A Framework for Design and Evaluation of Collaborative Virtual Environments

Nguyen-Thong Dang(†), Céline Chatelain(†), Jean-Marie Pergandi(†), Daniel Mestre(†)

(*) ISM – Institut des Sciences du Mouvement, UMR 6233 CNRS & Université de la Méditerranée
(**) CLARTE – Centre Lavallois de Ressources Technologiques

ABSTRACT

The paper presents an initial framework for studying issues in the design and evaluation of Collaborative Virtual Environments (CVEs) based on immersive projection technology systems. The framework consists of a software platform, developed using Virtool® software suite, which supports collaborative work among collaborators in different immersive systems (CAVE®-like environment, Workbench, etc.), and usability inspection of each iteration of the platform. The objective of this setting for the framework was to gain insights on both technological aspects regarding the development of such a platform and human factors issues on collaborative work within Virtual Environments; and more importantly on the interrelation between technological and human factors aspects for building usable Collaborative Virtual Environments. A live demonstration of the platform connecting two CAVE®-like environments situated in geographically distant places was organized to collect feedback from participants. The demonstration provided the users with different interaction metaphors addressing the three main characteristics of collaborative work through Virtual Environments: awareness of others, context sharing, and negotiation and communication. Details about the platform, results of an informal observational study though the live demonstration and lessons learned from conducting such a framework are presented in the paper.

KEYWORDS: Collaborative Virtual Environments, Design and Evaluation, Usability.

INDEX TERMS: H.5.2 [Information Interfaces and Presentation]: User Interfaces – Evaluation/methodology, Interaction styles, Theory and methods, User-centred design

1 INTRODUCTION

Immersive systems and Virtual Reality (VR) technologies, which become more and more accessible, provide tools for different actors in the design sector to analyze, evaluate, and validate their products through three-dimensional (3D) models before any physical prototyping phase. Yet, these sectors bring in different building trade around 3D models where their actors are more and more geographically farther apart for business strategy reason (subsidiary, subcontractor…). In this context, the need for an environment supporting collaborative work on 3D models among geographically separated collaborators emerges. This issue is the focus of an active research on Collaborative Virtual Environments (CVEs), a subfield of Computer Supported Collaborative Work (CSCW) which addresses “how collaborative activities and their coordination can be supported by means of computer systems”[4]. In fact, CVEs deal with the problem of real-time sharing workspace among multiple users at geographically different sites being connected to each other via a communication network. The shared workspace can be related either to two-dimensional (2D) user interfaces or to a spatial context through 3D user interfaces or Virtual Environments (VEs). The working interfaces have impact on the shared workspace and thus on the way users collaborate with each other in a common working environment.

Despite the increasing advances of VR technologies, the lack of ergonomics-based guidelines for collaborative work within VEs, usability in particular, is still one of the challenges to overcome so as to define usable CVEs. In general, “research on the usability of CVEs is still at an early stage”, as stated by Schroeder et al. [14]. Different efforts has been carried out for understanding about design, usability for CVEs (e.g., the COVEN (COllaborative Virtual ENvironments) project [13]); however, similar to the case of VEs, there are still few reported contributions to the structured design and evaluation of Collaborative Virtual Environments (as stressed by Economou and Pettifer [8] and by Tromp et al. [17]). In addition, few CVEs are available for studying collaborators’ behaviours within VEs. The lack of available collaborative platforms, as stressed by Schroeder et al. [14], for studying usability issues of collaborative work within CVEs introduces even more difficulties to get to usable CVEs.

In this context, the present paper proposes a framework for studying issues in the design and evaluation of CVEs. The framework involves the development of a software platform which supports collaborative work among collaborators at different CAVE®-like environments (CAVE is a recursive acronym that stands for CAVE Automatic Virtual Environment, which was introduced by Cruz-Neira et al. [6]) in combination with usability inspection of each iteration of the platform. The objective of this multidisciplinary setting of the framework was to gain insights on both technology and human factors aspects regarding such a platform for collaborative work within VEs; also on the interrelation between technological and human factors aspects for building usable CVEs.

The remainder of the paper is organized as follows. The next section provides an account of related work on design and evaluation of interfaces for collaborative work in virtual environments (VEs), followed by a brief discussion on the approach we followed. “The Platform” section describes the methodology we used to design the collaborative platform and also the state-of-the-art at the end of our design process. The
investigating the human behavioural aspects that affect projects on design and usability for CVEs. COVEN focused on consumer evaluations to assess social acceptability and utility of the demonstrators, and usability guidelines were continuously developed and evaluated for usability inspection of each iteration of the project. Therefore, in this section, related work on multidisciplinary setting for studying issues on the design and evaluation for CVEs, in particular in CVE-like environments, will be discussed.

Before going into detailed related work in a multidisciplinary setting regarding CVEs, it is worth briefly presenting some existing collaborative platforms. Among the existing platforms, DIVE (abbreviation of “Distributed Interactive Virtual Environment”), developed at the Swedish Institute of Computer Science (SICS), can be considered as one of the most mature multi-user virtual environments. It is being used worldwide for developing VEs applications [5]. DIVE, which is a tool kit for building distributed VR applications in a heterogeneous network environment, allows a number of users and applications to interact and communicate through a virtual environment in real-time. More technical details of DIVE can be found in [9]. Another platform supporting CVEs is MASSIVE (abbreviation of “Model, Architecture and System for Spatial Interaction in Virtual Environments”), developed at the University of Nottingham in UK (the latest version of MASSIVE is MASSIVE-3). This platform allows multiple users to communicate audio, graphics, and text media over networks. MASSIVE focuses on scale (i.e. simultaneously supporting as many users as possible) and heterogeneity (i.e. supporting interaction between users with different hardware platforms, different user interfaces). More technical details of MASSIVE can be found in [10]. Yet, a comprehensive review of other existing platforms and applications can be found in [5].

Regarding related work in a multidisciplinary setting, the COVEN project [13] is probably one of the most important projects on design and usability for CVEs. COVEN focused on investigating the human behavioural aspects that affect performance and satisfaction in VEs. The most interesting aspect of COVEN is that this project combined studies of issues on both technology and human factors sides though different studies of user behaviour and computational demands throughout the project. A relevant point regarding the framework of COVEN development and evaluation is the use of an iterative process for usability inspections of the project demonstrators. In a more detailed manner, the framework composed of 4 main threads of works including usability inspection, observation evaluations, consumer evaluations and usability guidelines. In fact, in addition to usability inspections of each iteration of the project demonstrators, observational evaluations were carried out to explore and understand main issues regarding collaborative work (i.e. presence, co-presence, interaction and collaboration), consumer evaluations to access social acceptability and utility of the demonstrators, and usability guidelines were continuously prepared throughout the project for future developments of CVEs. The user-centred framework composed of different activities as adopted by COVEN was proven necessary to build guidelines for the structured design and evaluation of CVEs.

We will now focus on studies on collaborative work within CVE-like environments, which relate more closely to the present study. Actually, few studies on CVEs have been carried out in these environments. Besides, existing studies have worked on varied topics; perhaps due to the fact that CVEs contain complex subjects involving different issues regarding both technology and human factors. We detail two studies that we found relevant in the context of the present study.

The first study on collaboration within CVEs, based on the DIVE platform, was carried out by Mortensen et al. [12]. This study investigated the feasibility of collaborative work in physically remote virtual environments: a CVE-like environment (a four-sided Trimension ReaCTor) at University College London (UCL) in UK and a head-tracked head-mounted display at University of North Carolina Chapel Hill (UNC-CH) in the US. In the study, each pair of participants (a confederate at UCL in UK, an experimental subject at UNC-CH in the US) who did not know each other had to collaborate to lift an object together and move it to another place. An avatar was used to represent the remote participant on each side. This study examined the relationship between several variables such as co-presence (i.e., the sense of being together with another person) and self-assessment of task performance of participants by way of a questionnaire. The participants’ feedback suggested that the co-presence was significantly and positively correlated with the self-assessment of task performance. It’s worth noting that in this study, the experimental subjects, which were the ones at UNC-CH, were wearing a head-tracked head-mounted display for the collaborative task, instead of being within a CVE-like environment.

Regarding studies on the collaborative work which involve CVE-like environments, we found an investigation on long-term use of CVEs, carried out by Steed et al. [15]. A five-sided TAN VR-Cube™ at Chalmers University in Sweden and a four-sided Trimension ReaCTor at UCL in UK were used in this study. Each pair of subjects carried out a series of different tasks over a long period of time (more than 210 minutes in average). An avatar with jointed arm animation, but no facial animation was used to represent the remote participant. One of the aims of this study was to examine whether the pair of subject who did not know each other would collaborate differently from the pair who know each other well. Other aims of this study were to examine how the participants’ experience changed over time, and to compare their experience during different tasks. The participants’ feedback was collected by way of a questionnaire, interviews and observations. Overall, the study shows that users could collaborate effectively over an extended period of time, that there was no big difference between the two types of pairs: friends and strangers. One interesting result concerns the avatar; some subjects found negotiating tasks harder because of the absence of facial expressions of avatar. The DIVE platform was also involved in this study.

Our approach

Inpired by the framework of development and evaluation for CVEs employed in the COVEN project, we pursued this research direction, focusing on collaborative work within CVE-like environments in particular. We worked on a collaborative platform for CVE-like environments, developed by using Virttools software suite. In addition, we used an iterative process for usability inspection of a demonstrator which was built from the platform. In the next section, we describe the framework we used to design and evaluation of the collaborative platform. We also describe the platform we achieved at the end of our design process.
3 The Collaborative Platform

3.1 Framework for design and evaluation of the platform

An iterative process was adopted to the design and evaluation of our collaborative virtual environment platform. Starting from the initial design of a demonstrator of the platform, the evaluation has been carried out at two different levels as follows.

- Inspection-based evaluation of the platform to uncover usability problems and design flaws. Those findings were then analyzed so as to suggest design improvements for the platform. A group of evaluators was involved in the inspection. The evaluators followed a set of guidelines to inspect the interface of collaborative work provided by the platform.
- Observational evaluations of participants’ behavior and collection of participant’s feedback though demonstration of the platform, so as to understand the behavior characteristic of users in CVEs and to reveal the design flaws from the point of view of novice users.

Feedback from each evaluation is subjected to improvements of the platform at both system and interface levels. The improvements of the platform resulted in an improved design which is subjected for successive evaluation and so on. The next two sections will detail the technological aspect of the platform and the usability inspection used for evaluation of the platform.

3.2 The Platform

The platform was developed based on the Virtools software suite, a complete development platform with different built-in libraries for interactive 3D content creation. Details of how to build a VR application based on Virtools would be out of the scope of the paper; however, we provide in following subsections some basic knowledge about Virtools so as to facilitate the understanding of our collaborative platform’s architecture.

3.2.1 Architecture of the Platform

In general, a VR application involves a virtual world which composes of different elements (e.g. 3D object, camera, etc.) with their own behaviour. All elements in the virtual world can interact with each other so as to provide the 3D and interactive content of a VR application. In a similar way, the virtual world in Virtools, known as composition, is an arrangement of one or more elements and their associated behaviours. In Virtools, behaviour is expressed as a script which is a set of interconnected Building Blocks (BBs). Each Building Block corresponds to a function, i.e. a ready-to-use solution to a known task. To facilitate the development of VR application, Virtools software suite provides different modules. Each module, which composes of building blocks, supports a specific aspect of VR application’s development. Developing a Virtools-based VR application involves in creating a composition based on these existing modules.

The platform is developed based on two Virtools modules: MultiUser Pack and VR Pack. MultiUser Pack provides a server (which provides supports for either client-server or peer-to-peer architecture) and a library of BBs supporting multi-user applications. The client-server architecture was used in our platform. VR Pack permits a Virtools composition to work on any VR platform thanks to VRNR (Virtual Reality Normalized Resources), a layer of abstraction allowing the portability of Virtools composition on different VR platforms (a VR platform in general terms composes of a set of input devices, a set of output devices and a tracking system) VRNR allows automatically building a composition based on the modules and parameters defined in a configuration file. The fact that VRNR is customized through configuration files improves the deployment of Virtools-based VR application on different types of VR platform. For example, the administrator of a VR platform can configure a composition according to his input devices, output devices and tracking system, without modifying the composition itself.

The platform is developed based on VRNR. The objective of our platform is to provide another layer of abstraction so as to facilitate and support collaborative applications. We call this abstraction level the Collaborative VRNR based level (CVRNR) (cf. Figure 2). Using VRNR, CVRNR allows automatically building compositions which work in the special context of CVEs.

3.2.2 Collaborative Interaction Protocol

Regarding the details of the platform, a collaborative interaction protocol was introduced to guarantee special demands of an environment supporting collaborative work. In more detail, this interaction protocol was designed to generalize and homogenize interaction between different software platforms. The initial development of the protocol was based on an analysis of the manner to describe interaction on different platforms as well as on several use cases. Detail of this analysis would be out of the scope of the paper; however, the results of this analysis suggested that interaction in a context of collaborative work may be divided into two categories: exclusive control - control of an object’s property by several users at the same time is not allowed and shared control - control of an object’s property by several users at the same time is allowed. In the two following paragraphs, each object is considered as a collection of properties.

- In case of exclusive control, when the user wants to access a property of an object, first s/he requests the
ownership of that object, then the property, and finally releases the object when still keeping control of the property. This mechanism allows the user to modify a property while other users can modify independently others properties so as to ensure both the safety of the property value and the access to other properties of the object.

- In case of shared control, a mechanism has been introduced to allow several users to modify the same property. The first user who gets the ownership of a property keeps the ownership of that property, but also integrates every contribution of the other users to it. When the owner of a shared property releases it, one of the other users takes instantaneously the ownership of the property.

Figure 3 describes the detailed structure of the CVNR. The CVNR is made up of two parts: core and tools. The module of user management also provides dynamic avatar creation. The core part composes of modules for session, user managements and dynamic avatar creation. As aforementioned, the client-server model was used in the platform. The MultiUser server was running on a separated PC. The core part takes care of the connection with the MultiUser Server. The tools part is a collection of modules used to interact with a 3D model. This part provides different metaphors for interaction in CVEs. Two tools are provided for the platform: the object selection tool and the object manipulation tool.

![Figure 3: Features of Collaborative VRNR](image)

We are going to detail the object manipulation tool which is the main part of the platform dealing with collaboration. The manipulation tool lets the user move object following the collaborative interaction protocol implemented as a state machine. This state machine (cf. Figure 4) has seven different states and its proper context for a selected object.

The initial state is the “IDLE” state. In this state, the manipulation tool is waiting for an object to be selected.

In the “IDLE” state, when an object is selected (cf. link (a) in Figure 4), the state machine enters the “Object Control Request” state in which the manipulation tool requests control of a parameter of the object. In case the control request is accepted (cf. link (b), Figure 4), the state machine enters the “Object Modification in Exclusive Mode” state in which the manipulation tool locks this parameter to prevent the access from other users. The modification is thus reserved to the owner of that parameter. Otherwise, if the requested parameter has already been controlled (cf. link (c), Figure 4), the state machine enters the “Access-Request Sending” state in which an access request (either for shared control or for exclusive control) is sent to the owner of the requested parameter.

In the “Object Modification in Exclusive Mode” state, if there is any access request (either for shared control or for exclusive control) from distant users (cf. link (d), Figure 4), then the state machine enters the “Access-Request Solving” state in which the manipulation tool waits for the answer from the parameter’s owner.

In the “Access-Request Sending” state, in case the parameter’s owner accepts the shared control request (cf. link (e), Figure 4), the state machine enters the “Object Modification in Cooperative Mode [CLIENT]” state. “[CLIENT]” is to mention the fact that the requesting user plays the role of a client of the owner of the requested parameter in this case. In case the exclusive control request is accepted (cf. link (f), Figure 4), the ownership of this parameter is given to the requesting user and the state machine returns to the “Object Modification in Exclusive Mode” state. Otherwise, in case the access request is refused (cf. link (g), Figure 4), the state machine returns to the initial “IDLE” state, the requesting user will be in a waiting list to be able to control this parameter once it is released.

In the “Access-Request Solving” state, if the parameter’s owner accepts the shared control request (cf. link (h), Figure 4), then the state machine enters the “Object Modification in Cooperative Mode [HOST]” state. “[HOST]” is to mention the fact that the parameter’s owner still keeps the ownership of that parameter and integrates contribution of the requesting users. In case the exclusive control request is accepted (cf. link (i), Figure 4), the parameter is released and the state machine returns to the initial “IDLE” state.

In the “Object Modification in Cooperative Mode [HOST]” state, the owner of a parameter integrates contributions of the user(s) in the “Object Modification in Cooperative Mode [CLIENT]” state. In case the last client gives up the control of that parameter (cf. link (j), Figure 4), the state machine returns to the “Object Modification in Exclusive Mode” state since the parameter’s owner is the only user who has access to the parameter at that moment. In case there are access requests from others users (cf. link (k), Figure 4), then the state machine returns to the “Access-Request Solving” state to solve these requests.

In the “Object Modification in Cooperative Mode [CLIENT]” state, in case the parameter’s owner releases his parameter (cf. link (l), Figure 4), the state machine returns to the “Object Control Request” state in which the first users in the list of clients of that parameter ask for the ownership of that parameter. This mechanism of owner swapping is completely transparent for users.

### 3.3 Usability Inspection

A collaborative working situation normally consists of two main phases: one phase in which the user works individually her/his
A demonstrator was built from the platform so as to provide us with an environment for studying issues on design and evaluation of CVEs. On the technical side, the demonstrator served to study issues on system design and performance. On the human factors side, it served to investigate usability issues of collaborative work within VEs and to design collaborative interaction metaphors as well.

In the following sections, we detail both different features of the demonstrator and impacts of the usability inspection on the design of collaborative interaction metaphors. Among the five characteristics of collaborative work aforementioned, we mainly focused on three points: Awareness of others, Shared context, Negotiation and communication for the usability inspection.

4.1 Demonstrator

The demonstrator allowed the user to work with a scene containing 3D objects in an immersive virtual environment. The user was able to select and move objects in the scene by using a given pointing technique. The user could work collaboratively with another user in a remote immersive virtual environment. The remote collaborator was represented by an avatar. Besides the avatar, different visual aids and interaction metaphors were introduced in the demonstrator to support the collaborative work within VEs. Details of the pointing technique, the avatar design and other collaborative interaction metaphors are presented in following subsections.

4.1.1 Pointing technique

A pointing technique is used to help the user select, move and position objects in the scene. In the demonstrator, the ray-casting technique was used for pointing. The ray-casting technique (the term used in [2]) presents a pointer under the form of a light ray emanating from the user's hand. The light ray indicates the direction of pointing and serves as a pointer for users to pick up and manipulate objects in the scene. The user used a tracked input device to control the pointing direction. The choice of ray-casting as the pointing technique for the demonstrator was quite evident since existing pointing techniques are mainly virtual hand-based or ray-based pointing techniques; besides, many studies have shown that ray-casting outperforms virtual hand in object selection tasks [3][11]. Virtual hand refers to a pointing technique which presents a 3D cursor in the form of a graphical presentation of human hand whose the position and orientation are mapped onto those of the input device.

In order to improve the mutual awareness among all collaborators, each collaborator was able to see both her/his light ray and that of remote collaborators. The light ray in the context
of CVEs played the role of visual aids for collaborators so as to help them be aware of activities of other remote collaborators. Details on the visual presentation of the light ray in this case appeared to be more important than in the case of single-user VEs. Thanks to usability inspection which focused on “awareness of others”, some improvements on the visual presentation of the light ray were added to overcome revealed usability issues.

The light ray indicates the position of the corresponding remote collaborator. The two ends of the light-ray were presented differently to overcome an eventual position ambiguity of the user’s hand. This is not an issue in single user VEs since the user is always aware of the start and the end points of the light-ray (the start of the light ray links to the user’s hand). Therefore, for the design of a box was added to the start of the light ray to indicate the position of the user’s hand.

Another issue is the colour code for the light ray of different collaborators. The colour code light ray in this case is among the visual aids to help identify one collaborator among others. For the design of the demonstrator which involves two collaborators, we used two different colours to distinguish the light ray of the local user from that of the remote collaborator.

4.1.2 Avatar

An avatar of the remote collaborator was introduced in the demonstrator. This was to improve the awareness of the local user about his/her remote collaborator. The avatar was a graphical presentation of the user in the VEs. As aforementioned, the light ray also contributes to indicate the position of the remote collaborator. However, the perception of the collaborator though a light ray moving in the air was quite disturbing. In case of collaborative work within VEs, an avatar presenting the remote collaborator might potentially improve the sense of co-presence in CVEs. In the demonstrator, the avatar was composed of three different parts: a head, an arm and a torso presenting the corresponding parts of the user.

4.1.3 Menu

A floating menu which is popped up on demand was introduced in the demonstrator. The floating menu serves as either a normal menu to provide the user with different options for object manipulation or a dialog box to display information and/or demands from the remote collaborator. The use of floating menu as dialog box provided an extra option for negotiation between collaborators. Even though verbal communication seems to be the most evident option to negotiate in this context, the problem of ambiguity between communication and commands, however, required another extra option to validate the command. The dialog box is useful in this case, particularly in case of many collaborators when the ambiguity could potentially become much more important.

4.1.4 Icon and Colour Code

Besides, the activities of the remote collaborator were made coherent by a colour metaphor applied to an icon attached to the selected object (cf. Figure 5). The icon turned red in the exclusive mode, i.e. no support for collaborative work and turned green in the collaborative mode, i.e. authorize the collaborative work. The icon in this case played the role of a shared artefact, visible to the remote collaborator to inform about the object state.

4.1.5 Communication

The platform had not supported the vocal communication yet. Therefore, the verbal communication was enabled to the users thanks to VoIP communication based software such as Skype™ and MSN™.

4.2 Observational Evaluations

An informal evaluation of the platform was organized at the Laval Virtual 2008 exhibition. A demonstration of a collaborative working scenario was presented in two CAVE®-like platforms: the first one is set of 4 projection walls: three sided-walls with 3m large for 4m height and the floor with 3x3 m) in Marseille and the second is a set of 4 projection walls, each with 3x3m in Laval, France. The purpose of the demonstration was to collect participants’ feedback on the usability of the collaborative platform connecting to two geographically distant sites in real-time condition. The impact of temporal lags in real-time condition and interaction metaphors for collaborative work were among the main focus of the evaluation.
4.3 Results and Discussion

There were about 30 participants (10 minutes per participant) who had tried the demonstration during Laval Virtual exhibition. The feedback was collected from the exchange between the expert user in Marseille and the participants in Laval, from the observation onsite in the CAVE-like environment in Laval and from informal discussion with participants after trials. Despite the low bandwidth network condition during the demonstration, all the participants had finished the collaborative task with success and with ease.

Many of the participants appreciated the use of 3D menu. The direct mapping of 3D menu compared to 2D menu could explain for that, supposing that all participants were familiar with the 2D user interface on PC. The negotiation using a 3D menu as dialog box seems more time-consuming compared to verbal negotiation as in real conditions. However, in case of collaborative work through VEs, the verbal negotiation can cause problems. During a verbal negotiation, vocal communication supports exchanging opinions to reach agreement/disagreement between two collaborators. Whereas both collaborators can understand the verbal exchanges, the VEs cannot distinguish the difference between communication and agreement/disagreement reached by both collaborators. An extra vocal communication to VEs should be used to communicate the agreement/disagreement to VEs. In this case, there are potentially ambiguities between collaborator-collaborator communication and collaborator-VEs vocal commands. The menu is still an evident option to validate a command. In the case of more than two users (in which the voice communication could be more ambiguous) and/or in case of many options to choose, the menu could be even more useful. The use of 3D menu for negotiation is thus promising.

The graphical presentation of the avatar was rather simple (only three parts of the user (the head, the arms and the torso) were presented) but still satisfied the participants. The avatar played well the role of indicating the position and intention of the remote collaborator. However, the effect of low bandwidth network was very noticeable for the avatar. The avatar moved itself sometimes without linking to the movement of the remote user. This caused the incoherence in the perception of the position and intention of the remote collaborator. The transmission instability and packet loss are among the main reasons explaining this problem. In a real condition, even with a high bandwidth network condition, the jitter and/or the interrupt of network communication is unavoidable; a solution to compensate to this problem should be envisaged. An avatar, even very simply presented, should be built from different sources of data received from different sensors attached to the user so that in case of lost of some data sources, the position and movements of the avatar can still be derived from other data sources.

Regarding the icon and colour code, the colour indicating the state of the object is very relevant. This is somehow a mapping to the colour code of “traffic light” which is intuitive and easy to understand. The colour in this case is a shared artefact which informs what is currently being done by collaborators in the context of the task goals.

5 Conclusion and Future Work

The paper has presented a software platform for CVEs in a multidisciplinary setting for studying issues of design and evaluation of CVEs. The platform was built on Virtools software suite which is novel compared to existing CVEs platforms such as DIVE and MASSIVE. The focus of the paper was not on specific issues of CVEs such as avatar design, pointing technique or negotiation and communication which are among the main features of the demonstrator. Instead, the effectiveness of such a
framework built on a Virtools-based platform for studying issues of CVEs was our main focus. From that point of view, some lessons can be drawn from the process used in design and evaluation of a demonstrator of a collaborative work based on the proposed Virtools-based platform.

First, Virtools software suite, Virtools MultiUser Pack and VR pack in particular, has proven to be feasible for building CVEs applications on different platforms. The platform presenting the present paper, which takes advantages of Virtools in supporting rapid prototyping, would be useful for investigating different issues in CVEs which normally required customized setting for each issue (e.g. avatar presentation, visual aids, etc.). Second, the iterative process of design and usability inspection has proven effective in the preparation of the demonstrator. The set of used guidelines was rather simple and still at a general level but efficient for systematically analyzing and identifying usability problems of the demonstrator. The only factor on the collaborative working side is effective for revealing usability issues of collaborative work within VEs. However, the individual mode (user interaction in VEs) is also an important factor which has impact on the collaborative mode (collaborative work though VEs) in CVEs. Existing heuristics [16] and ergonomic criteria [1] for design and evaluation of the VEs interface would be useful for the design and evaluation of individual mode in CVEs.

Besides, some interesting issues which were revealed throughout the study such as the perception of avatar, the use of verbal communication for both negotiation and issuing commands to VEs require in depth studies to gain more insight into these issues. The scenario in the demonstrator involving two collaborators was just a simple case of collaborative work. Complex scenario with three or more collaborators will be taken into consideration in further investigation. The cognitive charge regarding the awareness of others’ activities are potentially higher in those scenarios and would reveal more issues of collaborative work within VEs. Efforts to combine the verbal information with the avatar (e.g. who is talking, who is doing what) so as to understand the activities and demands of others are potentially higher in those cases. In this perspective, future investigations will focus on both quantitative and qualitative studies to get insight of those issues. The platform will be used as a test-bed for experimental study of different issues regarding collaborative work within VEs.

ACKNOWLEDGEMENT

This work was carried out in the framework of the Project ANR² Part@ge N° 06 TLOG 031

REFERENCES


¹ http://www.agence-nationale-recherche.fr/
² http://www.rntl-partage.org/