EXPERIENCES WITH THE EVALUATION OF CVE APPLICATIONS

ANTHONY STEED, JOLANDA TROMP

DEPARTMENT OF COMPUTER SCIENCE, UNIVERSITY COLLEGE LONDON

COMMUNICATIONS RESEARCH GROUP, UNIVERSITY OF NOTTINGHAM

a.steed@cs.ucl.ac.uk, jgt@cs.nott.ac.uk

Abstract

We present our experience with a usability evaluation of two prototype CVE applications being developed by the COVEN project. The applications are a teleconferencing system aimed at future business users of CVEs, and a holiday planning environment for the use of the general public. Our concerns in the evaluations were three fold: to refine the design of these applications, to gain further understanding of the component technologies of CVEs, and to reflect upon methodologies for evaluation. We report upon two main threads of investigation: a usability inspection of the applications, and a series of network trials. We discuss some important outstanding issues for the system designer, and suggest several guidelines for the application builder. Finally we reflect upon the methodologies chosen and suggest a focus for future CVE evaluations.

Keywords

Usability Inspection, Network Trials, Evaluation Frameworks, Methodology, HCI

1. INTRODUCTION

The Collaborative Virtual Environments (COVEN) project, is a 4 year European project to design and explore collaborative virtual environment (CVE) technology. Its aim is to demonstrate the feasibility of scaleable CVE worlds through prototype applications in the general area of virtual travel. In addition to the development of CVE fundamental technologies, COVEN is developing applications that demonstrate the utility of CVEs. The main focus of this paper is the evaluation of the first generation of the COVEN application. Our research has identified a significant number of usability issues at system, interaction and application levels, most of which will be applicable to all CVE systems and applications, not just those developed by COVEN.

The technical basis of COVEN is formed by CVE platforms developed in two of the partner institutions: dVS from Division Limited, and DIVE from the Swedish Institute of Computer Science. Both are mature toolkits, with the former being focused towards robust collaborative engineering applications, and the latter serving as a more general tool for the exploration of future CSCW applications. However both support some fundamental services in order to provide a *collaborative* environment, where users can co-operate in order to perform a task. These include support for awareness between users and ability to communicate.

Given the novelty of the COVEN applications, our first task was to lay the foundations for the usability evaluation through clarifying a framework for our activities. This outlined our goals, constraints, and general approach. Our work is based on three hypotheses:

- There are existing HCI design and evaluation methods for 2D applications which need to be translated to 3D/CVE applications and subsequently tested.
- There are CVE specific concepts based on human behavioural needs which are still under development and need to be explored and subsequently tested.
- There are CVE specific constraints on methodological aspects of evaluations which need to be identified and resolved.

The paper is structured as follows. In the following section we give an overview of the COVEN services and applications. We then explain our framework for evaluation in Section 3. The usability inspection and network trials are described in Sections 4 and 5. Section 6 discusses the direct results of these studies and their implications for future development.

2. APPLICATION OVERVIEW

The COVEN applications are built around a core set of services that were perceived to be essential for CVE technologies. Most central to collaboration is the support for *mutual awareness*, with possibly varying degrees. This involves awareness of the presence of other participants, but also recognition of the identity, role and current activity of the other participants. Also central to collaboration is, support for *communication* between participants. The users must be able to communicate with each other using audio and possibly other media including text, video and gesture. In order to allow multiple participants to co-operate there is a need for support of *resource management*. Resource management includes facilities to ensure that each user has a fair chance of accessing an object by introducing some kind of waiting queues, or, since the "laws of nature" can be different in a virtual world, multiple instances of an object. Another essential service is support for *participants roles and rights*. For example, this might cause the application to restrict the access required to manipulate an object to a specified group of participants. There is also a need for support of *object manipulation* including co-operative geometric edit operations as well as object hand-over. Finally, there should be support for *group navigation* within the environment, including support for a *global map*.

With a couple of exceptions these services are not found in single-user systems, and are indeed those aspects of a CVE system that distinguish them as so. The COVEN platforms themselves each support a subset of these services, and the remaining facilities have to be built into the applications themselves. Application development in COVEN is performed within a three-cycle software engineering process, and these applications are the product of the first cycle. The two COVEN application scenarios are a business conferencing suite, and a travel application.

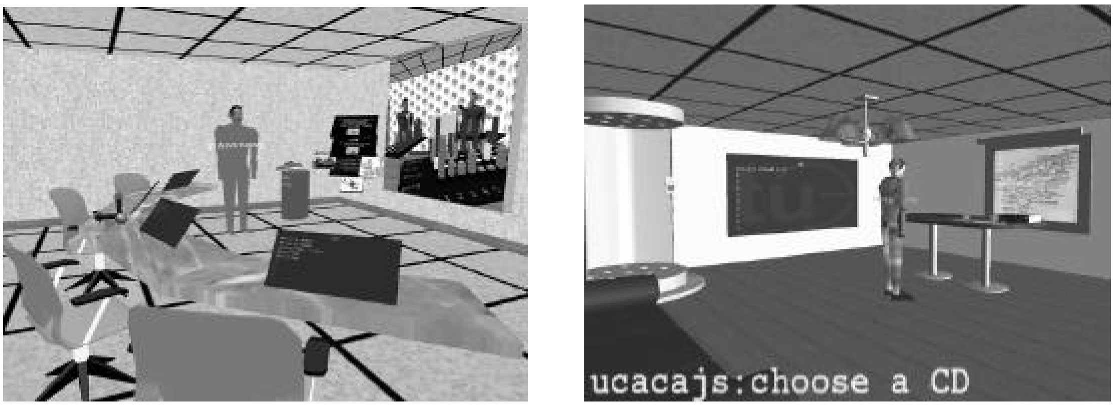

Figure 1 (a, b): Views inside the business and citizen applications

The business conferencing application supports two main tasks: collaborative presentations and a business game simulation. The application supports text message passing publicly or to named groups of receivers (users or message boards) to enable a basic level of communication, and participants are aware of each other through avatars which carry name tags. The presentation support is provided in a virtual conferencing suite shown in Figure 1a. In this room, users can communicate by passing objects to each other, or to display devices using a "bandy-box". In the figure, one participant has displayed a slide on the "holoview" large-screen display. The slide shows a view of the second main component of the business application - a spreadsheet like simulation based upon a simple model of product supply and demand. The spreadsheet consists of three desks, each with a set of bar graphs and controls which the participants can collaboratively manipulate in order to gain an understanding of the dynamics of the simulation.

The citizen application is broader in functionality, though it shares many of the same services for communication such as message boards, and heads-up message display. The scenario is one of visiting a virtual travel agency in order to retrieve information via 2D media (such as web pages and audio-visual presentation), and also through a visit to a virtual version of the destination. It consists of four zones. The first of these is the virtual travel agency which has doors leading to different zones for each possible destination. Only one of these destination specific zones exists at this time and this is the Rhodes Zone. In this room, groups of people can access information, and take advantage of the communications services to plan a group excursion (see Figure 1b). The individual or group can then use the teleporter (seen on the left in Figure1b) in order to travel to a virtual version of Rhodes which includes visualisations of hotel data, and re-constructions of ancient sites. Figure 2 shows a view of the virtual Rhodes zone during a network trial. Two other users are visible in the zone and the group as a whole has certain communication and navigation capabilities. The view we are seeing is that of the group leader, this is reflected by the user's having the notebook token which can be seen in the upper right corner of the screen. The final zone is a reconstruction of the temple of Lindos.

3. FRAMEWORK FOR EVALUATION

Virtual Environments are based on a family of new technologies configured in new ways to perform new functions, and one would expect this to pose problems when trying to apply existing evaluation methods. Indeed, a general tendency to ignore or minimise the evaluation has been observed (Durlach & Mavor, 1995). We therefore perceived it as an integral part of our evaluation task to develop and apply a systematic methodology to our evaluations in order to contribute to the development of a special evaluation tool kit for this area.

In the absence of an existing dedicated CVE evaluation methodology, our first task was the design of a framework for the COVEN usability evaluation, addressing the methodological constraints specific to CVE evaluation. Constraints are caused by the prototypical nature of CVE applications, and the need to understand the essential human behavioural needs which drive the development of the components of CVE technology, so one has to strike a balance between the concerns of *usability engineering* and *scientific enquiry* frameworks. In the context of CVEs these frameworks have not been developed, though certainly our viewpoint is represented in recent work on single-user VE technology (for example see Kaur, 1997; Rosenblum et al., 1996; and Welsh et al., 1996).

Evaluation of VEs and VE applications has progressed along two main fronts: demonstration of benefits to be achieved by the use of VE technology in particular applications domains, and general evaluation of VE systems themselves in order to make a "better" VE system. Measuring benefits of particular VE applications is a necessary technique in order to justify the continued development of the underlying hardware and software. However such studies (for example, Rosenblum *et al,* 1996 or Amburn and Marshak, 1996) are fairly specific in nature and generally do not involve case based studies of the elements of the system itself. Many studies have been undertaken in order to provide information about deficiencies in current technologies and comparison between alternative techniques (for example, Ware *et al.* 1993 or Piantanida *et al,* 1992).

A number of aspects of CVE technology direct and constrain our evaluation (Tromp, 1997). Firstly, the COVEN project is developing demonstrator applications intended to demonstrate the added-value of the CVE concept from the end-user and the customer point of view. This requires an evaluation of the overall usability of the CVE applications, together with elements of a cost/benefit analysis. This places our evaluation activities in the general framework of usability engineering methods (Melchior, Bösser, Meder, Koch & Schnitzler, 1995). However, standard usability engineering methods are developed for 2D applications and can not be expected to be directly applicable to 3D collaborative applications.

Secondly, the CVE technology on which these demonstrators are built is in its early stages and the human factors impact of its essential features still poorly explored (Durlach et. al, 1995). One of the objectives of COVEN is to investigate the human behavioural aspects which affect performance and satisfaction in CVEs. This requires focused exploratory studies of specific phenomena, which places our evaluation activities in the general framework of scientific inquiry (Groot, 1969). Focused studies should provide insight into the specific and unique features of CVEs, especially with regards to central concepts such as presence, co-presence, and awareness of one's virtual self and other participants. The general goal all VEs have in common is to create a sense of presence (Tromp, 1995; Slater and Wilbur, 1997). The specific goal of CVEs is to allow users to manage multiple tasks for collaboration.

Thirdly, due to the 3 dimensional, distributed, and prototype nature of CVEs there are a number of factors which further characterise the specificity of CVE usability studies. CVEs attempt to create a 3 dimensional place for people to interact in, so we are not only interested in human behaviour and performance with the application, but also in human behaviour and performance inside the CVE. Thus, in general observations and experiments need to be performed both from outside the CVE and from inside the CVE. Furthermore, a CVE allows multiple geographically distributed users to interact simultaneously within the CVE in real-time. As a result, network traffic is a major hindrance to system performance. Another implication due to the geographic distribution of the subjects, is that it is complicated to conduct proper controlled experiments. Finally, the prototype nature of the applications becomes another methodological issue in that it is often not feasible within the time and effort available to create different conditions for experiments, which means that the process of scientific inquiry is constrained. Also, there may remain defects in the functioning of the application, which means that it is difficult to conduct end-user experiments.

These methodological considerations form the basis of our framework for evaluation (COVEN D3.4, 1997). From this framework three main threads of work were derived:

- Usability inspections of the initial applications so as to uncover the main design flaws and allow to clean up the design, meanwhile adapting the method to 3D collaborative aspects.
- Observational evaluations of participants performing tasks in networked trials, so as to explore and better understand human behavioural concepts introduced by the CVE technology.
- Isolated auxiliary case-controlled experiments focused on the evaluation of factors of the central CVE concept of *presence*.

The results of these separate threads identified a significant number of usability issues at system, interaction and application levels, most of which will be applicable to all CVE systems and applications, not just those developed by COVEN. The following sections report on the first two threads of research. The third thread is reported elsewhere (Slater *et al,* 1997a; Slater *et al*, 1997b).

4. USABILITY INSPECTION

From the traditional usability engineering approach we identified the heuristic evaluation and cognitive walkthrough methods (Nielsen, 1994) as suitable techniques for this first evaluation. These techniques provide relatively quick feedback on prototype applications, and no experimental subjects are needed. Potential problems originating from the 2D, single user application focus of the techniques were identified, since they do not particularly address the issues of 3D, collaborative tasks that were expressed in our framework. Firstly we had to allow for the fact that CVE participants have a lot of freedom of navigation and manipulation and that interaction dialogues are not easy to constrain. Secondly we thought it dangerous to demand minimalism in the design of VE since the extra detail might help to enhance the participant's perception of space and sense of presence. Thirdly, an extra quality factor was introduced: adaptedness of the application to collaboration support.

Four independent inspectors carried out the inspection using task breakdowns of three scenarios for each application. The issues were collated and separated in to three classes: i) System problems including lack of functionality, performance and display quality. ii) Interface problems that concern the actions of navigating, and picking and selection of objects. iii) Application specific problems concerning the actual actions and meaning of objects within the environment. The complete list of issues is too extensive to list here in detail. We discuss the key issues in each category and give examples from the inspection reports where appropriate.

System Issues

System issues pervade all applications built upon the system. A typical issue is that of the simulation slowing or stopping when new scene components are loaded when changing zone. In our experience this is an issue with almost all VE toolkits. The important point here is that interaction and navigation become impossible for a very short time, as a result of a seemingly innocuous action. Given the user's expectation of free movement at all times, this suggests to the user that an error has occurred, or that the operation failed. This is also potentially serious for immersed users since the visual and proprioceptive cues will conflict.

A collection of issues were based around latency of certain event information, due to the centralisation of certain key services which would be too expensive to replicate. Again this problem is inherent in most VE toolkits. In this case it was manifested as asynchronous visual, audio and collision behaviour, and the most apparent result was a problem navigating through doors since the collision detection would de-activate before the visual cue of the door opening took place and resume whilst the door appeared to still be open. Other main usability issues included the lack of general undo for 3D environments, the lack of rules to allow users to lock objects during interactions and lack of spatialisation of audio.

Interaction Issues

The most fundamental issue with the interface of the COVEN demonstrators is the multiplicity of layers at which interaction occurs. There are aspects of all of the following in use in the applications: keyboard input, 2D Widget interaction, continuous mouse driven 3D control, and discreet 3D Widget interaction. Deciding which "level" of input is required was considered a major burden. Many quite simple actions would involve several interaction stages, and much of the swapping was due to the need to select suitable viewpoints from which to perform certain tasks. This is a fault found with most VE desktop interaction systems that support free navigation. Integrating all controls into one coherent structure is difficult on a desktop display because of the relative difficulty of providing easy to use 3D control. It is also difficult in an immersive system because of the difficulty of providing alternatives to text.

The primary issue with the 3D control was that of the modal control necessary in order to map navigation selection and manipulation to the mouse. Selection is performed with the left button, navigation with the middle, and manipulation with the right button after the object is selected. In addition there are subsidiary modes for navigation (left-shift enables slide), and manipulation (dragging in different area of screen mapped into different directions in the world). This led to some confusion over which mode is operative, and would occasionally lead to mistakes.

Other important issues include the fact that it was found to be impossible to carry objects using the desktop interface since navigation and manipulation are exclusive. Object-centred navigation was also thought to be desirable given the nature of some of the application tasks (notably inspecting the spreadsheets in the business applications). This finding is consistent with the central findings from the longitudinal evaluation of the MASSIVE-1 network trials, where users requested automatic behaviours, both for their own embodiments and for objects in the CVE (Tromp & Snowdon, 1997).

Application Issues

The application issues are broad in nature, from problems with objects whose operation is not obvious such as the teleporter, to wider topics such as how best to represent group services to group members. The application issues can be classified into issues concerning the collaboration services and application specific problems. The main issues which are raised about the collaboration services include some problems with the communication channels. In particular, problems were found with the lack of error feedback when sending text messages to named recipients, inconsistency of object reaction to selection and manipulation, mutual awareness between participants, and the mechanisms of group formation.

The inconsistency of object reactions was manifest in several situations and is perhaps the most important issue. Objects that have no behaviour can be selected, suggesting to the user that maybe the object does have some behaviour after all. Object actions are variously triggered by selecting the object or by trying to pick it up. Also, different objects turn different colours upon selection, the reason for which is not made clear to the user. Second in importance was the problem in supporting group formation. The group is initially formed when a leader picks up a notebook, which acts as the token of leadership. Participants then join the group by selecting the notebook token. This was problematical because it was not obvious that the group leader needed any special responsibilities, and it was also not obvious when the group was to be formed. Although some scenarios, such as a guided tour, need one person to have extra facilities such as group transportation, forcing this upon all groups was thought to be unnecessary.

The application issues were generally concerned with the affordances of objects and the lack of help with the VE itself. Two examples illustrate the breadth of issues. Firstly audio-visual presentations are available in the citizen application and the metaphor is that of using a CD-ROM player. However, the playing of a CD is activated by the initial pick event on the CD causing the CD to “fly” to the player. In effect this makes it impossible to pick up a CD and inspect it, which breaks the metaphor and causes confusion. This metaphor was originally chosen to circumvent the difficulty of performing the action of placing the CD in the drawer when using a desktop system. Secondly, travel to Rhodes was effected by entering a teleporter (see Figure 1b). However, this metaphor was complicated by the operation which involved 3 stages, firstly opening the teleporter, standing on the pad and then pressing the teleport button. This sequence is more akin to using a lift and might be better represented as such to make it easier to recognise the sequence of actions.

5. NETWORK TRIALS

The network trials were found to be useful for exploring CVE technology concepts such as how people used the various communication media, and to what extent they could collaborate successfully. 21 trials where held before the end of September 1997 with each trial lasting approximately one hour. Data was gathered using web-based form questionnaires posed after the end of each trial. Thirteen trials used the business application and five used the citizen applications with some trials using both, and others using default demos provided by the dVS software. The four regular participants in the trials were all experienced in the use of CVEs, though the particular platform was new to some.

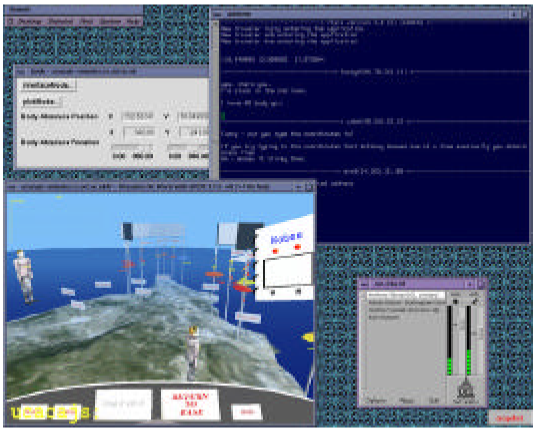

Figure 2: View of the Rhodes Zone of the Citizen Application during a network trial session

The dVS platform does not currently support audio directly, so the robust audio tool (RAT)[1] was used. In addition a collaborative chat system derived from ytalk[2] was found useful as a backup channel. Figure 2 shows the complete set-up during one of the network trials. Clockwise from top right the windows are: ytalk session with 4 participants, RAT interface showing the names of the participants, the dVS window showing a view of the Rhodes zone and a control window for the user's body.

There were three communication channels in the trials: audio, text and virtual environment. All three proved valuable for different reasons, but actual usage was heavily influenced by the perception of reliability of the channel. The audio channel was regarded as useful but not totally reliable. In particular not knowing if one's audio segment had been received by the others was found to be a problem. The reliability of the text communication was found to compensate for any lack of clarity in the audio. This is consistent with similar suggestions made by Benford et. al. (1996). However, the audio was seen as having immediacy and better support for expressing emotion.

There were two means of supplying text, the ytalk window, and text messages within the VE. The former was almost exclusively preferred since several messages would be visible at once and it was possible to use text functions on this window. However, the benefits of the VE text were that it was possible to send private messages, and that the participant did not need to switch focus to another window. A more serious side effect of the lack of integration between ytalk, RAT and the VE was that the different applications could not share state about who was trying to communicate in any particular channel. In particular it was found to be a problem that the avatars of the users did not animate when the person was speaking or typing, which is a serious issue since it becomes impossible to tell if two people are communicating, solely from their appearance and position. This is consistent with observations and predictions made by Heath et. al. (1995). As it is, proximity in the VE alone is neither a sufficient nor necessary condition for two participants to be communicating, and this interferes with effective collaboration.

Overall the participants found the applications useful and interesting, and application issues that were raised were usually about desired functionality. The criticisms that were made reflected issues also raised by the inspection, but yielding considerable detailed suggestions for improvements. This is consistent with expectations about using CVE experts as subjects, as expressed by Tromp (1997).

---

[1] RAT Software available from ftp://cs.ucl.ac.uk/mice/rat/current

[2] ytalk software available from (e.g.) ftp://ftp.cdrom.com/FreeBSD/packages-current/All/ytalk-3.0.2.tgz

6. DISCUSSION, CONCLUSION AND FUTURE WORK

Overall the usability inspection and network trials were successful in that they generated many issues that will drive the development of the COVEN applications, CVE technology and CVE evaluation techniques. The nature of issues generated by the network trial questionnaires is different to those from the inspection in that the participants are generally concerned with, and questioned about the actual utility and efficiency of the collaboration services rather than application components. This vindicates the extra effort of studying user behaviour and opinions at an early stage in the project, even if those users are not typical end-users. The one type of issue that was considerably more visible in the network trials than the inspections was the synchronisation of events across different media and perceived lag in object reactions.

CVE System Design

Usability issues caused by system issues could be coined CVE technology artefacts. Fundamentally an issue such as the scene freezing arises because the resource needed in order to generate a display is not immediately available. To make an analogy what is required in this case is the CVE equivalent of a window manager busy icon in order to indicate the relevant system state in addition to any application state that might be presented. We have seen that system issues can affect both interaction capabilities and make application tasks harder to perform. Thus certain application tasks should be easier if system issues were resolved. However other dependencies can not be inferred. It might be a case that an interaction style that solves one application problem causes another. There is evidence that in the desktop condition no one interaction style is suitable for all tasks (Steed and Slater, 1995). Such studies on desktop interaction styles have generally concentrated on individual tasks such as selection or navigation and it remains to be seen how the appropriate metaphors might be selected for a given large-scale task. Designing applications to be independent of the interaction style thus seems to be a desirable goal but, as the issue of the CD player demonstrates, this is difficult. What might be required is higher level representation of tasks that can be mapped onto different actions by the system, similar to an automated system for user interface component selection, proposed by Tromp and Snowdon (1997).

Although problems regarding system issues cause serious usability problems, it is not possible for the application author to resolve when they are coded into the system clients. It is also not obvious that any application choices that are appropriate for one interaction style will be suitable for all styles. For example in the default desktop interaction metaphor objects could be selected at any distance. This leads to situations where you may see an object being manipulated but not be sure who is manipulating the object since it could be anybody who had the object in view. This will not apply in the immersive case, since the default interaction metaphor has selection based on hand-object collision rather than ray-object intersection.

CVE Interaction Design

Guidelines for CVE development have not yet been explored to any great extent, although work on guidelines for single user VEs is underway (Kaur et. al., 1997). What is apparent is that there is a tension between making objects realistic in order to aid comprehension of space, and overloading objects with affordances that can not be supported. This has resulted in a number of problems in the applications. The overall metaphor of both applications is of realistic environments reminiscent of places that one might actually visit to perform the real-world task. However this is confounded by certain functionality being presented with metaphors that although consistent within themselves, are not consistent with the overall world metaphor (e.g. the teleporter). The role of metaphor in the environment would seem to be more complex than in 2D interfaces and thus the usability inspection process will need to be developed to reflect this. The experience of the network trial provided some direct guidelines for us to follow. As far as possible, the avatar of the user should represent all the actions that they are performing on the environment. This includes indicating the focus object manipulation and audio and text communication.

The issues generated by the inspection could be neatly categorised into system issues, interaction issues and application issues, which also seems a useful division for future work. Each category of issues is to be addressed at a different level of the development process, with system and interactions being within the domain of the CVE system designer, and application issues being the domain of the CVE application writer. We have presented some important outstanding issues for the system designer, and have suggested a few potential pitfalls that the application builder should avoid.

ACKNOWLEDGEMENTS

We would like to thank all those who took part in the usability inspections and network trials, including Adrian Bullock, Judith Dijkhuis, Veronique Normand, Andrew Fuzniak, Kurt Donkers and Gidi van Liempd. Our thanks also go to the rest of the members of the COVEN project.

REFERENCES

Amburn, P., Marshak, W.P., (1996). Design and Evaluation of an Air-to-Air Combat Debriefing System Using a Head-Mounted Display, *Proceedings of the IEEE 1996 Virtual Reality Annual International Symposium*, March 30- April 3 1996 , Santa Clara, California.

Benford, S., Bowers, J., Fahlen, L.E., Greenhalgh, C., Snowdon, D., (1995). User Embodiment in Collaborative Virtual Environments, *in Proc. ACM Conference on Human Factors in Computing Systems (CHI'95)*, ACM Press, Addison Wesley, pp. 242-249.

COVEN, (1997). Usage Evaluation of the Initial applications, Public Deliverable 3.4, A.Steed (ed), October 1997.

Durlach, N.I., Mavor, A.S., (1995). *Virtual Reality: Scientific and Technological Challenges*, National Research Council, National Academy Press, USA.

Heath, C., Luff, P., Sellen, A., (1995). Reconsidering the Virtual Workplace: Flexible Support for Collaborative Activity, in *Proc. of the European Conference on Computer Supported Cooperative Work* (ECSCW'95), Sweden.

Kaur, K., (1997). Designing Virtual Environments for Usability, Doctoral Consortium paper in: *Proceedings of INTERACT'97*, Sydney, July 1997, IFIP.

Kaur, K., Maiden, N., Sutcliffe, A., (1997). Interacting with Virtual Environments: an evaluation of a model of interaction, in: *Interacting with Computers*, Special Issue on VR.

Melchior, E-M, Bösser, T., Meder, S., Koch, A. and Schnitzler, F. (1995*). Handbook for practical usability engineering in IE projects*, Project ELPUB 105 10107, Telematics Application programme, Information Engineering section.

Nielsen, J, Mack, R.L, (1994). *Usability inspection methods*, John Wiley and Sons, New York, NY.

Rosenblum, L., Durbin, J., Obeysekare, U., Silbert, L., Tate, D., Templeman, J., Agrawal, J., Fasulo, D., Meyer, T., Newton, G., Shaley, A. (1996). Shipboard VR: From Damage Control to Design, *IEEE Computer Graphics and Applications*, November 1996.

Slater, M., Chrsyanthou, Y., (1997). View Volume Culling Using a Probabilistic Caching Scheme, in *Proc. ACM Virtual Reality Software and Technology VRST'97*, Lausanne, Switzerland 1997.

Slater, M., Steed, A., McCarthy, J., Maringelli, F., (1997a). The Influence of Body Movement on Presence in Virtual Environments, submitted for publication.

Slater, M., Steed, A., McCarthy, J., Maringelli, F., (1997b). The Virtual Ante-Room: Assessing Presence through Expectation and Surprise, submitted for publication.

Slater, M., Wilbur, S., (1997). A Framework for Immersive Virtual Environments (FIVE): Speculations of the Role of Presence in Virtual Environments, *Presence*, 6(6), MIT Press.

Steed, A., Slater, M., (1995). 3D Interaction with the Desktop Bat, *Computer Graphics Forum*, 14(2) 1995.

Tromp, J., (1995). Presence, Telepresence and Immersion: Interaction and Embodiment in Virtual Environments, in *Proc. Framework for Immersive Virtual Environments (FIVE'95)*, QMW College, London, December 1995.

Tromp, J.G., (1997). Methodology of Distributed CVE Evaluations, Poster Presentation at UK VR SIG'97, in: *Proceedings of UK VR SIG 1997*, Bristol.

Tromp, J.G., Snowdon, D., (1997). Virtual Body Language: Providing Appropriate User Interface in Collaborative Virtual Environments, in *Proceedings of ACM Symposium on Virtual Reality Software and Technology*, Lausanne, Switzerland.

Ware, C., Osborne, S., (1990). Exploration and Virtual Camera Control in Virtual Three Dimensional Environments, *Proceedings of 1990 Symposium on Interaction 3D Graphics*, ACM Press.

Welsh, R.B., Blackman, T.T., Liu, A., Mellers, B.B., Stark, L.W., (1996). The Effects of Pictorial Realism, Delay of Visual Feedback, and Observer Interactivity on the Subjective Sense of Presence, in *Presence: Teleoperators and Virtual Environments*, 5(3), MIT Press.