following: the start or end of speech has been detected, and the decoded speech from the recogniser. The start and end of speech is simply triggered using audio levels from the microphone. A short time after the end of speech is detected, the recogniser returns the best hypothesis of what was said. This is then passed on to the server for processing. The other inputs to the server are the touches recorded when the user touches either an object in the Virtual World or one of the fabric swatches on the palette. Touches on the palette are simply returned given the palette index. The server translates this index into an identifier for the fabric touched by using information stored in its world model.

Detecting when the user has touched an object in the world is slightly more complicated. Whilst it is possible to have objects activate code internal to the Virtual World when they are touched, in Viscape it is only possible to detect this externally by the Interface Control when the object has been created externally by the Control and placed in the Virtual World. Such objects, when touched or activated, call an external function in the Java applet, Interface Control. This function can be customised so that the touch can be relayed back to the server.

Given these inputs, the first stage of the NL server is to decide when a user's input turn is complete. This is controlled by an event handler which has application-specific information about the relative timing of the various touches, speech decodes etc. This allows us to determine when the user has finished

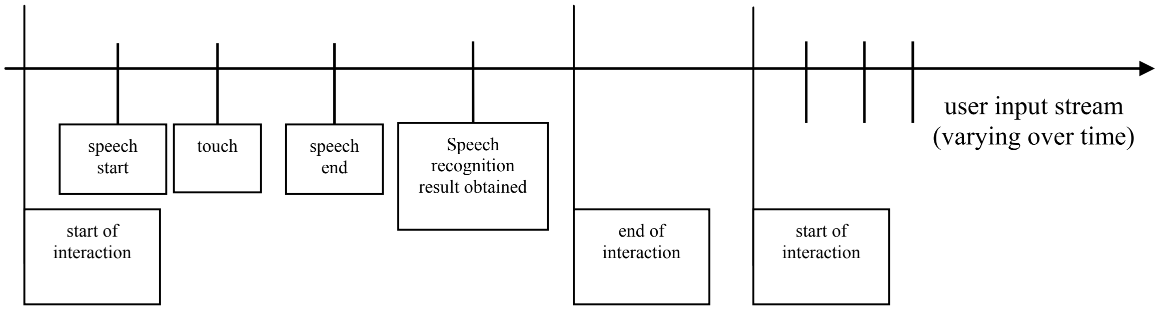



Figure 3: Segmenting the User Input Stream

interacting and it is the system's turn to respond. Figure 3 shows the problem of segmenting the inputs from the user stream.

The next stage is to integrate the touch and speech modalites using the Modality Integrator component. This component combines the different modalities into a single semantic representation of the user's interaction. It contends with the ambiguity and errors which occur in each modality and through unification, many hypotheses can be eliminated, leading to hopefully one or a small subset of possible interpretations. For further discussion on the processes involved in integration, see [1].

The NL server can control the external behaviour of the system, in terms of what is said and done. This incorporates a Dialogue Management component, designed so that reasonably complex dialogues can take place with the system adopting particular strategies. For example, the system may adopt a salesperson strategy when selling fabrics, or it may offer the expertise of an interior designer suggesting advice on fabric combinations.

## 2.3 WORLD KNOWLEDGE

Different types of world knowledge are contained in both the NL server and the Interface Control. The latter's role is to interpret high level commands from the server in order to control both the palette and the Virtual World. The server contains the intelligence which determines, for example, how to move objects around the room. Consider the following example: the user points to the sofa and says, "put that over there", and points to a position in the room. The position indicated is large enough for the sofa, but there is a chair in the way. The system can choose one of three actions: refuse the user's request, move the sofa to the position by moving the chair to one side, or swap the chair with the sofa. It chooses the last, and sends the following high level commands to the Interface Control: hide the sofa and the chair, place the sofa in the chair's position and the chair in the sofa's previous position. The Interface Control translates this into a representation that can be understood by the VW. Using functions provided by Viscape the Interface Control first sets the objects' visibility to false, moves them to specific locations, and then enables their visibility.

The NL server models a number of aspects of world knowledge. This includes the current state of the world: which fabrics are on the palette of swatches and on each item of furniture, walls and curtains, and the locations of each item of furniture. The server also contains information about a moderate number of fabrics and their properties, such as colours present, material, patterning, cost etc. The Interface Control's world knowledge essentially provides information for the translation of high level commands to low level commands for the Virtual World. This enables the Interface Control to be a light-weight client which has little intelligence in itself. At the present time, the functions performed by the server, such as calculating how to move objects around the room, are not particularly complex, but could be in a future system. This allows us to use a powerful computer to perform these functions, regardless of the machine that the user interface is running on.

# 3. FEATURES AND QUALITY ISSUES OF THE VIRTUAL WORLD

For our application of 3D retail, we require a number of features present in whichever Virtual World system we employ. Understandably, our first requirement is to be able to render a realistic sitting room that shows a number of items of furniture. The rendering quality is of particular importance in this domain, since it is vital that the customer using the system is able to get a feel for way the fabric appears, especially in juxtaposition to other fabrics. There is the need to empirically find which factors are of most importance to the user – for example, accurate representation of the colours is likely to be important since it has been found to be vital in electronic catalogues of fabrics, clothing etc. If the system is offering the furniture for sale, then the furniture objects must be rendered to a high standard, otherwise a lower standard will be acceptable. The hanging and folding of the curtains can give useful information about their texture, but may be computationally expensive to render. In a similar vein, we need to be able to accurately scale each pattern on objects depending on which object it is. A number of fabrics can be both upholstery and curtaining materials, and need to be scaled differently depending on which object they are applied to.

Another feature is the ability to move the user's viewpoint, simulating different kinds of movement. Currently, the user can command the system to go and look at another part of the room, examine an object, such as the sofa, in greater detail, or be taken on a predefined tour of the room. Rather than flicking between viewpoints, we have implemented a method of flying smoothly between points, gradually rotating the user's viewpoint. The effect is a rather cinematic panning motion, and after examining a number of alternatives, is the most effective for this application. The minimum acceptable speed of movement in the VW (and the associated frame rate) must be discovered empirically. It is important to offer smooth movement between points, however rendering a detailed room that contains many objects with high resolution textures on them is computationally expensive to the point that the movement can appear a little fragmented on machines that do not have adequate graphics acceleration, especially given the heavy demands on the processor by the speech recognition and talking head.

Poor VW performance also spoils the otherwise very powerful result of combining speech with VW movement (e.g. "Let's have a look at it" followed by a swift and smooth movement to a close-up of an accurately modelled sofa with a faithfully reproduced fabric on it, or by a jerky, slow approach to a very crudely modelled sofa). In order to compensate for this, it is possible to employ a visual trick: objects in the distance appear the same whether they are rendered in high quality or not. It is thus possible to render objects in the distance as much simplified objects with lower resolution textures. When the user reaches a specified distance away from them, the objects are replaced by a high resolution version. If the user is moving to look at a particular object such as a chair, then the surrounding objects do not need to be rendered to such high resolution whilst the user's viewpoint is moving since we can assume that his or her attention is on that particular object. Alternatively, it may turn out that although this is functionally adequate in a strict sense, it does not look good to the user. Finally, it should be noted that variable resolution may lead to variable speeds of movement in the VW, and it is important to avoid noticeable change of speed because, for example, a distance threshold has been passed.

A feature which has already been mentioned above, is the ability of the Interface Control to detect when the user has touched an object in the Virtual World and to identify which object it is. This is applicable to each object in the world, from the furniture to the fireplace, since the system must be able to pinpoint wherever the user is touching in 3D space, for example, when the user is moving the furniture around the room. The ability to control the location and appearance of objects such as the furniture leads on to our last requirement: to be able to make an object invisible/visible and to be able to place it exactly in <x,y,z> space with a particular rotation.

The above features led us to consider the Superscape Viscape Virtual World, which is able to implement all of the above features whilst providing a limited external interface to programming environments such as Java and C. To implement some of the features, it was necessary to utilise the scripting feature of Viscape. This does mean that transferring the system to use another Virtual World will require more work than if the controlling code was contained solely in either the NL server or the Interface Control components, but it is hoped that any other Virtual World system that we use would have a similar or greater level of functionality.

There are a number of additional features which are not essential to the application, but may be of use when developing a 'natural' interface. One such feature is a shared perceptual model of the world, i.e. shared between the NL server and the VW. This means that the NL server knows which objects are in the user's field of view in the VW. When there are a number of fabrics on the palette, only one of which is red, the user can ask the system, "put the red fabric on the sofa". In this instance the intended fabric is unambiguous and the server can use its world model of which fabrics on the palette at any particular time. In contrast, when the user moves around the room, he or she may have only one of the two chairs in his or her field of view and may ask the system, "put the red fabric on the chair". To the user it is unambiguous which chair is meant; however, without accurate modelling of where the user is in the Virtual World and in which direction he or she is facing, it is impossible for the system to determine which chair the user is referring to. In this case, the system would need to ask the user to unambiguously indicate the chair, for example, by touching it. Ideally, the NL server would be able to model which objects are in the user's field of view by knowing where the user is in the room. Currently, the server can determine the user's viewpoint at specific points, for example, after he or she has asked to go to specific point in the room. However, the user is able to stop whilst in mid-flight and even to take hold of Viscape's movement controls to move freely around the room without the intervention of the server. In these cases it is impossible to determine exactly where the user is using Viscape's Virtual World.

## 3.1 CREATING THE MUESLI VIRTUAL WORLD

The Virtual World and each of the objects within it were created using Superscape's VRT package. Whilst some objects, such as the fireplace, are simply one-sided objects which have a texture applied to them, others including the furniture are multifaceted to give them a degree of realism. The fabric textures were scanned in from either a paper catalogue or from actual samples of the fabric. The process of scanning involved using a high resolution colour scanner to initially scan the fabric and then post-editing to create a square of fabric of a standard size which could then be used as a repeat pattern. The repeat pattern is used to texture an object by uniformly tiling the graphic on particular surfaces. On each object we coded a scale factor so that although the scanned graphics were of a standard size, they would appear accurately scaled in the Virtual World. In order to create a realistic application, we scanned in a total of 140 fabrics, entering in each one's details into a database which could be referenced by the NL server.

## 4. RESEARCH ISSUES

The work done to date within the MUeSLI programme has raised many issues which may be of general interest to those working on similar systems. In this section we cover just two of these issues. At this stage, we are unable to give scientific answers to these questions, based on formal experiments or user trials, but we hope that sharing our informed opinions, based on experience, will be of benefit. In addition to the user trials mentioned above, we have in place a complementary programme of research, which aims to address some of these issues in a formal way.

## 4.1 USER'S PERCEPTION OF THE VIRTUAL ASSISTANT

In the MUeSLI system, the primary interaction metaphor is conversing with the Virtual Assistant (VA), so the VA is regarded as an essential part of the system. Initially, we have selected a realistic talking head with state-of-the-art text-to-speech synthesis, and we chose a female head after feedback from the focus groups in stage 2. However, there are many issues surrounding the use of the VA. A basic need is to gain empirical findings on user preferences for different possible VA's, as many types of VA have been used in development systems and made available to other developers, but little formal knowledge has been gained through user experiments. We are conducting a formal experiment to answer this question. In the early stages of this experiment, users will be exposed to a wide range of VA's, such as human video, human still, human still + moving lips, talking heads, a disembodied voice, non-human VA's such as a talking animal or a cartoon character, and inanimate VA's. Most of these VA's will be in a separate window, as in Figure 1, but we will also have a condition where the VA is in the VW itself. In these early stages, users will passively observe multimodal dialogue sequences, and then fill in questionnaires (asking about several different dimensions of preference) and be interviewed. A smaller number of VA's will then be selected for the final stage of the experiment, in which users will interact with the automatic system. We expect the preference for VA to be application dependent. Initially we are focussing on retail applications, but we are comparing two scenarios, in which we hypothesise that user preferences may be different: the furnishings scenario described in this paper, which is fairly sober and serious, and where the goods are expensive, and a "cool CD shop" scenario having opposite characteristics.

The experiment described above will hopefully provide information about general dimensions of user preference, such as "liking", "realism", "quality", "usefulness". Dimensions such as "trust" will need to be investigated subsequently in an experiment where the user has to put real trust in the VA, e.g. by making a real transaction, or where a prize may be won.

One of the underlying motivations for this experiment is that some VA's are much more demanding than others in terms of software development costs and processor power, e.g. realistic 3D talking heads with accurate lip synchronisation and realistic facial muscles. We need to know if such VA's are worth the overhead.

A related issue to the user's perception of the VA, is the effect which the VA has on the style of interaction. For example, does a realistic human VA encourage greater use of speech input, or more complex speech input? At present the MUeSLI system forces users to use speech as one of the input modalities in most input turns, but if users were able to drive the system purely by mouse, would they be more likely to do so with an inanimate VA such as a paper clip rather than a human VA?

## 4.2 Effect of the Virtual World on the Style of Interaction

When one has a multimodal system with input via speech, a 2D palette, and a 3D virtual world (VW), the style of interaction will have elements of the styles from systems with only one input modality. However, there will also be cross-modality effects. It is known, for example from the work of Oviatt [4],[5], that in a multimodal system with 2D graphics and speech input, the graphical input modality affects the nature of the spoken language. The language shows an increased use of references (particularly deixis), the sentences become shorter on average, and the phenomenon of alternation of modalities may arise, where the user oscillates between primary use of one modality or another, triggered for example by too many system error responses with the current modality. It will be interesting to investigate how the incorporation of a VW affects the other modalities, and in particular, whether there is an effect on the spoken language which is different from that observed for 2D graphics as an input modality

## References


[1] Churcher G.E. and Wyard P.J, Multimodal Integration in the MUeSLI Project, to appear in the *Proceedings of the 6th IEEE International Conference on Multimedia Computing and Systems (ICMCS'99)*, Florence, June 1999.

[2] Wyard P.J. and Churcher G.E. A Realistic Wizard of Oz Simulation of a Multimodal Spoken Language System, in the *Proceedings of the 5th International Conference on Spoken Language Processing (ICSLP'98), 1998*.

[3] Wyard P.J. and Churcher G.E., The MUeSLI Multimodal 3D Retail System, to appear in the *Proceedings of the ESCA Workshop on Interactive Dialogue in Multi-Modal Systems*, June 1999.

[4] Oviatt S. Integration and Synchronization of Input Modes during Multimodal Human-Computer Interaction. *In Referring Phenomena in a Multimedia Context and Their Computational Treatment*. Ed. Andre E., 1997.

[5] Oviatt S., "Toward Interface Design for Human Language Technology: Modality and Structure as Determinants of Linguistic Complexity". *Speech Communication*, 15(3-4) pp283-300, 1994.

[6] Page J.H., and A.P. Breen, "The Laureate Text-to-Speech system: Architecture and Applications". *BT Technology Journal*, January 1996, Vol. 14, No 1

# Real Time Gesture Based 3D Graphics User Interface for CAD Modeling System

J. M. Zheng, K.W. Chan, I. Gibson
Department of Mechanical engineering
The University of Hong Kong
Pokfulam Road, Hong Kong

## ABSTRACT

As computer technique progressed, a greater emphasis has been placed on intuitive human-computer interaction (HCI). Virtual Reality systems can offer a novel way for users to interact with the objects in the computer generated environment (the so-called Virtual Environment, VE). Through VR technology, we have the ability to replace the traditional input device, such as keyboard and mouse, with other modes such as speech and gesture.

In our research project, we will use CyberGlove, developed by Virtual Technology Inc., as an input device to develop a desktop CAD modeling system for conceptual designers. We elaborate the limitations of the Dataglove and use gestures to support intuitive human-computer interaction. To develop this gesture interface, we emphasize that the conceptual designers are allowed full-freedom to use different kinds of gestures to conduct various geometric shape operations instead of dependence solely on keyboard and 2D mouse. The designers can indicate objects or directions simply by pointing with the hand, and they can look at an object from a different point by just grasping and turning it. The “virtual tools” can be used for shaping, cutting, and joining objects. We employ the 3D GUIs for enhancing the gesture interface. In the context of the objects in the VE, the 3D menu and “virtual hand” float over the objects rather than being part of the scene. Various 3D cursors can be used to select from a menu or to manipulate the objects.

**Keywords:** Computer Graphics, Virtual Reality, Gesture Interface, Human-Computer Interaction

## 1 INTRODUCTION

As computer technique progressed, a greater emphasis has been placed on intuitive human-computer interaction (HCI). For many years, the prevalent input device was the keyboard that allowed users to input their command lines to the computer. As the time progressed, the advent of the mouse changed the face of computing applications and the “window” was born. It was this simple input device, within the price range of the masses, that revolutionized the computer industry. Now the majority of users work on a desktop computer with a keyboard, mouse or trackball and a monitor. The techniques for the tools of HCI can be distinguished: command language, function key, menu selection, icon and window[1]. These five aspects can be cast into three different interaction styles: command language, menu selection and direct manipulation.

Fortunately, many applications are changing and moving to 3D graphics, more and more researchers emphasize the development of human-computer interaction, for example, interfaces that can be used without learning. Technical developments could stimulate the interface researchers to make significant advances on the fundamental problems. Virtual Reality systems can offer a novel way for users to interact with the objects in the computer generated environment(the so called Virtual Environment, VE).

The advantage of VR can put users into a fully modeled, intangible world[2]. Users are most often part of the world where they interact with the objects. Through VR technology, we have the ability to replace the traditional input device, such as keyboard and mouse, with other modes such as speech and gesture. They can be used to create a preferred mental model for the users to reduce the mental workload by allowing recognition rather than recall. The gesture interface can bring us closer to intuitive interfaces so as to allow the users to directly manipulate the objects in the Virtual Environment(VE).

Direct manipulation is introduced to describe interfaces exhibiting the following characteristics[3]: i) Continuous visibility of objects of interest. For example, an object stays where it is until the operator moves it. ii) Rapid, incremental reversible operations whose impact is immediately visible. iii) Pointing, selecting and dragging techniques replacing the need to type complex syntax, such as command lines like MS-DOS or shell interfaces.

The graphical user interface(GUI) is now firmly established as the preferred user interface for users in most situations. It was first successfully commercialized by Apple with the Macintosh computer in the early eighties and has since become an integral part of every modern operating system for personal computers and graphics workstations. A number of studies have showed that the style of GUIs can enhance the productivity of HCI. People can easily express their preference through the GUI interfaces. As the prices of hardware decrease and the capabilities increase, it is affordable for us to introduce 3D GUIs into computer interfaces. 3D GUIs are three-dimensional graphics interfaces that utilize 3D graphics and offer significant potential for improvement over today's 2D GUI. Notable features of these 3D graphical models allow users to navigate through and interact with the objects in the Virtual Environment, typically using natural interaction styles.

In our research project in the Mechanical Engineering Department of Hong Kong University, we will use CyberGlove, developed by Virtual Technology Inc., as an input device to develop a desktop CAD modeling system for conceptual designers. We elaborate the limitations of the Dataglove and use gestures to support intuitive human-computer interaction. To develop this gesture interface, we emphasize that the conceptual designers are allowed full-freedom to use different kinds of gestures to conduct various geometric shape operations instead of dependence solely on keyboard and 2D mouse. The designers can indicate objects or directions simply by pointing with the hand, and they can look at an object from a different point by just grasping and turning it. The "virtual tools" can be used for shaping, cutting, and joining objects. We employ the 3D GUIs for enhancing the gesture interface. In the context of the objects in the VE, the 3D menu and "virtual hand" float over the objects rather than being part of the scene. Various 3D cursors can be used to select from a menu or to manipulate the objects.

## 2 BACKGROUND

The GUI was developed at Xerox PARC in the late seventies for the Star system[3] and today it has become the predominant design for software applications. The WIMP(Windows, Icon, Menu, Point-and-Click) graphics user interfaces have effectively supplanted command-line user interfaces. This development can be contributed significantly to the fact that the WIMP can provide a metaphor to integrate the visual elements of a graphical user interface into a recognizable and comprehensible framework. In recent years, the new kinds of interaction devices, such as Dataglove and 3D Mouse, are available for users to interact with three or more degrees of freedom. Interface designers are looking forward to evolving the two-dimensional WIMP counterparts into intuitive and natural interfaces that are more powerful and better matched to human sensory capabilities than those dependent solely on keyboard and mouse.

A large number of researches have reported on improving desktop metaphor GUIs[4][5][6]. Work in this area tends to be focussed on experimental systems directed at a particular application or application area. For example, a study by Geoff Leach[3] argued the possibility of 3D GUIs in the near future and discussed the elements of a prototype 3D GUI. The elements contain: i) a 3D window manager, performs the task of window management in response to user input; ii) 3D cursors, provide a metaphor framework for comprehension of the different behaviors. For example, the arrow cursor can be used for selecting objects, as equivalent of the 2D arrow in 2D GUI. The bar magnet cursor is used to select groups of objects within a certain distance of the cursor; iii) a 3D device, such as the Spaceball, controls 3D cursors with six degrees-of-freedom.

At Apple, a 3D MCF(meta content formats) browser, called HotSauce[4], has been developed to allow the user to explore a hierarchical(branching) structure in 3D space. Another example is the Information Visualizer, developed at Xerox PARC. It explores a 3D user interface paradigm suitable for applications to manipulate larger amounts of information[5].

Researchers in IBM Almaden Research Center[7] presented two experiments to argue the graphical means in the visual interface for directing user's attention. The studies have indicated that the image technique can be used to draw user's attention to the important information. For example, the darkening techniques can provide a wide range of masking levels that can produce strong pre-attentive visual cues to pop-out the targets. Darkening creates the effect of a spotlight on a darkened stage.

Because many interface designers rely on their intuition and common sense to design their system interface, these uniformed and untested intuition and senses may cause the interface to be unfriendly for the end users[8]. The aspect of user specification is also concerned with describing the structure of the interface between an interactive system and its users[9]. The goal of the specification research is to establish an effective and usable interface guidance by which the behaviors of our user interfaces can be completely and accurately defined.

Gesture interface also fits in with the intuitive interfaces. It allows the users to freely use gestures for representing various commands. The definition of the gesture is described by Kurtenbach and

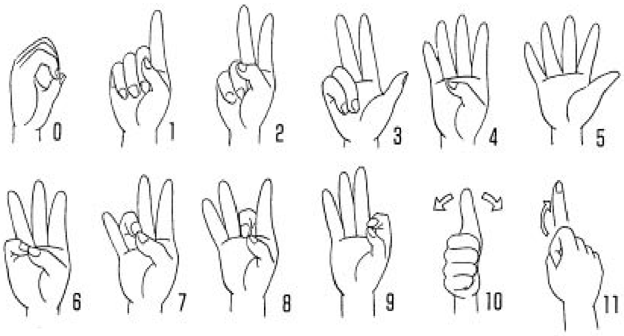



Figure 1: Gestures representing number

Hulteen[10] as*: "A gesture is a motion of the body that contains information. Waving goodbye is a gesture. Pressing a key on a keyboard is not a gesture because the motion of a finger on its way to hitting a key is neither observed nor significant. All that matters is which key was pressed."* Therefore, not all information conveyed by the body movement is considered a gesture. A gesture must be *"a movement of one's body that conveys meaning to oneself or to a partner in communication ... That partner can be a human or a computer"*[11]. For the CAD modeling system, the gesture should contribute to the final shape creation.

When we look into the taxonomies of the gestures described in the literatures we see many different classifications [11][12][13]. These differences are also visible in the type of gesture computer interfaces. For example, Caroline Hummels [11] classifies three types of computer gestures: gestures of pre-defined symbolic commands, gesticulation gestures, and act gestures. The gestures of pre-defined symbolic commands are based on technical constraints instead of human possibilities and needs. For example, the interface designer can pre-define gestures to guide the users for various actions. The gestures can be easily recognized. Since the gestures do not result from the daily life, they may be considered as another kind of some learned commands. The gesticulation gestures are based on the daily communication of people. However, this kind of gesture has little influence on the computer interface because the users only want to interact with the computer generated world instead of the real world. It is also very difficult for the systems to have an ability of accurately recognizing full, dynamic hand gestures due to the problems existing in the algorithms necessary for matching time-space curves. The act gestures are also called imitation gestures. The gestures can be divided in: descriptive gestures, manipulative gestures, utilization gestures and indicative gestures. The descriptive gestures are used to form a description of the object and manipulative gestures are used for manipulation of the object. The utilization and indicative gestures are used respectively to use the object and to select the object. Since the gesture interface only simulates the way that the user acts in the VE as that in the real world, it is suitable for intuitive computer interface design.

Alan Wexelblat[12] argued that the gestures can be characterized as public gestures(for others) and private gestures(for self).

# 3 Architectural Models of the Interface

## 3.1 Gesture Models

**Hand tracking system:** In order to effectively input three-dimensional gestures, we use CyberGlove that looks like a normal glove, fitting over the hand and fingers, as described in[14]. CyberGlove is a low-profile, lightweight glove with flexible sensors which measure the position and movement of the fingers and wrist. The sensor model features two bend sensors on each finger, four abduction sensors, plus sensors measuring thumb crossover, palm arch, wrist flexion and wrist abduction. The hand geometry model can be found in [14]. Polhemus 6 DOF (degrees of freedom) tracking sensors, mounted on the glove wristband, measure the position and orientation of the forearm in space.

**Gesture taxonomy:** The VR technology can offer a new possibility for us to use 3D GUIs and gesture technology to design an intuitive interface for the users. However, there is practically no design advice on how to construct usable 3D GUIs and gesture interfaces. As a first step in creating our interface design guidance, we will employ Caroline Hummels' classification to define our gesture behavior. The gestures used in our system are considered as pre-defined symbolic commands and act gestures that

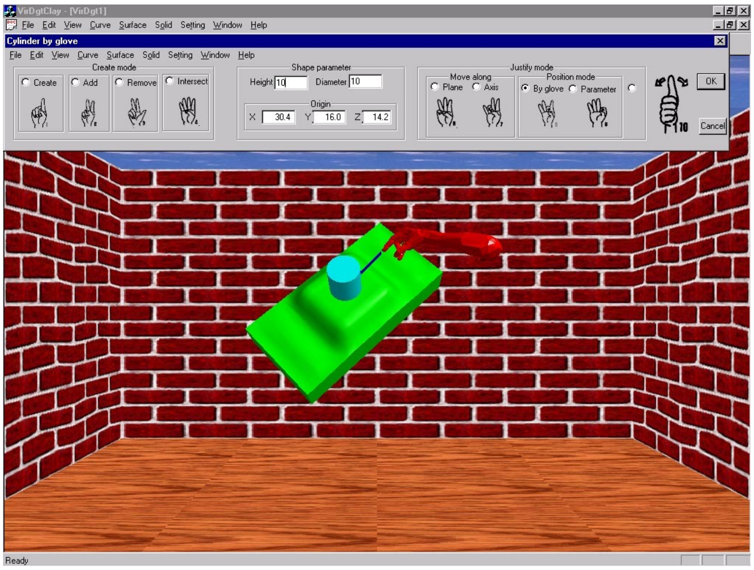



Figure 2: Gesture based dialog box

guide the users for various actions. The gestures are defined by very simple symbols, hence they are easily recognized; the examples are shown in figure 1. The gestures can be divided in: the indicative gesture and manipulative gesture. The indicative gestures are used for creating and choosing menu items or icons. The manipulative gestures are used to guide the users to define, select, and manipulate the objects. The commands defined by the gestures are interpreted by the icons in the menu item or in the dialog box, as shown in figure 2.

**Gesture generation**: Gestures can not be recognized by itself. For example, when we watch a videotape of someone gesturing with the sound turned off, the same gesture may have different explanations by different people and much of the information can be lost. In order to capture the meaning of the gesture, gesture generation has to be broken into two phases: analysis and recognition. Analysis is the process of converting the movements of the hands into a computational representation. Recognition is a context-dependent process. In order to real-time produce an accurate gesture model from the limited data, we will compute the distance of the specified points and then make them satisfy various conditions. We will discuss the computational method for producing a gesture to represent number "3" to show the process of creating a gesture.

Suppose $P_1(x_1, y_1, z_1)$ and $P_2(x_2, y_2, z_2)$ are two points on the glove, then the distance between them can be calculated as:

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}$$

To create a gesture for representing number "3", the following conditions must be satisfied: i) The distance between the distal point in the ring finger and the point on the metacarpus is less than a specified deviation. ii) The distance between the distal point in the little finger and the point on the metacarpus is less than a specified deviation. iii) The distance between the distal point in the thumb and the point on the metacarpus equals to the length of the thumb of the unflexed hand. iv) The distance between the distal point in the index finger and the point on the metacarpus equals to the length of the index finger of the unflexed hand; And, v) The distance between the distal point in the middle finger and the point on the metacarpus equals to the length of the middle finger of the unflexed hand. The above analysis can be used to validate the generated gestures.

## 3.2 User Interface Models

The purpose of our project is to provide the conceptual design engineer with intuitive and comfortable modeling techniques including sketching, feature based modeling, and free-form surface operation. The Sketcher allows the users to freehand sketch and dimension an "outline" of curves. The sketch can then be swept(extruded or revolved) to create a solid or sheet body. Feature based solid modeling lets users create features, such as holes, slots, and grooves, on a model. The interface designed for free-form surface allows the users to define and manipulate free-form objects, such as blends, sweeps, and deformation. The interface elements consist of floating menus, 3D widgets, and the gesture interface.

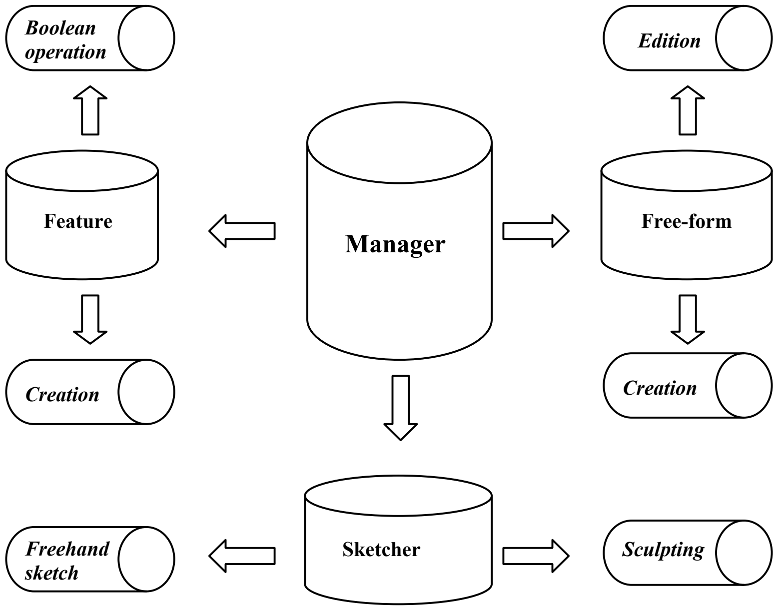



Figure 3: The interface hierarchical tree

**Floating menus:** Because the menu is well accepted in the 2D community, we will still employ the menu as one of the interface elements to provide a large amount of command choices. The floating menu displays two-dimensional menus in the three-dimensional world of VR. These menus are either text-based, describing the available choices with words, or graphically based, using icons to convey the available choices. The floating menus include: feature based menu, sketcher menu, and free-form menu.

**3D widgets:** 3D widgets are objects in the virtual world that present an intuitive, direct manipulation interface to the user. Using 3D widgets as user interface has many advantages. For example, the action of the "virtual hand", displayed in the virtual environment, directly conveys the action of the "real hand" in the "real world". Designing 3D widgets, however, is very hard. People are familiar with 2D widgets, such as 2D cursors. We can change their shape and introduce them into our 3D environment. These 3D cursors, floating over the objects rather than being part of the scene, are controlled by a 3D (six-or-more-degree-of-freedom input) device. In our system, we employ two cursors for interaction: the 3D arrow cursor and the laser beam cursor. The 3D arrow cursor is the 3D equivalent of the 2D cursor and is used for the selection of menu items. The laser cursor is used for the selection of the objects. It sends a "light ray" from the index finger. As the objects fall in the path of the light ray, the laser beam displays to give the user feedback with the object. Unlike the general 2D cursors, the laser cursor may select the object from a distance. When a particular object of interest is highlighted, it may be selected by a specified gesture.

**Gesture**: Gestures enables a user to give commands by making certain hand-gestures. The one-handed input gestures are used to provide a command context including menu operations, rigid solid manipulation and free-form surface definition and modification. Each of the gesture recognition systems works sequentially, receiving data sampled from a DataGlove device, analyzing these data, and sending them to an interpreter, that gives to user's gestures some meaning within an application context. The context hierarchical tree of the gesture interfaces consists of two aspects: i) the floating-menus-driven interfaces, which are used to cascade the modeling menu to appear; ii) task-based-dialog interfaces, within which the gestures icons are embodied to interpret various shape operations. The hierarchy diagram is shown in figure 3. Since the meaning of a gesture is previously associated with a command in the application context, the interpreter module allows the users to use the same gesture to guide different commands. For example, the gesture "1" can be used to drive a feature based modeling menu or to guide the user to make an "adding operation" in the task-based-dialog interfaces, as shown in figure 2 and figure 4.

The main characteristic of the gesture driven interface is that the manipulation commands are pre-defined for a given application in terms of the tasks a user is permitted to perform. It is very easy to define our own set of gestures and to train the system users to

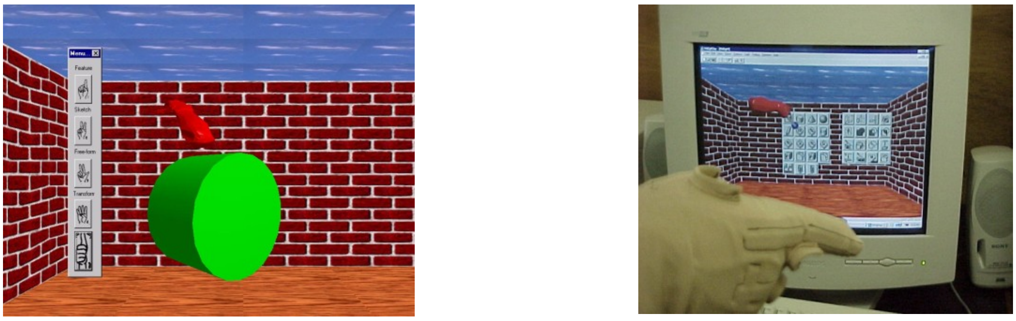


Figure 4: 3D Menu driven dialog box and 3D menu

explicitly recognize only gestures from this set. Our approach has the following advantages: i) the gestures do not need to be physically performed for the teaching procedure; ii) gestures can be easily defined by the system interface designers and used by other peoples.

## 4 IMPLEMENTATION

We have built a desktop Virtual Reality(VR) based modeling environment. The system is written in Microsoft Visual C++ and object-oriented methodology is used for the system design. It currently runs on a Windows NT workstation. The OpenGL graphics libraries are used for the construction of the object viewer that provides the designer with feedback on actions in the virtual environment. The VirtualHand system, developed by Virtual Technology, is used to track and recognize the hand position and the finger bending sensing. We use Visual C++ for the system design because Visual C++ is the standard development environment for both Windows and Windows NT platforms. Its Microsoft Foundation Class (MFC) library provides an architecture that reduces the complexity of our application and the time required to program. The system can allow the conceptual designer to use his/her hand to manipulate an object in computer generated environment in a natural, conversational manner through a gesture and 3D GUI based interface.

Initially the system displays a 3D design studio, and prompts the user to select one of the three modeling methods by specifying a gesture. For example, when the user issues a gesture based command to choose a feature based modeling method, a 3D menu will be popped in the 3D design studio.

**Feature based modeling method:** the method consists of two parts: feature creation and feature operation. For example, if the user wants to define a cylinder, he/she can use a 3D cursor to select a cylinder icon inside the feature based 3D menu. When the menu is selected, a gesture based dialog box is displayed to guide the user to define a cylinder feature by specifying an orientation, size, and location. The user can use a gesture to issue a command to allow the defining feature to do Boolean operations of union, subtraction, and intersection with the previously existing object. The union operation lets the user combine the volume of the defining feature with the existing body. The subtraction option lets the user subtract the defining feature from the existing body. This operation leaves empty space where the subtracted body existed. The intersection option lets the user create a body which contains the volume shared by the defining feature and the existing body.

**Sketcher:** The sketcher is a tool that lets the user create two-dimensional representations of profiles associated with an existing object. The user can create a rough outline of curves by the Virtual hand to capture his/her design intent. In most cases, a sketch is extruded or revolved to create a solid body or sheet body. Then, after you have changed the sketch, the associated body is evaluated accordingly.

**Free-form modeling method:** The user can conduct a gesture to get this option on the 3D menu. It lets him/her create free form features, a solid body or a surface. The free-form modeling method provides the user many tools to create free-form features, such as Swept, Through Curves, and Through Supporting Poles. The tool of Swept lets the user define a swept surface or solid as the shape swept out by a curve outline moving in a prescribed manner along a path in space. The moving curve outline is referred to as the *section profile*. The path is referred to as the *guide path*, because it guides the motion. The Through Curves tool option lets the user create a surface or body through a collection of curve outlines in one direction. The curve outlines are sketched by Virtual Hand on the referred planes. The Through Supporting Poles tool lets the user define a rectangular array of supporting poles through which the surface will pass. The surface interpolates each end point of the pole. Using this option, he/she has very good control over the surface in the sense that it always passes through the end points of the supporting poles that he/she has created.

The free-form modeling method also provides many tools for the user to edit a free-form surface. For example, it allows him or her to edit creation parameters in the above creation tools. In some of the edit options, he/she is given the opportunity to edit the surface through a so-called direct manipulation mode. For example, the surface feature based deformation tool allows the user to directly add a predefined surface feature to the surface.

## 5 DISCUSSION AND CONCLUSION

VR techniques have provided many interactive modes that are more powerful and better matched to human sensory capabilities than those dependent solely on keyboard and mouse. As the computer power and VR input device are dramatically enhanced in the near future, the desktop computer environment will be enriched. The 3D GUI and gestures can provide good driving applications for developing post-WIMP UIs where the keyboard and mice are absent. These techniques have the following advantages:

i) 3D interaction widgets can be controlled by VR input devices with three or more degrees of freedom. They are a natural evolution from their two-dimensional WIMP counterparts and can decrease the cognitive distance between widgets and the tasks.

ii) The 3D GUIs allow interface designers to easily express a preference for the interface. These 3D graphical models allow the users to interact with the computer by using more natural styles, hence enhance productivity and reduce the user's workload.

iii) The gesture based interface supports the perceptual-motor skills of the users. It allows the users full freedom in their gestures to interact with the computer.

It is feasible for us to create an intuitive human-computer interface for the CAD modeling systems by combining limitless gestures and 3D GUIs. The interface gives the conceptual designers a free hand in their approach and uses the gestures to create a conceptual design in a rapid way. The gesture recognition engines in our system can capture the gesture in a real time and offer an intuitive interpretation module that takes into account the gesture based dialog box for understanding the meaning of the gesture.

However, to design a "natural gesture" based interface is still a difficult task in the current technology. The problems are hard in the sense that there are no easy answers to them. We require exhaustive investigations and understanding of the fundamental difficulties that exist in the field. The following difficulties emerge from our own frustrations in trying to do good research and system design.

i) The guidance on how to design usable VEs are hard to find, even though significant usability problems have been reported. Our future work involves adapting the design properties into guidelines for design and checklist questions in evaluation methods.

ii) There is a fundamental tension between continuous and discrete gestures. The true human gesture is continuous. However, introducing the continuous gesture into our system interface is a very difficult task in the sense that it is almost impossible. One reason is that the rate for dynamically recognizing the continuous gestures is much lower than that of the discrete gestures. Another reason is that the continuous gestures also require additional information for the action. For example, when an object is held in a hand, an additional command is required to release the object from the hand. The problem with discrete gestures causes the user to learn another kind of language, and hence makes the system out of the naturalness. Therefore, the combining of discrete gestures with additional commands must make the users benefit from the gesture interface.

iii) Lack of tactile and touch information gives a mismatch with the perceptive feedback. The user needs intuitive force and tactile feedback when he/she gripes and cuts the object. It may cause usability problems in VE: getting a suitable viewing angle, and losing whereabouts after getting too close to objects.

iv) There is latency in the user-computer control loop. The latency is the delay between the time the user moves his/her hand and the time the modified image is displayed. This problem yields a severe problem when dynamical shape operation occurs.

Since gestures are a natural way of expressing person-to-person communication, they will play a very important role in human-computer interaction (HCI). They can be used to describe the shape and size of the objects and how to manipulate the objects when a command context is known. Despite the natural capability of interaction through gestures, gesture interfaces will stimulate a new area of research and provide more effective interactive modes for HCI.

## REFERENCES

[1] Morten Fjeld, Martin Bichsel & Matthias Rauterberg, *BUILD-IT: An Intuitive Design Tool Based on Direct Object Manipulation*, Proceedings of International Gesture Workshop, Biele-feld, Germany, pp. 297-308, September 1997

[2] Zheng J. M., Chan, K. W., Gibson, I., *Virtual Reality: a real world review on a somewhat touchy subject*, IEEE Potentials, pp. 20-23, April/May 1998

[3] Geoff Leach, Ghassan Al-Qaimari, Mark Grieve, Noel Jinks, Cameron McKay, *Elements of a three-dimensional Graphical User Interface*, Human-Computer Interaction INTERACT'97, Proceedings of IFIP TC13 International Conference on Human-computer Interaction, Sydney, Australia, pp 69-76, July 14-18, 1997

[4] Apple(1998),Hotsauce, http://developer.apple.com/mkt/ informed/appledirections/dec96/ hotsauce.html

[5] Robertson, G. R., Card, S. K., and Mackinlay, J. D., *Information Visualization Using 3D Interactive Animation*, Communications of the ACM, 36(4):57-71

[6] Perlin, K. and Fox, d., *Pad: An Alternative Approach to the Computer Interface*, SIGGRAPH Computer Graphics, pp 57-64, 1993

[7] Shumin Zhai, Julie wright Ted selker sabra-Anne Kelin, *Graphical Means of Directing Users' Attention in the Visual Interface*, Human-Computer Interaction INTERACT'97, Proceedings of IFIP TC13 International Conference on Human-computer Interaction, Sydney, Australia, pp 59-66, July 14-18, 1997

[8] Poupyre, I., Weghorst, S., Billinghurst, M., and Ichikawa, W., *Egocentric Object Manipulation in Virtual Environments: Empirical Evaluation of Interaction Techniques,* EUROGRAPHICS'98 Computer Graphics Forum, Vol. 17, No. 3, C41-C52, August 30- September 4, 1998

[9] Paul Chesson, *Towards a Method for User Interface Specification*, Human-Computer Interaction INTERACT'97, Proceedings of IFIP TC13 International Conference on Human-computer Interaction, Sydney, Australia, pp 620-622, July 14-18, 1997

[10] Kurtenbach, G., Hulteen, E. A., *Gestures in Human-Computer Communication*, In B. Laurel(Ed), The art of Human Computer Interface Design, Addison-Wesley,1990

[11] Caroline Hummels, Gerda Smets and Kess Overbecke, *An intuitive Two-Hand Interface for Computer Supported Product Design*, Proceedings of International Gesture Workshop, Bielefeld, Germany, pp. 197-208, September 1997

[12] Alan Wexelblat, *Research Challenges in Gesture: Open Issues and Unsolved Problems*, Proceedings of International Gesture Workshop, Bielefeld, Germany, pp. 1-12, September 1997

[13] Axel Kramer, *Classifying Two Dimensional Gestures in Interactive Systems*, Proceedings of International Gesture Workshop, Bielefeld, Germany, pp. 37-47, September 1997

[14] Http://www.thevrsource.com/

[15] Zheng, J. M., Chan, K. W., Gibson, I., *A New Approach for Direct Manipulation of Free-form Curves*, the international journal of the Eurographics Association: Computer Graphics Forum, EUROGRAPHICS'98, Vol. 17, No. 3, C327-C335, August 30- September 4, 1998

# Twente Workshops on Language Technology

The TWLT workshops are organised by the PARLEVINK project of the University of Twente. The first workshop was held in Enschede, the Netherlands on March 22, 1991. The workshop was attended by about 40 participants. The contents of the proceedings are given below.

**Proceedings Twente Workshop on Language Technology 1 (TWLT 1)**
*Tomita's Algorithm: Extensions and Applications*
Eds. R. Heemels, A. Nijholt & K. Sikkel, 103 pages.

Preface and Contents
**A. Nijholt** *(University of Twente, Enschede)*. (Generalised) LR Parsing: From Knuth to Tomita.
**R. Leermakers** *(Philips Research Labs, Eindhoven)*. Recursive Ascent Parsing.
**H. Harkema & M. Tomita** *(University of Twente, Enschede & Carnegie Mellon University, Pittsburgh)*. A Parsing Algorithm for Non-Deterministic Context-Sensitive Languages.
**G.J. van der Steen** *(Vleermuis Software Research, Utrecht)*. Unrestricted On-Line Parsing and Transduction with Graph Structured Stacks.
**J. Rekers & W. Koorn** *(CWI, Amsterdam & University of Amsterdam, Amsterdam)*. Substring Parsing for Arbitrary Context-Free Grammars.
**T. Vosse** *(NICI, Nijmegen)*. Detection and Correction of Morpho-Syntactic Errors in Shift-Reduce Parsing.
**R. Heemels** *(Océ Nederland, Venlo)*. Tomita's Algorithm in Practical Applications.
**M. Lankhorst** *(University of Twente, Enschede)*. An Empirical Comparison of Generalised LR Tables.
**K. Sikkel** *(University of Twente, Enschede)*. Bottom-Up Parallellization of Tomita's Algorithm.

The second workshop in the series (TWLT 2) has been held on November 20, 1991. The workshop was attended by more than 70 researchers from industry and university. The contents of the proceedings are given below.

**Proceedings Twente Workshop on Language Technology 2 (TWLT 2)**
*Linguistic Engineering: Tools and Products.*
Eds. H.J. op den Akker, A. Nijholt & W. ter Stal, 115 pages.

Preface and Contents
**A. Nijholt** *(University of Twente, Enschede)*. Linguistic Engineering: A Survey.
**B. van Bakel** *(University of Nijmegen, Nijmegen)*. Semantic Analysis of Chemical Texts.
**G.J. van der Steen & A.J. Dijenborgh** *(Vleermuis Software Research, Utrecht)*. Lingware: The Translation Tools of the Future.
**T. Vosse** *(NICI, Nijmegen)*. Detecting and Correcting Morpho-syntactic Errors in Real Texts.
**C. Barkey** *(TNO/ITI, Delft)*. Indexing Large Quantities of Documents Using Computational Linguistics.
**A. van Rijn** *(CIAD/Delft University of Technology, Delft)*. A Natural Language Interface for a Flexible Assembly Cell.
**J. Honig** *(Delft University of Technology, Delft)*. Using Deltra in Natural Language Front-ends.
**J. Odijk** *(Philips Research Labs, Eindhoven)*. The Automatic Translation System ROSETTA3.
**D. van den Akker** *(IBM Research, Amsterdam)*. Language Technology at IBM Nederland.
**M.-J. Nederhof, C.H.A. Koster, C. Dekkers & A. van Zwol** *(University of Nijmegen, Nijmegen)*. The Grammar Workbench: A First Step Toward Lingware Engineering.

The third workshop in the series (TWLT 3) was held on May 12 and 13, 1992. Contrary to the previous workshops it had an international character with eighty participants from the U.S.A., India, Great Britain, Ireland, Italy, Germany, France, Belgium and the Netherlands. The proceedings were available at the workshop. The contents of the proceedings are given below.

**Proceedings Twente Workshop on Language Technology 3 (TWLT 3)**
*Connectionism and Natural Language Processing*
Eds. M.F.J. Drossaers & A. Nijholt, 142 pages.


Preface and Contents

**L.P.J. Veelenturf** *(University of Twente, Enschede)*. Representation of Spoken Words in a Self-Organising Neural Net.

**P. Wittenburg & U. H. Frauenfelder***(Max-Planck Institute, Nijmegen)*. Modelling the Human Mental Lexicon with Self-Organising Feature Maps.

**A.J.M.M. Weijters & J. Thole** *(University of Limburg, Maastricht)*. Speech Synthesis with Artificial Neural Networks.

**W. Daelemans & A. van den Bosch** *(Tilburg University, Tilburg)*. Generalisation Performance of Back Propagation Learning on a Syllabification Task.

**E.-J. van der Linden & W. Kraaij** *(Tilburg University, Tilburg)*. Representation of Idioms in Connectionist Models.

**J.C. Scholtes** *(University of Amsterdam, Amsterdam)*. Neural Data Oriented Parsing.

**E.F. Tjong Kim Sang** *(University of Groningen, Groningen)*. A connectionist Representation for Phrase Structures.

**M.F.J. Drossaers** *(University of Twente, Enschede)*. Hopfield Models as Neural-Network Acceptors.

**P. Wyard** *(British Telecom, Ipswich)*. A Single Layer Higher Order Neural Net and its Application to Grammar Recognition.

**N.E. Sharkey & A.J.C. Sharkey** *(University of Exeter, Exeter)*. A Modular Design for Connectionist Parsing.

**R. Reilly** *(University College, Dublin)*. An Exploration of Clause Boundary Effects in SRN Representations.

**S.M. Lucas** *(University of Essex, Colchester)*. Syntactic Neural Networks for Natural Language Processing.

**R. Miikkulainen** *(University of Texas, Austin)*. DISCERN: A Distributed Neural Network Model of Script Processing and Memory.


The fourth workshop in the series has been held on September 23, 1992. The theme of this workshop was "Pragmatics in Language Technology". Its aim was to bring together the several approaches to this subject: philosophical, linguistic and logic. The workshop was visited by more than 50 researchers in these fields, together with several computer scientists. The contents of the proceedings are given below.

**Proceedings Twente Workshop on Language Technology 4 (TWLT 4)**
*Pragmatics in Language Technology*
Eds. D. Nauta, A. Nijholt & J. Schaake, 114 pages.


Preface and Contents

**D. Nauta, A. Nijholt & J. Schaake** *(University of Twente, Enschede)*. Pragmatics in Language technology: Introduction.

**Part 1: Pragmatics and Semiotics**

**J. van der Lubbe & D. Nauta** *(Delft University of Technology & University of Twente, Enschede)*. Semiotics, Pragmatism, and Expert Systems.

**F. Vandamme** *(Ghent)*. Semiotics, Epistemology, and Human Action.

**H. de Jong & W. Werner** *(University of Twente, Enschede)*. Separation of Powers and Semiotic Processes.

**Part 2: Functional Approach in Linguistics**

**C. de Groot** *(University of Amsterdam)*. Pragmatics in Functional Grammar.
**E. Steiner** *(University of Saarland, Saarbrücken).* Systemic Functional Grammar.
**R. Bartsch** *(University of Amsterdam)*. Concept Formation on the Basis of Utterances in Situations.
**Part 3: Logic of Belief, Utterance, and Intention**
**J. Ginzburg** *(University of Edinburgh)*. Enriching Answerhood and Truth: Questions within Situation Semantics.
**J. Schaake** *(University of Twente, Enschede).* The Logic of Peirce's Existential Graphs.
**H. Bunt** *(Tilburg University)*. Belief Contexts in Human-Computer Dialogue.

---

The fifth workshop in the series took place on 3 and 4 June 1993. It was devoted to the topic "Natural Language Interfaces". The aim was to provide an international platform for commerce, technology and science to present the advances and current state of the art in this area of research.

---

**Proceedings Twente Workshop on Language Technology 5 (TWLT 5)**
*Natural Language Interfaces*
Eds. F.M.G. de Jong & A. Nijholt, 124 pages.

Preface and Contents
**F.M.G. de Jong & A. Nijholt** *(University of Twente)*. Natural Language Interfaces: Introduction.
**R. Scha** *(University of Amsterdam)*. Understanding Media: Language vs. Graphics.
**L. Boves** *(University of Nijmegen)*. Spoken Language Interfaces.
**J. Nerbonne** *(University of Groningen)*. NL Interfaces and the Turing Test.
**K. Simons** *(Digimaster, Amstelveen)*. "Natural Language": A Working System.
**P. Horsman** *(Dutch National Archives, The Hague)*. Accessibility of Archival Documents.
**W. Sijtsma & O. Zweekhorst** *(ITK, Tilburg)*. Comparison and Review of Commercial Natural Language Interfaces.
**J. Schaake** *(University of Twente)*. The Reactive Dialogue Model: Integration of Syntax, Semantics, and Pragmatics in a Functional Design.
**D. Speelman** *(University of Leuven)*. A Natural Language Interface that Uses Generalised Quantifiers.
**R.-J. Beun** *(IPO, Eindhoven)*. The DENK Program: Modeling Pragmatics in Natural Language Interfaces.
**W. Menzel** *(University of Hamburg)*. Title.
**C. Huls & E. Bos** *(NICI, Nijmegen)*. EDWARD: A Multimodal Interface.
**G. Neumann** *(University of Saarbrücken)*. Design Principles of the DISCO system.
**O. Stock & C. Strapparava** *(IRST, Trento)*. NL-Based Interaction in a Multimodal Environment.

---

The sixth workshop in the series took place on 16 and 17 December 1993. It was devoted to the topic "Natural Language Parsing". The aim was to provide an international platform for technology and science to present the advances and current state of the art in this area of research, in particular research that aims at analysing real-world text and real-world speech and keyboard input.

---

**Proceedings Twente Workshop on Language Technology 6 (TWLT 6)**
*Natural Language Parsing: Methods and Formalisms*
Eds. K. Sikkel & A. Nijholt, 190 pages.

Preface and Contents
**A. Nijholt** (*University of Twente*). Natural Language Parsing: An Introduction.
**V. Manca** (*University of Pisa*). Typology and Logical Structure of Natural Languages.
**R. Bod** (*University of Amsterdam*). Data Oriented Parsing as a General Framework for Stochastic Language Processing.

**M. Stefanova & W. ter Stal** (*University of Sofia / University of Twente*). A Comparison of ALE and PATR: Practical Experiences.
**J.P.M. de Vreught** (*University of Delft*). A Practical Comparison between Parallel Tabular Recognizers.
**M. Verlinden** (*University of Twente*). Head-Corner Parsing of Unification Grammars: A Case Study.
**M.-J. Nederhof** (*University of Nijmegen*). A Multi-Disciplinary Approach to a Parsing Algorithm.
**Th. Stürmer** (*University of Saarbrücken*). Semantic-Oriented Chart Parsing with Defaults.
**G. Satta** (*University of Venice*). The Parsing Problem for Tree-Adjoining Grammars.
**F. Barthélemy** (*University of Lisbon*). A Single Formalism for a Wide Range of Parsers for DCGs.
**E. Csuhaj-Varjú and R. Abo-Alez** (*Hungarian Academy of Sciences, Budapest*). Multi-Agent Systems in Natural Language Processing.
**C. Cremers** (*University of Leiden*). Coordination as a Parsing Problem.
**M. Wirén** (*University of Saarbrücken*). Bounded Incremental Parsing.
**V. Kubon and M. Platek** (*Charles University, Prague*). Robust Parsing and Grammar Checking of Free Word Order Languages.
**V. Srinivasan** (*University of Mainz*). Punctuation and Parsing of Real-World Texts.
**T.G. Vosse** (*University of Leiden*). Robust GLR Parsing for Grammar-Based Spelling Correction.

---

The seventh workshop in the series took place on 15 and 16 june 1994. It was devoted to the topic "Computer-Assisted Language Learning" (CALL). The aim was to present both the state of the art in CALL and the new perspectives in the research and development of software that is meant to be used in a language curriculum. By the mix of themes addressed in the papers and demonstrations, we hoped to bring about the exchange of ideas between people of various backgrounds.

---

**Proceedings Twente Workshop on Language Technology 7 (TWLT 7)**
*Computer-Assisted Language Learning*
Eds. L. Appelo, F.M.G. de Jong, 133 pages.

Preface and Contents
**L.Appelo, F.M.G. de Jong** (*IPO / University of Twente*). Computer-Assisted Language Learning: Prolegomena
**M. van Bodegom** (*Eurolinguist Language House, Nijmegen, The Netherlands*). Eurolinguist test: An adaptive testing system.
**B. Cartigny** (*Escape, Tilburg, The Netherlands*). Discatex CD-ROM XA.
**H.Altay Guvenir, K. Oflazer** (*Bilkent University, Ankara*). Using a Corpus for Teaching Turkish Morphology.
**H. Hamburger** (*GMU, Washington, USA*). Viewpoint Abstraction: a Key to Conversational Learning.
**J. Jaspers, G. Kanselaar, W. Kok** (*University of Utrecht, The Netherlands*). Learning English with It's English.
**G. Kempen, A. Dijkstra** (*University of Leiden, The Netherlands*). Towards an integrated system for spelling, grammar and writing instruction.
**F. Kronenberg, A. Krueger, P. Ludewig** (*University of Osnabruek, Germany*). Contextual vocabulary learning with CAVOL.
**S. Lobbe** (*Rotterdam Polytechnic Informatica Centrum, The Netherlands*). Teachers, Students and IT: how to get teachers to integrate IT into the (language) curriculum.
**J. Rous, L. Appelo** (*Institute for Perception Research, Eindhoven, The Netherlands*). APPEAL: Interactive language learning in a multimedia environment.
**B. Salverda** (*SLO, Enschede, The Netherlands*). Developing a Multimedia Course for Learning Dutch as a Second Language.
**C. Schwind** (*Universite de Marseille, France*). Error analysis and explanation in knowledge based language tutoring.
**J. Thompson** (*CTI, Hull, United Kingdom/EUROCALL*). TELL into the mainstream curriculum.
**M. Zock** (*Limsi, Paris, France*). Language in action, or learning a language by whatching how it works.

Description of systems demonstrated:

**APPEAL** (*Institute of Perception Research, Eindhoven*)
**Bonacord, Méli-Mélo, ect.** (*School of European Languages & Cultures, University of Hull*)
**Computer BBS in language instruction** (*English Programs for Internationals, University of South Carolina*)
**Discatext** (*Escape, Tilburg)*
**Error analysis and explanation** (*CNRS, Laboratoire d'Informatique de Marseille*)
**ItalCultura, RumboHispano and IVANA** (*Norwegian Computing Centre for the Humanities, Harald*)
**It's English** (*Department of Educational Sciences, Utrecht University*)
**Multimedia course for learning Dutch** (*SLO, Enschede*)
**Part of CATT** (*Department of Computer Engineering and Information Science, Bilkent University, Ankara*)
**PROMISE** (*Institut für Semantische Informationsverarbeitung, Univerität Osnabrück*)
**Speech-Melody trainer** (*Institute of Perception Research, Eindhoven*)
**The Rosetta Stone** (*Eurolinguist Language House Nijmegen*)
**Verbarium and Substantarium** (*SOS Nijmegen*)
**WOORD** (*Applied Linguistics Unit, Delft University of Technology*)
**FLUENT-II** (George Mason University, Washington)

---

The eighth workshop in the series took place on 1 and 2 December 1994. It was devoted to speech, the integration of speech and natural language processing, and the application of this integration in natural language interfaces. The program emphasized research of interest for the themes in the framework of the Dutch NWO programme on Speech and Natural Language that started in 1994.

---

**Proceedings Twente Workshop on Language Technology 8 (TWLT 8)**
*Speech and Language Engineering*
Eds. L. Boves, A. Nijholt, 176 pages.

Preface and Contents

**Chr. Dugast** (*Philips, Aachen, Germany*). The North American Business News Task: Speaker Independent, Unlimited Vocabulary Article Dictation

**P. van Alphen, C. in't Veld & W. Schelvis** (*PTT Research, Leidschendam, The Netherlands*). Analysis of the Dutch Polyphone Corpus.

**H.J.M. Steenken & D.A. van Leeuwen** (*TNO Human factors Research, Soesterberg, The Netherlands*). Assessment of Speech Recognition Systems.

**J.M. McQueen** (*Max Planck Institute, Nijmegen, The Netherlands*). The Role of Prosody in Human Speech Recognition.

**L. ten Bosch** (*IPO, Eindhoven, the Netherlands*). The Potential Role of Prosody in Automatic Speech Recognition.

**P. Baggia, E. Gerbino, E. Giachin & C. Rullent** (*CSELT, Torino, Italy*). Spontaneous Speech Phenomena in Naive-User Interactions.

**M.F.J. Drossaers & D. Dokter** (*University of Twente, Enschede, the Netherlands*). Simple Speech Recognition with Little Linguistic Creatures.

**H. Helbig & A. Mertens** (*FernUniversität Hagen, Germany*). Word Agent Based Natural Language Processing.

**Geunbae Lee et al.** (*Pohang University, Hyoja-Dong, Pohang, Korea*). Phoneme-Level Speech and natural Language Integration for Agglutinative Languages.

**K. van Deemter, J. Landsbergen, R. Leermakers & J. Odijk** (*IPO, Eindhoven, The Netherlands*). Generation of Spoken Monologues by Means of Templates

**D. Carter & M. Rayner** (*SRI International, Cambridge, UK*). The Speech-Language Interface in the Spoken Language Translator

**H. Weber** (*University of Erlangen, Germany*). Time-synchronous Chart Parsing of Speech Integrating Unification Grammars with Statistics.

**G. Veldhuijzen van Zanten & R. op den Akker** (*University of Twente, Enschede, the Netherlands*). More Efficient Head and Left Corner Parsing of Unification-based Formalisms.

**G.F. van der Hoeven et al.** (*University of Twente, Enschede, the Netherlands*). SCHISMA: A natural Language Accessible Theatre Information and Booking System.
**G. van Noord** (*University of Groningen, the Netherlands*). On the Intersection of Finite Staate Automata and Definite Clause Grammars.
**R. Bod & R. Scha** (*University of Amsterdam, the Netherlands*). Prediction and Disambiguation by Means of Data-Oriented Parsing.

---

The ninth workshop in the series took place on 9 June 1995. It was devoted to empirical methods in the analysis of dialogues, and the use of corpora of dialogues in building dialogue systems. The aim was to discuss the methods of corpus analysis, as well as results of corpus analysis and the application of such results.

---

**Proceedings Twente Workshop on Language Technology 9 (TWLT 9)**
*Corpus-based Approaches to Dialogue Modelling*
Eds. J.A. Andernach, S.P. van de Burgt & G.F. van der Hoeven, 124 pages.

Preface and Contents
**N. Dahlbäck** (*NLP Laboratory, Linköping, Sweden*). Kinds of agents and types of dialogues.
**J.H. Connolly, A.A. Clarke, S.W. Garner & H.K. Palmén** (*Loughborough University of Technology, UK*). Clause-internal structure in spoken dialogue.
**J. Carletta, A. Isard, S. Isard, J. Kowtko, G. Doherty-Sneddon & A. Anderson** (*HCRC, Edinburgh, UK*). The coding of dialogue structure in a corpus.
**J. Alexandersson & N. Reithinger** (*DFKI, Saarbrücken, Germany*). Designing the dialogue component in a speech translation system – a corpus-based approach.
**H. Aust & M. Oerder** (*Philips, Aachen, Germany*). Dialogue control in automatic inquiry systems.
**M. Rats** (*ITK, Tilburg, the Netherlands*). Referring to topics – a corpus-based study.
**H. Dybkjær, L. Dybkjær & N.O. Bernsen** (*Centre for Cognitive Science, Roskilde, Denmark*). Design, formalization and evaluation of spoken language dialogue.
**D.G. Novick & B. Hansen** (*Oregon Graduate Institute of Science and Technology, Portland, USA*). Mutuality strategies for reference in task-oriented dialogue.
**N. Fraser** (*Vocalis Ltd, Cambridge, UK*). Messy data, what can we learn from it?
**J.A. Andernach** (University of Twente, Enschede, the Netherlands). Predicting and interpreting speech acts in a theatre information and booking system.

---

The tenth workshop in the series took place on 6-8 December 1995. This workshop was organized in the framework provided by the Algebraic Methodology and Software Technology movement (AMAST). It focussed on algebraic methods in formal languages, programming langauges and natural languages. Its aim was to bring together those researchers on formal language theory, programming language theory and natural language description theory, that have a common interest in the use of algebraic methods to describe syntactic, semantic and pragmatic properties of language.

---

**Proceedings Twente Workshop on Language Technology 10 (TWLT 10)**
*Algebraic Methods in Language Processing*
Eds. A. Nijholt, G. Scollo and R. Steetskamp, 263 pages

Preface and Contents
**Teodor Rus** (*Iowa City, USA*). Algebraic Processing of Programming Languages.
**Eelco Visser** (*Amsterdam, NL*). Polymorphic Syntax Definition.
**J.C. Ramalho, J.J. Almeida & P.R. Henriques** (*Braga, P*). Algebraic Specification of Documents.

**Teodor Rus & James, S. Jones** (*Iowa City, USA*). Multi-layered Pipeline Parsing from Multi-axiom Grammars.
**Klaas Sikkel** (*Sankt Augustin, D*). Parsing Schemata and Correctness of Parsing Algorithms.
**François Barthélemy** (*Paris, F*). A Generic Tabular Scheme for Parsing.
**Frederic Tendeau** (*INRIA, F*). Parsing Algebraic Power Series Using Dymanic Programming.
**Michael Moortgat** (*Utrecht, NL*). Mutimodal Linguistic Inference.
**R.C. Berwick** (*MIT, USA*). Computational Minimalism: The Convergence of the Minimalistic Syntactic Program and Categorial Grammar.
**Annius V. Groenink** (Amsterdam, NL). A Simple Uniform Semantics for Concatenation-Based Grammar.
**Grzegorz Rozenberg** (*Leiden, NL*). Theory of Texts (abstract only).
**Jan Rekers** (*Leiden, NL*) **& A Schürr** (*Aachen, D*). A Graph Grammar Approach to Graphical Parsing.
**Sándor Horvath** (*Debrecen, H*). Strong Interchangeability and Nonlinearity of Primitive Words.
**Wojciech Buszkowski** (*Poznan, P*). Algebraic Methods in Categorial Grammar.
**Vladimir A. Fomichov** (*Moscow, R*). A Variant of a Universal Metagrammar of Conceptual Structures. Algebraic Systems of Conceptual Syntax.
**Theo M.V. Jansen** (*Amsterdam, NL*). The Method of ROSETTA, Natural Language Translation Using Algebras.
**C.M. Martín-Vide, J. Miquel-Verges & Gh. Paun** (*Tarragona, E*). Contextual Grammars with Depth-First Derivation.
**Pál Dömösi** (*Kussuth University, H*) **& Jürgen Duske** (*University of Hanover, G*). Subword Membership Problem for Linear Indexed Languages.
**C. Rico Perez & J.S. Granda** (*Madrid, E*). Algebraic Methods for Anaphora Resolution.
**Vincenzo Manca** (*Pisa, I*). A Logical Formalism for Intergrammatical Representation.

---

The eleventh workshop in the series took place on 19-21 June 1996. It focussed on the task of dialogue management in natural-language processing systems. The aim was to discuss advances in dialogue management strategies and design methods. During the workshop, there was a separate session concerned with evalutation methods.

---

**Proceedings Twente Workshop on Language Technology 11 (TWLT 11)**
*Dialogue Management in Natural Language Systems*
Eds. S. LuperFoy, A. Nijholt and G. Veldhuijzen van Zanten, 228 pages

Preface and Contents
**David R. Traum** (*Université de Genève, CH*). Conversational Agency: The TRAINS-93 Dialogue Manager.
**Scott McGlashan** (*SICS, SW*).Towards Multimodal Dialogue Management.
**Pierre Nugues, Christophe Godéreaux, Pierre-Olivier and Frédéric Revolta** (*GREYC, F*). A Conversational Agent to Navigate in Virtual Worlds.
**Anne Vilnat** (*LIMSI-CNRS, F*). Which Processes to Manage Human-Machine Dialogue?
**Susann LuperFoy** (*MITRE, USA*).Tutoring versus Training: A Mediating Dialogue Manager for Spoken Language Systems.
**David G. Novick and Stephen Sutton** (*Portland, USA*). Building on Experience: Managing Spoken Interaction through Library Subdialogues.
**Latifa Taleb** (*INRIA, F*). Communicational Deviation in Finalized InformativeDialogue Management.
**Robbert-Jan Beun** (*IPO, NL*). Speech Act Generation in Cooperative Dialogue.
**Gert Veldhuijzen van Zanten** (*IPO, NL*). Pragmatic Interpretation and Dialogue Management in Spoken-Language Systems.
**Joris Hulstijn, René Steetskamp, Hugo ter Doest** (*University of Twente, NL*), **Stan van de Burgt** (*KPN Research, NL*) **and Anton Nijholt** (*University of Twente, NL*). Topics in SCHISMA Dialogues.
**Gavin Churcher, Clive Souter and Eric S. Atwell** (*Leeds University, UK*). Dialogues in Air Traffic Control
**Elisabeth Maier** (*DFKI, D*).Context Construction as Subtask of Dialogue Processing -- the VERBMOBIL Case.
**Anders Baekgaard** (*CPK, DK*). Dialogue Management in a Generic Dialogue System.

**Wayne Ward** (*Carnegie Mellon University, USA*). Dialog Management in the CMU Spoken Language Systems Toolkit.
**Wieland Eckert** (*University of Erlangen, D*). Understanding of Spontaneous Utterances in Human-Machine-Dialog.
**Jan Alexandersson** (*DFKI, D*). Some Ideas for the Automatic Acquisition of Dialogue Structure.
**Kristiina Jokinen** (*Nara Institute of Science and Technology, JP*). Cooperative Response Planning in CDM: Reasoning about Communicative Strategies.
**Elizabeth Hinkelman** (*Kurzweil Applied Science, USA*). Dialogue Grounding for Speech Recognition Systems.
**Jennifer Chu-Carroll** (*University of Delaware, USA*). Response Generation in Collaborative Dialogue Interactions.
**Harry Bunt** (*Tilburg University, NL*). Interaction Management Functions and Context Representation Requirements.
**Peter Wyard and Sandra Williams** (*BT, GB*). Dialogue Management in a Mixed-Initiative, Cooperative, Spoken Language System.
**Rolf Carlson** (*KTH, SW*). The Dialog Component in the Waxholm System.
**Laila Dybkjær, Niels Ole Bernsen and Hans Dybkjaer** (Roskilde University, DK). Evaluation of Spoken Dialogue Systems.
**Vincenzo Manca** (*Pisa, I*). A Logical Formalism for Intergrammatical Representation.

---

TWLT 12 took place on 11-14 September 1996. It focussed on 'computational humor' and in particular on verbal humor. TWLT12 consisted of a symposium (Marvin Minsky, Douglas Hofstadter, John Allen Paulos, Hugo Brandt Corstius, Oliviero Stock and Gerrit Krol as main speakers), an essay contest for computer science students, two panels, a seminar organized by Salvatore Attardo and Wladyslaw Chlopicki and a two-day workshop (Automatic interpretation and Generation of Verbal Humor0 with a mix of invited papers and papers obtained from a Call for Papers.

---

**Proceedings Twente Workshop on Language Technology 12 (TWLT 12)**
*Computational Humor: Automatic Interpretation and Generation of Verbal Humor*
Eds. J. Hulstijn and A. Nijholt, 208 pages

Preface and Contents
**Oliviero Stock** *'Password Swordfish': Verbal Humor in the Interface.*
**Victor Raskin** *Computer Implementation of the General Theory of Verbal Humor.*
**Akira Ito and Osamu Takizawa** *Why do People use Irony? - The Pragmatics of Irony Usage.*
**Akira Utsumi**. *Implicit Display Theory of Verbal Irony: Towards a Computational Model of Irony.*
**Osamu Takizawa, Masuzo Yanagida, Akira Ito and Hitoshi Isahara** On Computational Processing of Rhetorical Expressions - Puns, Ironies and Tautologies.
**Carmen Curcó** *Relevance Theory and Humorous Interpretations.*
**Ephraim Nissan** *From ALIBI to COLOMBUS. The Long March to Self-aware Computational Models of Humor.*
**Salvatore Attardo** *Humor Theory beyond Jokes: The Treatment of Humorous Texts at Large.*
**Bruce Katz** *A Neural Invariant of Humour.*
**E. Judith Weiner** *Why is a Riddle not Like a Metaphor?*
**Tone Veale** *No Laughing Matter: The Cognitive Structure of Humour, Metaphor and Creativity.*
**Tony Veale and Mark Keane** *Bad Vibes: Catastrophes of Goal Activation in the Appreciation of Disparagement Humour and General Poor Taste.*
**Kim Binsted and Graeme Ritchie** *Speculations on Story Pun.*
**Dan Loehr** *An Integration of a Pun Generator with a Natural Language Robot.*
**Cameron Shelley, Toby Donaldson and Kim Parsons** *Humorous Analogy: Modeling 'The Devils Dictionary'.*
**Michal Ephratt** *More on Humor Act: What Sort of Speech Act is the Joke?*

TWLT 13 took place on 13-15 May 1998. It was the follow-up of the Mundial workshop, that took place in München in 1997. Both the Mundial workshop as TWLT13 focussed on the formal semantics and pragmatics of dialogues. In addition to the three-day workshop in Twente, with invited and accepted papers, on 18 May a workshop titled ‘Communication and Attitudes’ was organized at ILLC/University of Amsterdam.

**Proceedings Twente Workshop on Language Technology 13 (TWLT 13)**
*Formal Semantics and Pragmatics of Dialogue (Twendial'98)*
Eds. J. Hulstijn and A. Nijholt, 274 pages

Preface and Contents
**Nicholas Asher** *Varieties of Discourse Structure in Dialogue*
**Jonathan Ginzburg** *Clarifying Utterances*
**Steve Pulman** *The TRINDI Project: Some Preliminary Themes*
**Henk Zeevat** *Contracts in the Common Ground*
**John Barnden** *Uncertain Reasoning About Agents' Beliefs and Reasoning, with special attention to Metaphorical Mental State Reports*
**Thomas Clermont, Marc Pomplun, Elke Prestin and Hannes Rieser** *Eye-movement Research and the Investigation of Dialogue Structure*
**Robin Cooper** *Mixing Situation Theory and Type Theory to Formalize Information States in Dialogue*
**Jean-louis Dessalles** *The Interplay of Desire and Necessity in Dialogue*
**Wieland Eckert** *Automatic Evaluation of Dialogue Systems*
**Jelle Gerbrandy** *Some Remarks on Distributed Knowledge*
**Jeroen Groenendijk** *Questions in Update Semantics*
**Wolfgang Heydrich** *Theory of Mutuality (Syntactic Skeleton)*
**Wolfgang Heydrich, Peter Kühnlein and Hannes Rieser** *A DRT-style Modelling of Agents' Mental States in Discourse*
**Staffan Larsson** *Questions Under Discussion and Dialogue Moves*
**Ian Lewin** *Formal Design, Verification and Simulation of Multi-Modal Dialogues*
**Nicolas Maudet and Fabrice Evrard** *A Generic framework for Dialogue Game Implementation*
**Soo-Jun Park, Keon-Hoe Cha, Won-Kyung Sung, Do Gyu Song, Hyun-A Lee, Jay Duke Park, Dong-In Park and Jörg Höhle** *MALBOT: An Intelligent Dialogue Model using User Modeling*
**Massimo Poesio and David Traum** *Towards an Axiomatization of Dialogue Acts*
**Mieke Rats** *Making DRT Suitable for the Description of Information Exchange in a Dialogue*
**Robert van Rooy** *Modal subordination in Questions*
**Adam Zachary Wyner** *A Discourse Theory of Manner and Factive Adverbial Modification*
**Marc Blasband** *A Simple Semantic Model*

TWLT14 was held on 7-8 December 1998. It focussed on the role of human language technology in the indexing and accessing of written and spoken documents, video material and/or images, and on the role of language technology for cross-language retrieval and information extraction. The workshop consisted of a series of accepted papers.

**Proceedings Twente Workshop on Language Technology 14 (TWLT 14)**
*Language Technology in Multimedia Information Retrieval*
Eds. D. Hiemstra, F.M.G. de Jong and K. Netter, 194 pages

Preface and Contents

**Hans Uszkoreit (DFKI, Saarbrücken)** *Cross-language information retrieval: from naive concepts to realistic applications*
**Paul Buitelaar, Klaus Netter and Feiyu Xu (DFKI, Saarbrücken)** *Integrating Different Strategies for Cross-Language Retrieval in the MIETTA Project*
**Djoerd Hiemstra and Franciska de Jong (University of Twente)** *Cross-language Retrieval in Twenty-One: using one, some or all possible translations?*
**David A. Hull (Xerox Research Center Europe)** *Information Extraction from Bilingual Corpora and its application to Machine-aided Translation*
**Arjen P. de Vries (University of Twente)** *Mirror: Multimedia Query Processing in Extensible Databases*
*Douglas E. Appelt (SRI International) An Overview of Information Extraction Technology and its Application to Information Retrieval*
**Paul E. van der Vet and Bas van Bakel (University of Twente)** *Combining Linguistic and Knowledge-based Engineering for Information Retrieval and Information Extraction*
**Karen Sparck Jones (Cambridge University)** *Information retrieval: how far will* really *simple methods take you?*
**Raymond Flournoy, Hiroshi Masuichi and Stanley Peters (Stanford University and Fuji Xerox Co. Ltd.)** *Cross-Language Information Retrieval: Some Methods and Tools*
**Andrew Salway and Khurshid Ahmad (University of Surrey)** *Talking Pictures: Indexing and Representing Video with Collateral Texts*
**Wim van Bruxvoort (VDA informatiebeheersing)** *Pop-Eye: Using Language Technology in Video Retrieval*
**Istar Buscher (Südwestrundfunk, Baden Baden)** *Going digital at SWR TV-archives: New dimensions of information management professional and public demands*
**Arnold W.M. Smeulders, Theo Gevers and Martin L. Kersten (University of Amsterdam)** *Computer vision and image search engines*
**Kees van Deemter (University of Brighton)** *Retrieving Pictures for Document Generation*
*Steve Renals and Dave Abberly (University of Sheffield) The THISL Spoken Document Retrieval System*
**Wessel Kraaij, Joop van Gent, Rudie Ekkelenkamp and David van Leeuwen (TNO-TPD Delft and TNO-HFRI Soesterberg)** *Phoneme Based Spoken Document Retrieval*
**Jade Goldstein and Jaime Carbonell (Carnegie Mellon University)** *The use of MMR, diversity-based reranking in document reranking and summarization*
**Michael P. Oakes, Chris D. Paice (Lancaster University)** *Evaluation of an automatic abstractic system*
**Danny H. Lie (Carp Technologies, The Netherlands)** *Sumatra: A system for Automatic Summary* **Generation**
**Marten den Uyl, Ed S. Tan, Heimo Müller and Peter Uray (SMR Amsterdam, Vrije Universiteit Amsterdam, Joanneum Research)** *Towards Automatic Indexing and Retrieval of Video Content: the VICAR system*
**Anton Nijholt (University of Twente)** *Access, Exploration and Visualization of Interest Communities: The VMC Case Study (in Progress)*
**Joanne Capstick, Abdel Kader Diagne, Gregor Erbach and Hans Uszkoreit (DFKI, Saarbrücken)** *MULINEX: Multilingual Web Search and Navigation*
**Klaus Netter and Franciska de Jong (DFKI, Saarbrücken and University of Twente)** *OLIVE: speech based video retrieval*
**Franciska de Jong (University of Twente)** *Twenty-One: a baseline for multilingual multimedia*

The proceedings of the workshops can be ordered from Vakgroep SETI, Department of Computer Science, University of Twente, P.O. Box 217, NL-7500 AE Enschede, The Netherlands. E-mail orders are possible: twlt_secr@cs.utwente.nl. Each of the proceedings costs Dfl. 50.