Systematic Usability Evaluation and Design Issues for Collaborative Virtual Environments

Abstract

This paper presents results of the longitudinal usability and network trials that took place throughout the COVEN (COllaborative Virtual ENvironments) Project. To address the lack of understanding about usability design and evaluation for collaborative virtual environments (CVEs), a deductive analysis was used to systematically identify areas of inquiry. We present a summary of the analysis and the resulting framework through which various complementary methods were utilized during our studies. The objective of these studies was to gain a better understanding about design, usability, and utility for CVEs in a multidisciplinary setting.

During the studies, which span four years, we undertook longitudinal studies of user behavior and computational demands during network trials, usability inspections of each iteration of the project demonstrators, consumer evaluations to assess social acceptability and utility of our demonstrators, and continuous preparations of design guidelines for future developers of CVEs.

In this paper, we discuss the need for such activities, give an overview of our development of methods and adaptation of existing methods, give a number of explanatory examples, and review the future requirements in this area.

1 Introduction

Collaborative virtual environments (CVEs) are a novel application area of computing technology, demanding an understanding of human-computer interaction in 3D real and virtual environments, and of human-human interaction as mediated by computers and virtual spaces, not yet available in the human-computer interaction (HCI) literature. As yet, there have been few reported contributions to the structured design and evaluation of virtual environments (VEs). In turn, this has contributed to and also has been a consequence of the paucity of guidelines about the usability in VEs (Wilson, 1996). Understanding of what is meant by usability in VEs is relatively poor, and there is little agreement on which attributes among the many variables of the VE interface are significant, whether they are different in different use scenarios and user circumstances, and how these might be operationalized in design practice. Since the start of the COVEN project, a number of efforts have appeared in print that address this lack of structured advice on design and evalua-
tion. Amongst these are Gabbard (1997), who attempted to create a taxonomy for VE design; Parent (1998), who created a virtual environment task analysis workbook for the creation and evaluation of virtual art exhibits; Kaur, who has defined a number of interaction cycles for single-user VE interaction, from which design properties were proposed that significantly increased the usability of their test application (Kaur, Sutcliffe, & Maiden, 1998); Fencott (1999), who developed a model to aid the design of VE content; Hix and colleagues, who have given a convincing example of the cost effectiveness of iterative usability design and evaluation through assessments and improvements of a VE for battlefield visualization (Hix, Swan, Gabbard, McGee, Durbin, & King, 1999); Eastgate (2001) and Wilson, Eastgate, and D’Cruz (2002) have developed a structured design and evaluation process, VEDS (the virtual environment development structure) whose major stages are analysis, specification, and storyboarding; building VEs; and enhancing user performance, presence and interactivity, and usability and evaluation.

If we extend our review to consider CVEs as against virtual environments used by single users or else groups of users employing a single display screen, we find little advice on usability, design, and evaluation. Exceptions are work on system improvements, that is, the more technology-oriented work, which include Greenhalgh’s 1999 dissertation on network load of CVEs, Lloyd’s 1999 dissertation on group formation in CVEs, and Reynard’s 1998 dissertation on video integration in CVEs to create mixed realities. A sociologically oriented research project is Hindmarsh’s 1997 dissertation on the collaboration process in real and mediated spaces. Fraser’s PhD work uses ethnographic observations of CVE users to guide design improvements for the support of object-focused interaction between CVE participants.

This paper describes the evaluation processes and associated evaluation methods for the assessment of early prototypes of CVEs. The work in the paper has grown out of the need to conduct formative evaluation within a particular project, but the developments and lessons learned have been generalized as far as possible. The main discussion will be of the choice, development, and use of a number of evaluation methods and their integration into a coherent approach. However, in employing these methods within a real CVE development project, we have identified a number of emerging ideas on CVE design that allow a start to be made upon defining guidelines to assist CVE designers. How usability applies to the design of CVEs is a very broad topic, and, other than making comments on CVE usability factors, this paper concentrates on how people actually collaborate within CVEs and upon the design contributions that will enhance and support collaboration.

In subsection 1.1, we discuss the goal of the usability evaluations of the COVEN project. In section 2, we review collaboration technologies in general and place CVE technology in it. We also describe the current design practice within which CVE development is taking place. In section 3, we give an overview of the framework we devised for the COVEN development and evaluation activities. We introduce the four major threads of work and the standard evaluation methods, which we adapted so that they can now be profitably applied to CVEs. In section 4, we describe our evaluation strategy and utilization of the evaluation methodologies. In section 5, we describe the results we generated by using these methods, and in section 6 we discuss the merit of the methods, including clarification of new design issues introduced by CVE technology development.

### 1.1 Evaluation Within the COVEN Project

The goal of the COVEN project was to demonstrate the feasibility and utility of scalable CVE worlds through prototype applications in the general area of virtual travel, summarized by Normand et al. (1999). Demonstrators and CVE platforms were built in three engineering cycles, with each implementation phase followed by an evaluation phase. Each evaluation phase involved a weekly or fortnightly network trial scheme with user-tests to collect system, network, and usage data. Descriptions of the COVEN platform, some of the demonstration applications, and technical results from the network trials can be found in companion papers.
(Frécon, Smith, Steed, Stenius, & Ståhl, 2000, Greenhalgh, Bullock, Frécon, Lloyd, & Steed, 2000). The happy frequency of the network trials allowed us to adopt a longitudinal approach for our usability evaluations. The role of evaluation was originally to perform a summative evaluation of the success of the applications themselves. However, from a very early stage, it was apparent that more formative evaluations were required to develop the theoretical underpinnings of our approach, and that, instead of focusing specifically on the applications, it was also important to consider the general task of collaboration, the 3D nature of CVEs, and the methods used for evaluation.

To address the lack of VE-specific usability tools, COVEN performed investigative empirical work, contributing to the understanding of CVE design. Our point of origin was that we needed to address the usability problems posed by CVE technology by investigating the human behavioral aspects that affect performance and satisfaction in VEs. (See Figure 1.)

Our approach to method development was based on and constrained by three premises:

1. Existing HCI design and evaluation methods for 2D applications that need to be translated to 3D/CVE applications and tested for their appropriateness.

2. CVE-specific concepts are being developed from an understanding of human behavior generally that are still poorly understood and need to be explored, tested, and refined.

3. CVE-specific constraints on evaluation methodology need to be clarified and incorporated in the experimental setup.

It should be noted that the current state of knowledge and the relatively low level of sophistication of CVEs do not yet support the development of completely new evaluation methods and metrics. In any case, this is probably not appropriate. We have looked to draw upon, to adapt where appropriate, and to utilize existing methods of evaluation, some from human-computer interaction generally (Helander, Landauer, & Prabhu, 1997) and some from evaluation in other settings (Bales, 1951; Neale, 1997). Such methods and approaches have been used as they may be appropriate to the three-dimensional, real-time, and shared experiences available within CVEs.

It would be wrong to discuss evaluation of collaboration within CVEs as if this were somehow a totally new phenomenon. There have long been studies of peoples’ interaction and collaboration in real-world settings (Goffman, 1967; Argyle, 1969; Kendon, Harris, & Ritchie Key, 1975) and, more recently, of collaboration within other technologies such as audio conferencing, video conferencing, computer-supported cooperative work and media space conferencing (Gaver, Sellen, Heath, & Luff, 1993; Finn, Sellen, & Wilbur, 1997; Hindmarsh, Fraser, Heath, Benford, & Greenhalgh, 1998). Understanding, evaluation of, and design to enhance collaboration within CVEs should all be informed

Figure 1. Framework within which the development of the COVEN CVE evaluation and design strategy took place.
by an understanding of collaboration in these other environments.

2 Background to CVE Technology

To better understand the technological developments that culminated in the development of virtual environment conferencing software, we present a short overview of previous collaboration technologies. (See Table 1 for a summary.)

Audio conferencing has been found to be notoriously difficult to manage, in terms of managing turn-taking between multiple participants, and for large groups in terms of identifying who is speaking (Walters, 1995). Video conferencing was a natural technical extension of audio conferencing, allowing for the inclusion of facial expressions and a limited set of gestures to be part of the group interaction. However, with video conferencing, dialogs have been found to be significantly longer, with more interruptions, than for audio conferencing, particularly when transmission is delayed (O’Malley, Langton, Anderson, Doherty-Sneddon, & Bruce, 1996). Also, sharing documents is still a problem because participants cannot see when and where in the document was being pointed at (Heath & Luff, 1991). With the introduction of media spaces—video conferencing with additional video cameras aimed at documents and workspaces—it became possible to include shared document views and an increased awareness of the video conference participants’ background for the other participants (Gaver, 1992; Gaver, Sellen, Heath, & Luff, 1993). However, it was difficult to ascertain which aspects of participants’ own activities and workspace were visible to their colleagues (Heath, Luff, & Sellen, 1995). Participants still had difficulties working

<table>
<thead>
<tr>
<th>Mediating technology</th>
<th>Definition</th>
<th>Known problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audio conferencing</td>
<td>Group working over telephone systems.</td>
<td>• Difficult to ascertain who is talking</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• No way to share documents in real time</td>
</tr>
<tr>
<td>Video conferencing</td>
<td>Group working via video connections. Inclusion of facial expressions and a limited set of gestures.</td>
<td>• Dialogs significantly longer, more interruptions</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Problem sharing documents because of lack of detail</td>
</tr>
<tr>
<td>Media space conferencing</td>
<td>Video conferencing with additional video cameras aimed at documents and workspaces.</td>
<td>• Fixed cameras leave gaps in the view of the remote space</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Difficult to make sense of colleagues’ conduct</td>
</tr>
<tr>
<td>Virtual conferencing</td>
<td>Group working via a shared computer-generated graphical space with avatars representing participants.</td>
<td>• Field of view limited</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Object interaction and navigation clumsy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Limited set of gestures</td>
</tr>
</tbody>
</table>
together because the separate, fixed cameras leave gaps in the view offered of the remote space. CVEs were then proposed as a virtual conferencing medium, an alternative to video conferencing (Benford, Bowers, Fahlen, Greenhalgh, & Snowdon, 1995). Virtual conferencing was seen as having several potential advantages over audio or video conferencing by providing a continuous workspace that can be shared by physically remote users in real time.

### 2.1 Collaborative Virtual Environments

CVEs provide virtual embodiments—or avatars—for users, allowing expression of a limited set of nonverbal behaviors, a communication channel (be it audio, text, or both), and shared spaces including shared objects (recently reviewed by (Liew, 2000)). The virtual embodiments provide users with a means to be collocated in the same virtual space, regardless of the actual geographical locations of each user—although within this system there is little support to convey gaze direction, facial expressions, or gestures that are occurring in real time, but with a few exceptions such as the inclusion of some facial expressions (Guye-Vuilleme, Capin, Pandzic, Thalmann, & Thalmann, 1998; Slater, Howell, Steed, Pertaub, Gaurau, & Springel, 2000) and live video images of the face (Reynard, 1998). Precise maneuvering of the virtual embodiment is difficult due to the limited input technology (such as the mouse, keyboard, and joystick). Little information is available about the real environment in which other participants are physically located, or about any changes that occur in their physical space that affect the behavior of that user in the virtual space.

Common behaviors that people exhibit during collaboration in the real world—such as eye contact, gaze duration, and touch—are not normally available in the virtual world, and are therefore not visible to others in the virtual space. Participants in a CVE have to establish and maintain a set of social norms or an etiquette of social conduct during the interaction, using a much lower bandwidth than is available in real-world settings. If a user cannot interact with the interface effectively, this creates a large cognitive overhead, and consequently the tasks involved in successful collaboration cannot be performed in an efficient and satisfactory way. A serious potential problem for usability evaluations of collaborative services is that dealing with the prototype technology that enables collaboration in CVEs generally creates a considerable cognitive overhead that distracts people from getting on with the actual collaboration task.

### 2.2 Design Considerations for CVEs

We considered a better insight in design practice an important guide in creating system design methodologies, an approach that has recently been emphasized by Shackel (2000). To find out where and how design guidance is lacking, we asked five CVE designers to tell us about their particular design problems and practices in an interview of an hour each. A basic form of analytical induction (Groot, 1969) was used to draw conclusions from the interviews. The interview questions were carefully developed based on two pilot studies on CVE designers, observations of the CVE designers in action, and our own background knowledge on CVE design issues. The designers interviewed all had a background in computer science and had been involved in building CVE demonstrators, ranging from work on one project for half a year to work on eight or more projects over the past six years. All but the pilot interviews took place in the workplace of the CVE designers, with the designers sitting behind the machine they typically used to work. This allowed them to illustrate issues by showing examples of their work on the screen. The results from the interviews gave an extensive amount of qualitative information about CVE design practices.

The CVE designers interviewed work within two somewhat contradictory constraints:

- **A human constraint:** the CVE has to be effective and intuitive for participants.
- **A machine constraint:** the CVE has to take up minimum computational load and network traffic.
The respective, potentially conflicting solutions to satisfy these two constraints are:

- use of realistic representations and metaphors to allow users to transfer their intuition and everyday knowledge to the VE, and
- simplification of these representations.

Throughout the design process, the design choices are influenced by a constant tension called the performance constraint (Howard, 1997). The performance constraint refers to the fact that the update rate decreases as the number of polygons increases. This constraint will always exist, although the polygon budget will increase over time as VR technology gets more advanced. As computers become faster, either model complexity or the update rates can increase, but rarely both. We therefore included in our usability activities the need to formulate and test minimum user and application needs. We attempted to find ways of conveying this information to the CVE designers in a suitable format that would allow them to design supporting user interfaces within the limited available computing resources. In other words, to give the designer team an informed strategy with which to reach decisions about simplification.

The interviewees interpreted “design” in two different ways. The same word was used to talk about two significantly different tasks:

- design in terms of computer code (assuming a designer with a computer science background), and
- design in terms of the form of objects (assuming a designer with an artistic or engineering background).

Each design task belongs to a different discipline, with different associated skills. However, both design tasks need to take into account how to make things usable for multiple, collaborative, distributed users. We therefore incorporated in our usability activities the need to formulate and test usability principles for CVEs in a suitable format that would allow different designers to carefully specify the perceptual message they wish to convey to CVE users. To this end, we developed a CVE usability design method that used the interaction cycles from the inspection, the combined project consortium expertise, usability guidelines, and the concept of narrative affordances as the guiding principle. (See subsection 5.2 and 6.2 for more details.)

A strong need was expressed for both types of guidelines. We therefore incorporated in our usability activities the task to formulate and test usability design principles for CVEs in a top-down functional analysis that would allow designers to carefully specify the perceptual message they wish to convey to CVE users. To this end, we developed a CVE usability design method that used the interaction cycles from the inspection, the combined project consortium expertise, usability guidelines, and the concept of narrative affordances as the guiding principle. (See subsection 5.2 and 6.2 for more details.)

To summarize the results from the interviews, the design process as a whole is governed by making design decisions that satisfy a performance constraint. At the same time, the design requirements are driven by the usability demands of the users. The only way to allow both driving forces to have an equal impact on the final design is by:

- identifying those aspects of the design that are directly influenced by tradeoff decision-making,
- clarifying user needs and computational needs to guide the reasoning that leads to design decisions, and
- making informed computational simplification decisions based on an understanding of the specified user needs.

In the absence of precise guidelines for CVE usability design, the design options should be discussed with a complete design team, minimally composed of a designer, a programmer, a usability expert, and a client.
The team should work in unison so that none of the design tradeoffs is made by one expert alone. In previous work, we have tried to increasingly clarify our understanding of those design tradeoff areas (Steed & Tromp, 1998; Tromp, Istance, Hand, & Kaur, 1998), and we have summarized the results in Table 2.

### Table 2. Tradeoff Design Areas and Associated Usability Problem Categories

<table>
<thead>
<tr>
<th>Category of CVE usability problem</th>
<th>Design tradeoff decision dimension</th>
<th>Usability problem areas</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware/network/software problems</td>
<td>prototype development vs. demonstrable applications</td>
<td>lack of functionality</td>
</tr>
<tr>
<td></td>
<td></td>
<td>latency in performance</td>
</tr>
<tr>
<td></td>
<td></td>
<td>poor display quality</td>
</tr>
<tr>
<td>System problems</td>
<td>runtime performance vs. user performance</td>
<td>usability solutions not automatically device independent</td>
</tr>
<tr>
<td></td>
<td></td>
<td>high-ping users judged as uncooperative, low-ping users judged as uncollaborative</td>
</tr>
<tr>
<td></td>
<td></td>
<td>high-end users judged as higher in status, competence, and trustworthiness than low-end users</td>
</tr>
<tr>
<td>Application problems</td>
<td>object representation vs. affordance representation</td>
<td>the meaning of objects within the environment</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the apparent or unapparent availability of actions</td>
</tr>
<tr>
<td></td>
<td></td>
<td>realism vs. simplification choices based on performance constraints only</td>
</tr>
<tr>
<td>Interface problems</td>
<td>presence and copresence vs. minimalist design</td>
<td>interaction struggles in 3D space, such as navigating in 3D space, picking of 3D objects, and positioning precisely</td>
</tr>
</tbody>
</table>

3 Framework of COVEN Development and Evaluation

In accordance with the Telematics Programme of the Fourth Framework Programme of the EU, the COVEN project was planned and executed in a user- and customer-driven manner. This user-centered design process meant that the entire development process, including initial planning and the feasibility studies, included and emanated from the analysis of user needs and requirements. The principal aim of the evaluation work packages was “internal” in that they were either formative evaluations of demonstrators under development or summative evaluation of the final demonstrators and platforms. However, incorporating the additional requirements for methodological and scientific development discussed in section 2, we arrived at four main threads of work. (See also Figure 2).

- **Usability inspections**: checking the application so as to uncover the main design flaws and cleaning up the design, while adapting the method to include inspection of those parts of the design that are expected to support 3D interaction and collaboration

- **Observational evaluations**: participants performing controlled experimental tasks in networked trials focused on the evaluation of factors of the central CVE concept of presence, copresence, interaction, and collaboration, so as to explore and better understand these concepts

- **Consumer evaluations**: assessment of the attitudes to travel information CVEs amongst the general public and the perceptions of added value of CVE travel applications
• Usability guidelines: formulation of our design practice, design problems, and design solutions in a structured manner, as a first set of guidelines.

3.1 Inspection Method

Usability inspection is a generic name for a set of methods in which evaluators inspect or examine usability-related aspects of the user interface (Nielsen & Mack, 1994; Helander, Landauer, & Prabhu, 1997). Usability inspections are normally used at a stage in the usability engineering cycle when a first user interface design has been generated and its usability and utility for users needs to be evaluated, but users cannot be used yet due to the prototypical nature of the software. The inspection identifies usability problems, ranking them in order of seriousness for usability breakdown, and recommending ways of fixing the problems. The inspection methods have been developed on, and for, 2D applications, so we proceeded cautiously, reflecting on the effectiveness of the method as we applied it to 3D interfaces, and subsequently adapting it where necessary during the three iterations of our project.

The inspection method we found most directly applicable was the cognitive walkthrough method and additionally we attempted to use the heuristic evaluation method (Melchior, Bösser, Meder, Koch, & Schnitzler, 1995). For a recent overview of the method, see also Preece, Rogers, and Sharp (2002). This method uses an explicitly detailed procedure to simulate a user’s problem-solving process at each step in tasks employing an interface to see if the simulated user’s goals and memory for actions can be assumed to lead to the next correct action. In the case of CVEs, such user guidance is particularly important because of the freedom in the choice of interactions that is typical for these 3D environments. Inspections do not involve actual users, so the method is particularly suited to be applied immediately after the release of each design.

We performed our inspections with multiple inspectors and combined our results to collect maximum information, and we reflected as a team on how to adapt the method to 3D applications. Additionally, we asked our designers to perform an inspection, using our final adapted method description, to access the educational benefits to be gained by exposing inspectors without usability expertise to the usability concerns of CVEs.

3.2 Observation Method

The introduction of a new technology into the workplace or home can have unknown interactions and consequences, that affect the perceived usability of the technology. The organizational and social setting into which computers are integrated directly influences the success (or failure) of the computer software and needs to be taken into account to achieve the maximum ecological validity of the research data (Vicente, 1999). We therefore conducted observations of the actual use of our application in the working context of the real networked system connecting geographically dispersed users performing representative tasks in the CVE. Ethnographic observation analyzes sequential interactions and generally leads to a rich and detailed understanding of the meaningful relations between otherwise seemingly unrelated acts. (See, for example, Heath et al., 1995). The great advantage of this approach is that it gives insight into real acts in the real activity context in which they take place, thus greatly increasing the quality of the data (Lindgaard, 1992) in terms of the depth of the
derived insights. The disadvantages of this method are that it leads to largely anecdotal evidence and has few quantitative data to support its claims. It also tends to lead to concentration upon small (geographically, temporally, functionally, and interpersonally) fragments of behavior, potentially missing the “big picture” needed to make recommendations for redesign. Therefore, we developed a method to perform a sequential interaction process analysis of the temporal and spatial activities of CVE users. Interaction process analysis is a standard empirical method described by Bales (1951) for use originally in observation of small group interactions as they occur. The heart of the method is a way of classifying direct interaction as it takes place, act by act, and a series of ways of summarizing and analyzing the resulting data so that they yield useful information (Bales, 1951; Kanuritch, Farrow, Pegman, Wilson, Cobb, Crozier, & Webb, 1997; Neale, 1997). This method provides a means for collecting quantitative data from observations. We observed both experienced and novice users.

### 3.3 Consumer Evaluation

During the COVEN project, there were two consumer evaluation phases. One took place at the beginning of the project and established the need and requirements of a travel information CVE, based on the experiences of travel agents. The other took place at the end of the project and consisted of two separate consumer evaluations: one in-situ in a tourist agency and the other in a controlled experimental setting.

At the beginning of the project, well-known travel agents were surveyed to establish travelers’ needs in the context of their age and their social, economic, and background knowledge. Travel agents were interviewed, and the most frequently asked questions by customers were stratified and analyzed. These were cost, where to go and what to see, what to do there, and how to get there. The application and usability requirements specification included these and other CVE-specific traveler activities (such as exploration and travel rehearsal) in increasingly detailed scenario descriptions and function specifications (COVEN Del 2.1, 2.4, 2.7). The application and usability requirements specification documents were important during the project, bringing clear structure to the work, defining common goals for all contributors to the design process, and creating a communication medium that was equally accessible by the programmers, the designers, the usability experts, and the managers of our project.

At the end of the project, we performed two consumer evaluations aimed at establishing the acceptance of real users to a technology and travel rehearsal service such as our CVE aims to demonstrate. This is especially important in the light of concerns expressed about the effect of large-scale, internet-based CVEs on society as a whole (Chester, 1998).

### 3.4 Usability Guidelines

Due to the early state of CVE development, there is a gap between requirement listings for CVEs and actual, known design solutions. We believe that to bridge this gap CVE designers need to be provided with a systematic method that supports the decision-making process involved in coming from requirement specifications to design implementations. Part of the usability work during the COVEN project included monitoring and summarizing our work practice, which culminated in two documents describing the issues involved in designing CVEs (COVEN Del2.6, Del2.9).

### 4 Evaluation Strategy

The main aspect that makes CVEs different from single-user VEs is that they employ the network to allow multiple geographically dispersed users to interact in the same virtual world to perform collaborative tasks. CVEs need to enable and mediate communication between people, but we do not have a precise specification as to what exactly needs to be mediated to support collaboration in virtual space. If humans are to use the system effectively, we have to support them in the major acts of collaboration that are equivalent to related parts of any real-world collaboration. Thus we need three things:
An understanding of real world collaboration.
An understanding of how people actually use the CVEs that exist to date.
Tools to make the acts and roles of collaborating people explicit.

We therefore focused our efforts on clarifying sociological and psychological knowledge about small-group collaboration, specified how this real-world collaboration can and does take place in CVEs, and adapted our selected methods to analyze the collaborative aspects of CVE design and use.

4.1 Focus on Collaboration

From the literature on real-world, small-group interaction, collaboration, and verbal behavior (Robertson, 1997; Hindmarsh & Heath, 2000a, 2000b), we created a hierarchical task analysis (HTA) of collaboration, flagging tasks and subtasks essential to collaboration that would likely be unavailable in CVEs or different than similar tasks in reality. (See Tromp, 2001 for more detail on the HTA for collaboration.) See Figure 3 for the first three levels of the HTA.

Collaboration is about groups of people working together to achieve certain goals. An individual’s ability to work on collaborative tasks relies upon peripheral awareness of the others and a subtle monitoring of the activities of the other participants. Collaboration between people sharing the same workspace—be it virtual or physical—involves the on-going and seamless transition between individual and collaborative tasks. Thus, collaboration can be broken down into unfocused collaboration, in which the individual monitors the other participants’ activities without getting involved, and focused collaboration, in which individuals are working closely together. Both focused and unfocused collaboration are largely accomplished through alignment towards the focal area of activity, such as a document, where individuals coordinate their actions with others through peripheral monitoring of the others’ involvement in the activity “at hand” (Heath et al., 1995). See figure 4 and 5 for a radically simplified HTA of peripheral awareness and focused collaboration.

A collaborating group is defined as any number of persons engaged in interaction with each other in a single meeting or a series of such meetings to reach a certain goal. In a group each member receives some impression or perception of each other member distinct enough so that he can, either at the time or in later questioning, give some reaction to each of the others as an individual person, even though it be only to recall that the other was present. Taking turns in a conversation, the alternation of action and inaction, and the subsequent rotation of performance among individuals in a group is the most salient feature of group dynamics (Markel, 1975). See Table 3 for a summary of the identified interactions that take place during collaboration.

We therefore carefully created collaborative task support within our CVE and included all aspects of human-human interaction, mediated by our CVE, in our exploratory approach. We performed rigorous inspections of the interface to eliminate all major usability problems before using representative users for our experiments, and had short training sessions to make subjects familiar with the basic interface commands. Without having performed our requirements specification, the inspection, observation, and training sessions, we would have been able to measure only the most immediate interface struggles.

4.2 Longitudinal Network Trials

During the COVEN evaluation cycles were three six-month periods with networked trials lasting two to four hours, and involving four sites with between four and sixteen simultaneous users; the trials took place on a weekly or fortnightly basis. This allowed us to adopt a longitudinal research approach, an important source of data for psychologists studying groups of people. Longitudinal studies follow an individual or group of individuals over an extended period of time, with observations made at periodic intervals. After each trial, we asked the participants to answer two standard questionnaires, with one aimed at assessing the hardware, software, and technological problems encountered, and the other aimed at assessing the degree and quality of the collaboration between users. Additionally, we ran controlled, net-
Figure 3. Top levels of hierarchical task analysis of collaboration for CVEs.
worked, small-group interaction experiments with representative users, interviewing the subjects afterward and asking them to answer a standardized interaction anxiousness scale questionnaire (Leary, 1983) and our own task-related questionnaires, and we performed our observational analyses.

We investigated the network efficiency of ATM as well as ISDN and Internet in relation to data traffic as typically generated by CVEs (COVEN Del. 3.3, 3.5, 3.7). Both DIVE (Frécon et al., 2000) and dVS (Rygol, Ghee, Naughton-Green, & Harvey, 1996) were used as the underlying CVE system. There is a considerable burn-in period during networked testing, in which each distributed site has to fine-tune its network connectivity and prepare the hardware to allow it to see and hear the other effectively. This burn-in period increases slightly with the addition of each extra user machine. We started off using the relatively stable demonstration worlds that came with the applications, gradually establishing connectivity between four sites and incrementally adding more users per site. From our first inspections and hierarchical task analyses of collaboration made as part of the inspections, we created a number of representative task scenarios based on thorough analysis of all issues related to CVE interaction (COVEN Del 3.5). These scenarios were used to run small-group interaction experiments in both DIVE and dVS. Secondly, we explored collaboration processes in a virtual world especially developed for that purpose, the WhoDo game, a collaborative murder mystery environment (see Figure 6 and 7), also discussed elsewhere (Greenhalgh et al., 2000; Tromp, 2001), and we ran experiments based on a realistic user scenario in the final COVEN demonstrator, the London Travel Demonstrator (Steed, Frécon, Avataré-Növ, Pemberton, & Smith, 1999).

4.3 Inspection Methodology

We performed inspections at each stage of the design-evaluation cycle. As mentioned in subsection 3.1, most existing inspection methods were developed on and for 2D applications. We selected two methods commonly used—the cognitive walkthrough method and the heuris-

---

**Figure 4.** Simplified HTA of peripheral awareness.

**Figure 5.** Simplified HTA of focused collaboration.
tic evaluation method—to see how well they would suit inspection of CVEs. To first assess their applicability for CVE evaluation, we did not make any significant changes to either method prior to the first inspection.

Four evaluators were involved in the first inspection, each covering both the DIVE and dVS versions of the COVEN platform. Each evaluator performed the inspection in their own laboratory and independent from the others. Afterwards, we analyzed and combined the reports, and the results subsequently went to the designers of the COVEN platforms. We also gave the designers a questionnaire to assess the usefulness of the

<table>
<thead>
<tr>
<th>Social behavior</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Verbal communication</td>
<td>The exchange of audio information to establish and maintain contact between individuals engaged in focused or unfocused interaction.</td>
</tr>
<tr>
<td>Phatic communication</td>
<td>The exchange of stereotyped phrases and commonplace remarks to establish and maintain a feeling of social solidarity and well-being.</td>
</tr>
<tr>
<td>Spatial regulation</td>
<td>The arrangement of single or combined average body-size related spaces around and between people and objects, signifying temporary or permanent micro-territories, where each cultural tradition has its own micro-territorial sizes and arrangements.</td>
</tr>
<tr>
<td>Proxemic shifts</td>
<td>Patterns of interpersonal distance in face-to-face encounters accompanying and influencing changes in the topic or in the social relationship between speakers (i.e. situational shifts).</td>
</tr>
<tr>
<td>Turn taking</td>
<td>Nonverbal communication accompanying verbal communication has an important role in the understanding of social interaction and turn-taking during interaction. Common turn-taking cues are head nodding, face looking, smiling, head touching, and speaking, including simultaneous speech.</td>
</tr>
<tr>
<td>Peripheral awareness</td>
<td>A subtle monitoring of the other participants’ activities; the individual monitors the other participants’ activities without getting involved. This is largely accomplished through alignment towards the focal area of activity, such as a document.</td>
</tr>
<tr>
<td>Trust building</td>
<td>Establish and confirm one’s perceived trustworthiness as a competent collaborator, by being perceived by the other participants as acting according to the social norms.</td>
</tr>
<tr>
<td>Reciprocity</td>
<td>An individual’s ability to be simultaneously both perceiver and perceived of their own embodied actions as well as the perceiver of others’ actions.</td>
</tr>
<tr>
<td>Indexicality</td>
<td>Our ability to point at objects and locations and refer to them with indexical expressions such as <em>that</em>, <em>there</em>, and so on.</td>
</tr>
<tr>
<td>Gaze</td>
<td>Gaze direction, gaze duration, gaze patterns, gaze awareness, mutual gaze, and head turning indicate direction and type of attention given to something, aid in turn-taking, and provide communication feedback.</td>
</tr>
</tbody>
</table>
inspection report and the inspectors a questionnaire assessing the effectiveness of the method as applied to CVEs. The designers reported that the report was very useful in helping them organize and prioritize redesign of the application, with especially major and cosmetic changes being easy to identify. The usability problems proved hard to fix, however, because a new design tended to introduce new usability problems. Both designers and inspectors found that the inspection method did not sufficiently focus on redesign suggestions. The inspectors reported that the method was very useful in finding usability problems, but that the task scenarios generated from the inspection were too linear. For example, the task trees used for the inspection do not take into account the freedom of interaction with which every CVE user is endowed. This was reflected by the fact that each inspector had to add widely varying adaptations to the task trees during the inspections, depending on which tasks they performed first. Additionally, the inspectors found the heuristic evaluation less effective in finding detailed usability problems than the cognitive walkthrough. In fact, Nielsen developed the statements, or heuristics, used in the heuristic evaluation from an analysis of 256 usability problems found during the inspection of 2D applications. Although we created heuristics adapted to 3D interfaces based on these first results (Tromp, 2001), we decided to suspend the use of the heuristic evaluation until more CVE-specific heuristics are formulated from an analysis of a large set of documented CVE-specific usability problems (further discussed in the final section of this article).

Recently, Baker, Greenberg, and Gutwin (2001) published eight heuristics based on a theoretical analysis of teamwork between distance-separated groups that are as yet unvalidated. Based on our own experiences, revisions of the cognitive walkthrough method were devised for the second iteration of the COVEN platform evaluation (Del 3.5). We changed from using task-related scenarios to generic interaction scenarios in an attempt to cover the multiple tasks inherent in collaboration and the wide variety of task action choices that comes with the freedom of interaction in CVEs. This switch also allowed us to incorporate aspects of collaborative activity that were hard to capture in the task-focused scenarios. During the second and third iterations of our usability evaluations, we additionally used interaction cycles (Sutcliffe, 1995) to inspect all interactive objects and system functions that involved the users. These interaction cycles (system initiative cycle, normal 2D action cycle, normal 3D action cycle, normal cooperative 2D action cycle, normal cooperative 3D action cycle) were created based on the freedom of interaction as we observed in our evaluations and on the interaction scenarios we observed using a video camera.
cycle, goal-directed exploration cycle, and exploratory browsing cycle) are based on the work of Sutcliffe and Kaur (2000) and Kaur (1998) about usability design guidance for single-user VEs. We integrated and adapted their method with our own findings and added a cycle of collaboration to additionally inquire into the multiuser aspects specific for CVEs. (See Table 4 for the inspection cycle questions and severity ratings.) Additionally, we adapted the report forms to encourage the inspectors to add more specific and constructive redesign suggestions.

This adapted inspection method was tested by four inspectors in the second evaluation phase. The reports were used to redesign the application, give the project managers a way to make explicit the prioritization of interface fixes based on a time/benefit analysis, and give the scientists in our project an in-depth report of usability problems typical to CVEs.

Based on experience of applying the method in the second evaluation phase, we refined it once again. In the third phase of evaluation, four CVE designers were asked to apply the method themselves during the engineering phase to assess whether this would give them a better insight in the usability issues involved in design. The designers were asked to answer a questionnaire addressing the effectiveness of their utilizing the method. The CVE designers claimed to have gained educational benefits from performing the inspection; most notably, they felt they had an increased sense of the importance of feedback of actions taken in the CVE, and had a better feel for the design of task flow in general. The designers found it time consuming and difficult to assign the interaction cycles to the CVE tasks; sometimes more than one task cycle seemed to fit the task, and sometimes none did. This led to rich discussions about the nature of the tasks involved, a troubleshooting list for the manual, and better instructions.

As a result of the experience in the third evaluation phase, the inspection method was updated once more. The final version of the method has a procedure to simulate a user’s problem-solving process at each step in the user task by using a floorplan of the CVE. We also incorporated new inspection questions to check if one general task action cycle can be assumed to lead to the next correct task action cycle, on the level of object representation and placement in the total CVE space.

4.4 Observation Methodology

For our CVE observations, we used a video recording of the video-out signal of a user’s CVE interactions visible through the CVE window on their desktop computer screen. In our observations, we were additionally interested in the physical behaviors of the user through whose virtual eyes we were watching the CVE. To incorporate this information, we had a video camera aimed at the user (framing their upper body and face). This image then was recorded onto the same video recording as the video-out signal of the users’ screen in a small window in the corner of the screen using a picture-in-picture technique.

As the people in the observed group interact with each other, the observer breaks their behavior down into the smallest meaningful units she can distinguish and records the scores by noting

- the time at which the behavior occurs,
- the number of the person exhibiting the behavior,
- the proper category of observed behavior, and
- the number of the recipient person or object.

The observer follows the interaction continuously in this microscopic manner, attempting to keep the scores in the sequence in which they occur and to omit no item of behavior.

Interactions in a group take place over time. The interactions show uniformities, repetitions, and tendencies to occur in certain sequences. For instance, at the beginning of a meeting people typically greet each other, during a meeting they listen to each other, and during the end of a meeting they say their goodbyes. These interactions can be classified into categories based on their similarities and differences. The observer classifies every item of behavior she can observe and interpret based on the list of categories. The classification the observer makes hinges on the inference that the observed behavior has function(s), either by intent or effect. The sequences of the categorized and recorded interactions can subsequently be analyzed to reveal patterns in behavior.
<table>
<thead>
<tr>
<th>System initiative</th>
<th>Normal task action 2D</th>
<th>Normal task action 3D</th>
<th>Goal-directed exploration</th>
<th>Exploratory browsing</th>
<th>Collaboration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is it clear to the user that the system has taken control?</td>
<td>Will the users be trying to produce whatever effect the action has?</td>
<td>Can the user form or remember the task goal?</td>
<td>Does the user know where to start looking?</td>
<td>Can the user determine a pathway for movement?</td>
<td>Can the user locate the other user(s)?</td>
</tr>
<tr>
<td>Can the user resume control at any point and is the appropriate action clear?</td>
<td>Will users be able to notice that the correct action is available?</td>
<td>Can the user specify an intention of what to do?</td>
<td>Can the user determine a pathway towards the search target?</td>
<td>Can the user execute movement and navigation actions?</td>
<td>Can the user recognize the identity of the other user(s), tell the other users apart?</td>
</tr>
<tr>
<td>Are the effects of system actions visible and recognizable?</td>
<td>Once a user finds the correct action at the interface, will they know that it is the right one for the effects they are trying to produce?</td>
<td>Are the objects or part of the environment necessary to carry out the task-action (user’s new intentions) visible?</td>
<td>Can the user execute movement and navigation actions?</td>
<td>Can the user recognize objects in the environment?</td>
<td>Are the communication channels between the users effective?</td>
</tr>
<tr>
<td>Are the system actions interpretable?</td>
<td>After the action is taken, will users understand the feedback they get?</td>
<td>Can the objects necessary for the task action be located?</td>
<td>Can the user recognize the search target?</td>
<td>Can the user interpret identity, role, and behaviors of objects?</td>
<td>Are the actions of the other user(s) visible and recognizable?</td>
</tr>
<tr>
<td>Is the end of the system action clear?</td>
<td>Is there an obvious next action to perform for the user, now that this task has ended?</td>
<td>Can the users approach and orient themselves to the objects so the necessary action can be carried out?</td>
<td>Can the users approach and orient themselves to the objects so the necessary action can be carried out?</td>
<td>Can the user remember important objects or locations?</td>
<td>Can the user act on a shared object while keeping the other user(s) in view?</td>
</tr>
<tr>
<td>Is there an obvious next action to perform for the user, now that this task has ended?</td>
<td></td>
<td>Can the user decide what action to take and how?</td>
<td></td>
<td>Can the user decide what action to take and how?</td>
<td>Can the user easily switch views between the shared object, other locations/objects of interest, and the other user(s)?</td>
</tr>
</tbody>
</table>

**Severity Ratings**

0. I don’t agree that this is a usability problem at all.
1. Cosmetic problem only: need not be fixed unless extra time is available on project.
2. Minor usability problem: fixing this should be given low priority.
3. Major usability problem: important to fix, so should be given high priority.
4. Usability catastrophe: imperative to fix this before users test the system.
Seven basic interaction categories were created: communicate, manipulate, navigate, position, scan, gesture, and external. These categories were developed in the following way. First, an observation of a recording of a representative CVE interaction session was made, noting down all single units of behaviors. This list of observed behaviors was grouped and condensed into 26 categories. A focus group (Jordan, 1998) of six participants (consisting of psychologists, sociologists, and ethnographers) assessed the categories based on their professional expertise. These categories were subsequently tested on another recording of a representative CVE interaction session. The frequency and fit of the categories was analyzed and further reduced to seven basic categories. These seven categories were assessed by another focus group of six participants (consisting of VR researchers, evaluators, and developers) by applying them to another representative CVE interaction recording, followed by a lively discussion of the merits of the technique afterwards.

5 Results

5.1 Longitudinal Network Trial Results

We analyzed the results from the longitudinal questionnaire that was filled in by the participants after each network trial and interpreted their replies with regards to the social interactions that take place during collaboration as evoked in subsection 4.1 and Table 3. Due to space constraints, we present only a summary of these results here, but a longer discussion including quotes from the questionnaires can be found in Tromp (2001). It has to be noted that the participants in our experiments usually successfully performed their tasks. However, the quality of their collaboration experience was often in danger due to the limitations of the technology and the frustrations this caused them. Confident, competent participants were seen to be reduced to shy, inhibited, and quietly angry ‘collaborators.’

Verbal communication is hampered by breakups in the audio transmissions due to network congestion. Participants have problems making themselves understood, understanding other speakers, and following the flow of the discourse among each other. This is especially problematic when speakers are nonnative to the common language used during the collaboration. Participants can feel very uncomfortable through this lack of understanding and have been observed to give up on the collaboration process. Participants have to speak much more clearly, slowly, and louder than normal, which can be problematic, especially in shared offices.

Phatic communication, especially important between people who do not know each other well, is generally not well supported in CVEs and participants frequently expressed feeling inhibited in their interactions due to the absence of means to express this type of communication. Typical problems were found with initiating, continuing, and ending interactions in a polite manner.

Spatial regulation is not as easy to adhere to in the CVE as in the real world due to difficulties with navigation and fine-tuned positioning. Participants frequently unintentionally obstructed each other’s view with their virtual embodiments, and navigated through each other’s embodiments, walls, and objects, which was sometimes perceived as rude. There was an indication that, even if a CVE room was a realistic replica of real room, the virtual room was perceived as too small or the virtual embodiments too large. This is possibly due to the small field of view.

Proxemic shifts were performed by expert and novice participants; however, both types of users had problems navigating where they wanted to be, especially when they all tried to view the same small object at the same time. This suggests that they are not aware of the precise dimensions of their size and the effect of this in terms of distance to the other participants.

Coverbal behavior is important in negotiating turn-taking in conversations and interactions, but most of the commonly used actions are not available in CVEs. The lack of these cues made participants feel uncomfortable and isolated and contributed to breakdown in the communications.

Peripheral awareness in CVEs is limited to perception of movement in the field of view and any surround sound cues picked up from the wider surroundings. Participants often panned their view 360° to update their knowledge of their surroundings, or felt the need to
move backwards to increase their field of view. Peripheral awareness of the real environment while engaged in the CVE is minimal due to the fact that CVE users wear a headset to hear and speak to the other CVE participants. Peripheral awareness of the CVE while engaged in the real environments is limited to visual only when the headset is taken off. These limitations make switching attention between the two environments more difficult.

Building trust is a slow process in CVE interaction. Participants cannot easily establish whether inaction or the absence of response of other participants should be attributed to a breakdown in their connectivity, a temporary absence of the participant from the CVE window, or a lack of interest in being engaged in the activity at hand. Especially when participants do not know each other well or are not overly familiar with the CVE as a medium, they are quick to interpret lack of reaction as a lack of respect for the social norms of interaction, and thus the trustworthiness of the nonresponding participant.

Reciprocity was difficult to establish for participants mostly due to the small field of view but also because it is hard to tell where precisely the other participant is looking. Typical problems were with being able to interact with distant objects, causing the actor to be out of the view of the participants near the object. Participants tried to overcome these problems by giving a running commentary of their actions to the other participants and describing their location to the others.

Indexicality is similarly difficult to achieve, as participants cannot see the direction of each other’s gaze and might not be able to see each other point and see the object being pointed at in the same view. It was difficult for the participants to follow directions of other users, and their references had to be sufficiently detailed and slow for the other participants to be able to catch up with due to delays in network traffic between the geographically dispersed participants.

5.2 Inspection Results

The inspection provided a large number of apparent problems with the interface that are not only applicable to the COVEN platforms but to CVE technologies in general. We categorized the problems according to their root technical cause, and we found that they could be broken down into three rough categories.

- System problems including lack of functionality, performance, and display quality.
- Interface problems that concern the actions of navigating, and picking of objects.
- Application-specific problems concerning the actual actions and meaning of objects within the environment.

System problems are not often apparent immediately but can result in less than optimum strategies having to be taken in interface and environment design. For example, perhaps the most serious problem encountered was navigating through the doors in the environment. Although ostensibly an application problem (because the door is simply an object with a behavior), the problems were exacerbated by three factors. Firstly, the door opening caused the scene on the other side of the door to be loaded, which caused a short stall during the time that the relevant information is loaded. Secondly, the script to control the door runs on a central server, so the remote clients have to send a signal that delays the door opening. And, thirdly, synchronization problems between the client view and the collision detection (which also runs on a central server) meant that, although the participant might see an open door, they could not enter it because they had not received the correct response from the collision system yet. These kinds of lag and resulting consistency problems have led to a large open research area as addressed by Vaghi (2002).

Interface problems arise because of the nature of the desktop display where input devices with few degrees of freedom (mouse and keyboard) were used to perform six-degree-of-freedom tasks and where there is a mix of 3D with traditional 2D WIMP interface. This can cause problems with interface modalities, such as there being several control modes depending on which mouse and modifier buttons are depressed, and the actual mapping of user motions into three-dimensional transformations. These are aspects of the interface that are plainly not
obvious to a na"ıve user, and they are a continuous struggle to the expert user. Interface control problems thus tend to be pervasive across evaluations because they are constrained by the capabilities of the underlying VR toolkit. We classified interface problems into the following categories.

- **Modality and mapping**: There often were several control modes, depending on which mouse and modifier buttons are depressed, each with a potentially nonintuitive mapping to 3D.
- **2D Disturbing 3D**: 2D menus pop up over the CVE window, thus disrupting interaction in the 3D interface.
- **Unsupported 3D actions**: Many actions that could be performed directly inside the 3D space are relegated to 2D menus.
- **Lack of 3D feedback**: Actions performed on 2D menus that affect 3D objects often do not give recognizable feedback in the 3D environment.
- **Two object tasks**: The 2.5D “drag and drop” mechanism is not supported in CVEs, making selection and manipulation an arduous task.

We believe that these interface problems have been neglected to date and deserve greater attention because they are common to all desktop-style user interfaces. Based on our findings we predict the need for a large research effort into 3D selection, manipulation, and feedback; better support for 3D actions inside the CVE; and implementation of 2D menu interaction inside the CVE to increase visibility to all users. We summarize these research topics in subsection 6.3.

Application problems are more-general problems with the participant’s understanding of the purpose of the application components. Making spaces and objects realistic endows them with affordances that cannot always be satisfied (Gaver, 1992; Wilson, Eastgate, & D’Cruz, 2002). There is a balance to be struck between making the objects realistic in appearance so that they may be recognized, and making functionality apparent to the user (Tromp & Fraser, 1998). Essentially, the conflict is over resolving how to communicate to the freely moving CVE participants what objects portray actions within where in the environment, and when, in what order, and how to perform them. These application problems can be summarized as follows.

- The perceptual affordances of CVE design are insufficiently exploited, leaving users confused about the type of actions available.
- The sequential affordances of CVE design are insufficiently exploited, leaving users confused about the order in which actions should be performed.
- The narrative affordances of CVE design are insufficiently exploited, leaving users confused about the purpose of the environment and the objects in it.

Ideally, all possible affordances that objects could provide should be covered, even if it is just feedback about an inability to perform an action. The reality of the situation is that, within the size constraints of the environment or within the time constraints to build the environment, not all affordances can be catered for. We perceived a need for more structure in the design of the interactions. The design notion of narrative affordances refers to the observed need for a narrative structure throughout the CVE space, a guidance for the sequence of tasks and object interactions that has been designed deliberately (Tromp, 2001). Our final design document (COVEN, Del 2.9) tries to take these issues into account. (See section 6.2).

### 5.3 Observation Results

Based on our understanding of real-world collaboration, we expected to observe shifts in viewpoint between collaborators and shared objects, and communication about the task at hand. For the observational analysis reported here, we analyzed three separate sections: the beginning, middle, and end of a total collaboration session. Although this data set is small, the results are similar to those found with a larger data set reported by Tromp (2001). Our data set consists of 148 observations, recorded during a total of three separate minutes of interactions between six users, watching the CVE through the eyes of one user. Figure 8 and Table 5 provide an overview and breakdown of the frequencies with which the observed collaborative acts occurred.

Shifts in viewpoint, accomplished by fine-tuned navi-
gation and scanning acts, occur with a frequency of 63.5%, whereas communication acts occur 31.8% of the time, and the residue is made up of checks on windows external to the CVE window (4.1%) and gesturing (0.7%). It has to be noted that the interface allows for only one type of gesture (pointing), which is probably why the percentage of observed gesture acts is as low as it is.

We were interested in finding systematic relations between behaviors, and Table 6 shows the frequencies of transition from one behavior to another. The total (column T) indicates the number of adjacent code pairs for each category, and each cell indicates how often each transition occurred (for example, Communicate was followed by a Scan on six occasions). We compared and analyzed the beginning, middle, and end of a collaboration session (WhoDo Experiment I, 30-9-98). For each column (Communicate, Manipulate, and so on) are three subcolumns containing the totals of each analyzed section (Subcolumn 1: Begin; Subcolumn 2: Middle; Subcolumn 3: End). This allowed us to compare any differences in the frequencies of adjacent acts as a complete collaboration progresses over time.

The frequency with which a communication is followed by another communication increases as the collaboration session progresses. Navigation has a higher frequency during the beginning and middle parts of the collaboration than during the end. Navigation and scanning acts occur mostly during the beginning stage of the collaboration (when the stability of the CVE application has to be checked). Furthermore, communications are sometimes followed by a scan, scanning often precedes navigation, and navigation is often followed by a scan. A scan is more likely to be followed by a communication during the beginning of the collaboration, and scanning is a frequent activity during the total collaboration session, as is navigation and communication (although the high frequency of the latter two are to be expected).

Scanning occurs during the total meeting and has the highest frequency during the middle part of the collaboration. Communication acts during collaboration are supported by more than twice the amount of meta-collaboration acts. Navigation often involves many fine-tuned positioning acts to encompass the most advantageous viewpoint for collaboration.

In total, 21 different types of navigation acts have been observed. The two most frequently observed navigation acts are moving backwards to increase the field of view (25.9%) and moving forwards to make the collaboration circle smaller (16.7%), the remaining nineteen navigational acts account for the residue. Subjects can be seen trying to “back up” to increase their field of view, trying to encompass as many objects and subjects as possible in their view at the same time. Without collision detection, this results in them “falling out of the room,” where they back up to the point of going through a wall. With collision detection, it can be difficult to get all relevant items in one view. Indeed, one subject in the post-test interview complained that the room was “too small.” One common solution for this problem is to use an “out-of-body” view, but navigation in this mode is more complicated.

Nine different types of scanning have been observed. The two most frequently observed scanning acts are scanning by turning the view more to the left (27.5%) and scanning by turning the view more to the right (32.5%). Subjects in the experiments can be seen repeatedly making a sweeping move from object to speaker, and back to object, and so on, during collaboration. They can also be observed having trouble making this sweeping movement smoothly, repeatedly overshooting their goal.
Eleven different types of communication acts have been observed. The two most frequently observed communication acts are communications about the task at hand (29.8%) and communications about the collaboration itself (21.3%). Slightly less than half (46.8%) of all observed communication acts are concerned with verifying having heard, seen, understood, or having been heard, seen, or understood by the other participants. Of this type of communication act (to verify), the two most often observed were communication to verify having heard (14.9% of total communication acts) and communication to verify being personally present (8.5% of total communication acts). The residue of the total (2.1%) is made up of text chat communications, often used for double-checking something or to exchange social amenities. Thus, of all observed communications, only 29.8% are directly concerned with the collaboration task at hand, whereas 68.1% are directly concerned with keeping the collaboration going (meta-collaboration).

To summarize, the collaboration process was found to be insufficiently supported in the CVE used for the experiments, creating problems for the users when trying to build up and maintain an understanding of who does what, when, aimed at whom, with what results on what, and for whom. General patterns of CVE user collaboration acts lie in the realm of continuous small adjustments in viewpoint, triggered by the happenings in the shared space. Subjects in our experiments had three main problems:

- keeping the referenced shared object and other participants in the same view,
identifying the referenced object and also the participant who is referring to the object, and
• monitoring the activities of the other participants while acting or navigating themselves (breakdown of peripheral awareness).

Many of these CVE user activities can be partly or fully automated, which we hypothesize will help to lower the cognitive overhead of controlling the virtual embodiment, so that users can get on with their collaborative task more effectively. Our design guidelines try to take these issues into account. (See subsection 6.2.)

5.4 Consumer Evaluation Results

One consumer evaluation involved a common user PC placed at a travel agency and used by a group of customers between the ages of 11 to 55, with differing backgrounds, thus testing the application in a typical usage situation. Usability data were collected afterwards by means of a questionnaire that focused on how the users experienced the application, and if they were satisfied with the innovative way of making holiday reservations after having visited its virtual model. Almost everybody in the subject group expressed a willingness to use such a system to choose and book holiday destinations. As the number of Internet users grows, the demand for such applications may grow, and travel agencies may be forced to change the way they interact with customers.

The second consumer evaluation involved an experimental task in a laboratory setting, based on a model of the total travel experience (from anticipating a journey, to the actual travel, to the memories afterwards), focusing on hypotheses about the added benefits of VR technology. Although only one experiment took place in this context, the results indicate that consumers greatly enjoyed using such an application, providing them with an added “fun factor” in travel rehearsal. It also pointed out that, for certain tasks inherent in the total travel experience (such as itinerary planning), the application used in standalone mode is perfectly acceptable for users, and for other tasks (such as being able to organize meetings in the CVE) they would be quite happy to use Internet-based connections. Generally, the subjects in this experiment asked for information about more holiday destinations of a rich, realistic, and detailed fashion, indicating a large market for applications such as our demonstrator.

6 Lessons Learned and Future Work

6.1 Cost/Benefit

Despite problems of applying established inspection methods, we found that inspections could be a quick way to improve and evaluate a CVE application. Without having performed our inspections on each subsequent iteration of the system, we would not have been able to perform our networked experiments in a smooth and controlled fashion because too many bugs would have remained. We found that performing an inspection took one day, provided that the requirements specification is available to the inspector(s) and provided that the designer and programmer of the application are actively involved in the task together with the usability expert. Writing the inspection report takes another couple of days, depending on the number of differently focused reports requested (such as reports for the designers, usability engineers, managers, and researchers). The inspection reports provided us with a rich source of reference for redesign and discussion during the development phases. (see Table 7.)

The networked trials and longitudinal observations took place in three phases, and the beginning of each phase consisted of a considerable technical effort to establish a stable networking infrastructure. The network trials gave us important feedback about user adaptation over time, and helped us in our task of highlighting particular usability issues as being key. CVEs place a high demand on applications and network infrastructure, and we found that the networked trials themselves were very difficult and time consuming to run and needed careful orchestration.

The observations illustrate that studies based on time-and-motion concepts and methods can yield extremely valuable information, revealing factors that were previously unknown. The goal of this analysis was to collect quantifiable data with the added insights of qualitative
data about collaboration. Although this could have been pursued using a combination of network traffic analysis and ethnographic observation, no quantifiable data would have been gathered about the nature and types of navigational and communication acts that users perform in a CVE. Adopting this approach allowed us to make inferences that build a greater understanding of the atomic acts involved in collaboration mediated by a CVE.

Our consumer evaluations were preliminary in that we tested the reception of our application, which was never intended as a finished end product, but instead as a demonstrator of future technology. The evaluations indicated consumer interest and acceptance of the technology, and a demand for more such applications from both customers and salespeople of travel information. This guided us to advise the EU in our final reports that commercializing CVEs for virtual travel information involves the need to develop new interaction paradigms both for the information providers and the customers.

### 6.2 Usability Guidelines

The COVEN usability guidelines documents (COVEN Del 2.6, Del2.9) focused on objective means to specify the perceptual message that is conveyed by the CVE to the users. Our document is the description of a method with which a design team can decide which are the most essential human needs to support, and how to negotiate simplifications of representations and interaction. We attempted to provide such means by creating a functional breakdown of CVE usability requirements. These deliverables do not list all CVE usability requirements that are known to date, nor is it an exhaustive list of heuristics or guidelines to build CVEs for usability. Although we agree that such information would be extremely useful (c.f. Carr & England, 1995; Steed & Tromp, 1998; Hix et al., 1999), much more work is required before such a document can be written.

A general problem area for CVE design that we found was with the flow of interaction in 3D space. Because of the freedom of navigation and interaction typical for CVEs, it is difficult to guide users through the CVE interface towards their goals. When functionality is not obvious, dealing with the interface creates an unnecessarily large cognitive overhead, slowing the user down and frustrating them. Freedom of navigation and interaction makes it difficult to predict what actions users will take, and in what order they will perform these actions. Users have been shown to struggle with finding the right order in which to perform actions, with finding their way through the environment, and with navigating into precise positions (Kaur, 1998; Tromp & Snowdon, 1997). Simplifications of the representations due to performance constraints make it difficult for users to predict which operations are available and which are not, and users have been shown to struggle with the interface for these reasons (Kaur, 1998; Steed & Tromp, 1998). These usability problems have led us to formulate a CVE usability design method that uses the notion of narrative affordances, a design notion that can help to continually guide users through their CVE interactions (COVEN Del. 2.9; Tromp, 2001)—a conceptual model of generic CVE space, which separates the CVE experience into architectural space, semantic space, and social space (Figure 9)—and we formulated design guidelines to complement this functional breakdown. Combined, this method aims to assist CVE designers in specifying the perceptual message for the

<table>
<thead>
<tr>
<th>Cycle function</th>
<th>Severity 4</th>
<th>Severity 3</th>
<th>Severity 2</th>
<th>Severity 1</th>
<th>Totals</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inspection 1 “Initial Demonstrator”</td>
<td>10</td>
<td>1</td>
<td>5</td>
<td>10</td>
<td>26</td>
</tr>
<tr>
<td>Inspection 2 “Online Demonstrator”</td>
<td>7</td>
<td>3</td>
<td>5</td>
<td>3</td>
<td>18</td>
</tr>
<tr>
<td>Inspection 3 “Final Demonstrator”</td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>7</td>
</tr>
</tbody>
</table>

Table 7. Overview of the Changes in Results of the Inspections
CVE users, based on the available knowledge of “best design” in the absence of more-specific guidelines.

6.3 Future Research Directions

The COVEN project has come to an end, but the usability research and development areas that were initiated during the project will continue. A summary of future research directions can be presented as follows.

- Performance of a meta-analysis of all types of usability problems found with single-user and multiuser VEs to date, providing a good start for the development of CVE design and evaluation heuristics.
- Development of a standard CVE inspection method. Further work is required to assess whether the six interacting cycles that have been proposed here are exhaustive for the inspection of the complete repertoire of interactions available in CVEs.
- More explicit theory for design of flow of interaction for CVEs, creating a structured overview of narrative affordances that will improve situational awareness and guide the freely moving users, communicating the available interactions and the best order in which they can be performed.
- Experimentation and evaluation of design solutions for smooth and flexible 3D selection, manipulation and feedback, better support of 3D actions inside the CVE, and implementation of 2D menu interaction inside the CVE to increase visibility to all users.

6.4 Conclusions

The research presented here has generated four products that aim to improve the design and evaluation process of CVE technology.

- Observation method, which is generally applicable to CVE user observation and analysis of collaborative behaviors. The categories used for the analysis can be adjusted to refocus the topics under observation. It was shown how we developed and applied the method, and references are provided to more-detailed descriptions of how to use the method.
- Inspection method, which is generally applicable to CVE evaluation. This method can be used during all design stages, assessing the usability of the design for each interactive element in the total task of a CVE user. We have shown how this method was developed to better support the nature of CVE tasks.
- Usability design method, which is generally applicable to CVE design. This can be used during the specification of the actual look and feel of the CVE spaces, CVE objects, and CVE interactions. It distills earlier COVEN work on usability guidelines and provides a method to generate a narrative to guide the user without constraining them unduly.
• **Hierarchical task analysis of collaboration**, which is generally applicable to defining requirements specifications that should support collaborative behavioural needs.

**Acknowledgments**

This work has been funded by the European Commission, ACTS Project COVEN, N. AC040, and finalized during the IST Project VIEW of the Future, IST-2000-26089, the UK IRC project EQUATOR, funded by the EPSRC, and the IRMA project, funded by the European Commission, IMS Project 97007. Many people contributed to the work reported here. We cannot acknowledge all of them by name, but we would like to specifically thank Prof. Alistair Sutcliffe, Dr. Kulwinder Kaur Deol, Dr. Veronique Normand, Dr. Dave Lloyd, and the anonymous reviewers, whose helpful and insightful comments have strengthened the paper enormously.

All COVEN documentation mentioned in this paper, including detailed evaluation reports, manuals for method application, example and results with COVEN applications and design guideline documents, are available via the COVEN Web site: http://coven.lancs.ac.uk.

**References**


Steed, A., & Tromp, J. G. (1998). Experiences with the evalu-


