



- D3.5, Usage Evaluation of the Online Applications

Part C: CVE Design & Inspection Method

Project Number:	ACTS Project N. AC040
Project Title:	COVEN - Collaborative virtual environments
Deliverable Type:	P*

CEC Deliverable Number:	A040-UCL-CS-DS-P-035c.b1
Contractual Date of Delivery to the CEC:	31 st October 1998
Actual Date of Delivery to the CEC:	6 th November 1998
Title of Deliverable:	Usage Evaluation of the Online Applications
Work package contributing to the Deliverable:	WP3
Nature of the Deliverable:	R**
Author(s):	Jolanda Tromp (Nottingham), Anne-Marie Sandos (Nottingham), Anthony Steed (UCL), and Stefan Thie (KPN).

Abstract: This document describes the COVEN Inspection method. This is a new method developed to address the unique design properties of CVEs. The method is based on the traditional HCI Inspection method, the method for Inspection of single user virtual environments by Sutcliffe and Kaur, the cycle of collaboration developed for COVEN, described in Del3.5A, and Del3.5B, and a user context analysis method from the RESPECT Project. The development of the COVEN Inspection method is further informed by interviews with CVE designers to establish their design practice and problems, and an assessment of the reception of the previous Inspection report by the COVEN designers.

Keyword list:

Collaborative Virtual Environments, Usability Inspection, Presence, Cognitive Walkthrough, Heuristic Evaluation

***Type: P-public, R-restricted, L-limited, I-internal**

****Nature: P-Prototype, R-Report, S-Specification, T-Tool, O-Other**

1	INTRODUCTION	3
1.1	ADAPTATION OF 2D INSPECTION METHOD NEEDED	3
1.2	ORGANISATION OF THIS DOCUMENT.....	3
1.3	ACKNOWLEDGEMENTS	3
2	CVE DESIGN & INSPECTION METHOD	4
2.1	INTERACTION CYCLES	4
2.2	SEVERITY RATINGS	9
2.3	THE STEP BY STEP PROCESS.....	9
3	EXAMPLE INSPECTION – USER CONTEXT ANALYSIS	10
3.1	USER’S INITIAL GOALS	10
3.2	ANTICIPATED USER GROUPS.....	10
3.3	USER DESCRIPTION	10
3.4	USER GROUP DESCRIPTION	10
3.5	USER TASK DESCRIPTION	10
4	EXAMPLE INSPECTION - THE TASK SCENARIOS.....	11
5	EXAMPLE INSPECTION - TASK ANALYSIS	17
5.1	INTRODUCTION.....	17
5.2	INTERNET FRONT-END AND AVATARS.....	17
5.3	COMMUNICATION.....	18
5.4	GROUP FUNCTIONALITY AND SUBJECTIVE VIEWS.....	20
5.5	CROWD CONTROL AND AUTONOMOUS AGENTS	22
5.6	EMBEDDED APPLICATION FUNCTIONALITY	23
6	EXAMPLE INSPECTION – INTERACTION CYCLE ANALYSIS.....	26
7	EXAMPLE INSPECTION – COGNITIVE WALKTHROUGH	29
8	EXAMPLE INSPECTION – REPORT	39
9	METHOD DEVELOPMENT.....	45
9.1	INFLUENCE OF PREVIOUS INSPECTION RESULTS	45
9.2	RESULTS OF INTERVIEWS WITH CVE DESIGNERS	45
9.3	FUTURE WORK ON THE DESIGN AND INSPECTION METHOD	49
10	FINAL CONCLUSIONS.....	52
11	REFERENCES.....	53

1 Introduction

The Inspection method described in this document is a new method developed by COVEN to address issues unique to usability testing of Collaborative Virtual Environments. This method is an adaptation of the traditional HCI-based Inspection (Nielsen & Mack, 1994). The method is also considerably extended based on an early version of the pioneering development of an Inspection method for single user VEs (Sutcliffe & Kaur, 1998). The method reported here, is a method to inspect multi-user CVEs. It is under development throughout the COVEN Project. It has now been applied to the second iteration of the usability studies of the COVEN Project (Act 3.4). This document describes the new Inspection method and the extensions, illustrated with the results from COVEN Inspection II.

With this inspection method two main goals can be accomplished

- Guide the design of a CVE at the level of objects, interactions and task-flow.
- Test the CVE at the level of representative user needs and struggles.

The representation of the environments has a strong bearing on how user tasks will be carried out. By going through the actions which a user will be expected to perform in the CVE, and noting the things that are not obvious or difficult one is inspecting the design.

1.1 Adaptation of 2D Inspection Method Needed

Virtual Environments (VEs) are a novel application area of computing technology, demanding an understanding of human-computer interaction and computer mediated human interaction in virtual spaces. Standard usability engineering and HCI evaluation techniques do not directly address the usability problems introduced by these new applications. Examples of such issues are 3D object interaction, freedom of navigation in a pseudo 3D space, and human-human collaboration mediated by the VE.

There are neither Cognitive Walkthroughs (CWs), nor Heuristic Evaluations (HEs) specifically for the Inspection of Collaborative Virtual Environment (CVE) interfaces. CWs and HEs have been developed to inspect single-user, 2D interfaces, and although many of the individual CW and HE items could also apply to the interfaces of CVEs, none of the items address interface issues which occur only in CVEs.

1.2 Organisation of this Document

The remainder of this document is organised as follows: Section 2 gives an overview of the method and outlines the forms used. Sections 3 to 8 give an example of the method through application to the COVEN citizen application. Section 9 discusses the utility of the method as described, a review of designer practice and the reception of the previous Inspection report. Finally, the document concludes with a discussion of future directions of work required to improve the method.

1.3 Acknowledgements

Large parts of this document are based on other people's work. To be precise:

The COVEN Inspection method has been built on top of the *Inspection method for single-user VEs* by Prof. Alistair Sutcliffe and Kulwinder Kaur.

The Scenario Description is based on the text of Del. 2.7 "*Specification of the extensions to the COVEN Demonstrator*".

The User Context Analysis is based on parts of the *User Requirements Framework Handbook*, Deliverable 5.1 of the RESPECT Project.

The COVEN Inspection method has been developed by Jolanda Tromp (Nottingham), with help of Anne-Marie Sandos (Nottingham) for the interviews with the CVE designers, Stefan Thie (KPN) for the Interaction flow diagrams and the cycle-task allocations, Anthony Steed (UCL) for applying the method to the COVEN application, and Kulwinder Kaur and Alistair Sutcliffe for advice on developing the method.

2 CVE Design & Inspection Method

The CVE Design and Inspection method described here is a step by step guide to perform a Cognitive Walkthrough (CW) of the CVE interface. In order to perform an inspection of a CVE design, a short CVE user context analysis needs to be created first. This means identifying the main users, the situation of use and the CVE system functions and objects. Next, representative task scenarios are developed which one needs to create a task-tree, which is subsequently used to create steps for the CW.

Performing a CW on tasks in a CVE is not as straightforward as performing a CW on tasks in a 2D desktop application. Typical CVE tasks may be rather unstructured, and therefore difficult to represent fully in the task-tree, and consequently difficult to inspect fully. However, it is possible to specify general CVE tasks, unique to VEs and CVEs, and use these tasks as action cycles for the inspection. E.g. interaction with three dimensional virtual objects in a three dimensional virtual space, navigation in a three dimensional virtual space, and human-human collaboration in multi-user three dimensional virtual space which may be comprised of multiple rooms or worlds.

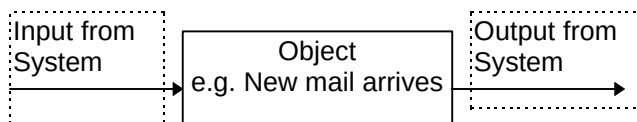
The COVEN Inspection method has been built on top of the Inspection method for single-user VEs developed by Prof. Alistair Sutcliffe and Kulwinder Kaur (Sutcliffe and Kaur, 1998).

2.1 Interaction Cycles

In order to address these tasks, six task interaction cycles have been identified (Kaur et. al., 1997; Tromp, 1998). These action cycles are used to identify user needs, task scenarios and cognitive walkthrough items for the Inspection. The cycles appear throughout the description of the Inspection method and help guide the usability engineer through the Inspection task. It is very well possible that for one task several task action cycles are possible. This is not disturbing but very helpful in clarifying what can be done with a CVE desktop interface and how.

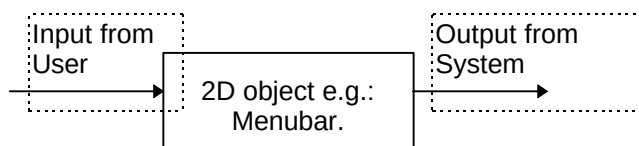
- System Initiative Cycle, where the user has to deal with the system temporarily taking control over the cause of events in the CVE, either because the user has caused this to happen or because the system has instructions to do so.
- Normal task action Cycle 2D, where a user is interacting with 2D information in the environment in order to achieve a certain goal, such as text menu's or 2D pop-up displays.
- Normal task action Cycle 3D, where a user is interacting with a 3D object in the environment in order to achieve a certain goal.
- Goal Directed Exploration Cycle, where a user is searching for something known to be in the environment in order to achieve a certain goal.
- Exploratory Browsing Cycle, where a user is navigating through the CVE in order to achieve an understanding of the world layout, or world order.
- Collaboration Cycle, where the user is interacting with other users in the CVE, either to collaborate on a certain task, or to socialize.

2.1.1 System Initiative Cycle



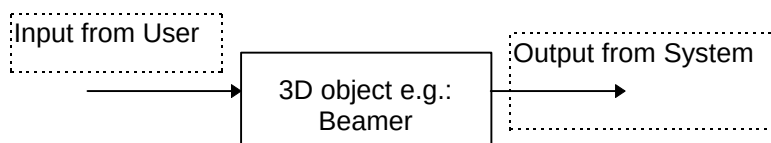
Function	Generic Task Scenario for System Initiative	Potential Problem and Design Solution	Severity Rating	Ref. Numb
	i: Is it clear to the user that the system has taken control?			
	ii: Can the user resume control at any point and is the appropriate action clear?			
	iii: Are the effects of system actions visible and recognizable?			
	iv: Are the system actions interpretable?			
	v: Is the end of the system action clear?			

2.1.2 Normal task action Cycle 2D



Function	Generic Task Scenario for a Normal Task Action 2D	Potential Problems and Design Solutions	Severity Rating	Ref. Numb
	i: Will the users be trying to produce whatever effect the action has?			
	ii: Will users be able to notice that the correct action is available?			
	iii: 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?			
	iv: After the action is taken, will users understand the feedback they get?			

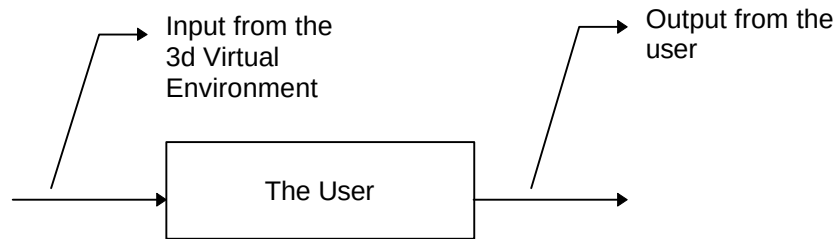
2.1.3 Normal task action Cycle 3D



Function	Generic Task Scenario for a Normal Task Action	Potential Problems and Design Solutions	Severity Rating	Ref. Numb

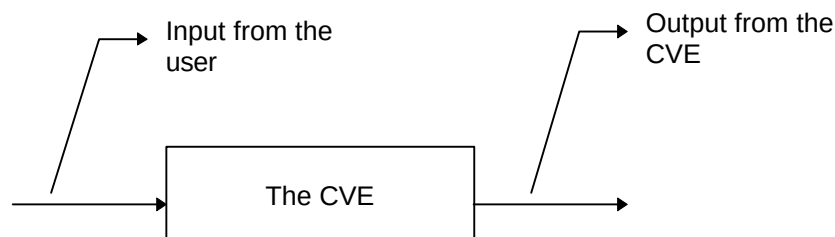
	i: Can the user form or remember the task goal?			
	ii: Can the user specify an intention of what to do?			
	iii: Are the objects or part of the environment necessary to carry out the task-action (users new intentions) visible?			
	iv: Can the objects necessary for the task action be located?			
	v: Can the users approach and orient themselves to the objects so the necessary action can be carried out?			
	vi: Can the user decide what action to take and how?			
	vii: Can the user carry out the manipulation or action easily?			
	viii: Is the consequence of the users action visible?			
	ix: Can the user interpret the change?			
	x: Is it made clear to the user what the next correct/needed action could be?			

2.1.4 Goal Directed Exploration Cycle



Function	Generic Task Scenario for Goal Directed Exploration	Potential Problems and Design Solutions	Severity Rating	Ref. Numb
	i: Does the user know where to start looking?			
	ii: Can the user determine a pathway towards the search target?			
	iii: Can the user execute movement and navigation actions?			
	iv: Can the user recognize the search target?			
	v: Can the user approach and orient themselves to the objects so the necessary action can be carried out?			
	vi: Can the user decide what action to take and how?			
	vii: Can the user carry out the manipulation or action easily?			
	viii: Is the consequence of the users action visible?			
	ix: Can the user interpret the change?			
	x: Is it made clear to the user what the next correct/needed action could be?			

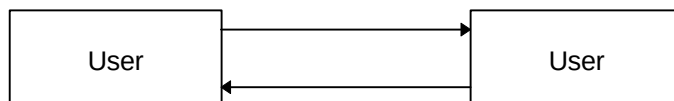
2.1.5 Exploratory Browsing Cycle



Function	Generic Task Scenario for Exploratory Browsing	Potential Problems and Design Solutions	Severity Rating	Ref. Numb

	i: The user determines a pathway for movement.			
	ii: The user executes movement and navigation actions.			
	iii: The user recognizes objects in the environment.			
	iv: The user interprets identity, role and behaviors of objects.			
	v: The user remembers important objects or locations.			
	vi: The user forms a mental map of the explored environment			

2.1.6 Collaboration Cycle



	Generic Task Scenario for Collaboration	Potential Problems and Design Solutions	Severity Rating	Ref. Numb
	i: Can the user locate the other user(s)?			
	ii: Can the user recognize the identity of the other user(s), tell the other users apart?			
	iii: Are the communication channels between the users effective?			
	iv: Are the actions of the other user(s) visible and recognizable?			
	v: Can the user act on a shared object while keeping the other user(s) in view?			
	vi: Can the user easily switch views between the shared object, other locations/object of interest and the other user(s) (sweep from one to the other)?			
	vii: Can the user get an overview of the total shared space and all other users in it?			
	viii: Can the user tell when there are interruptions in the attention of the other user(s) to the CVE?			

2.2 Severity Ratings

During the Inspection severity ratings between 0 and 4 are given for each problem found. These ratings can be regarded as guidance for the urgency of redesign and development, and they can be used as recommendations for a plan of action (see Section 8). The severity of a usability problem is a combination of five factors:

- The frequency with which the problem occurs: Is it common or rare?
- The impact of the problem if it occurs: Will it be easy or difficult for the users to overcome?
- The persistence of the problem: Is it a one-time problem that users can overcome once they know about it or will users repeatedly be bothered by the problem?
- The design concept: Does this design concept radically decide and/or exclude user options?
- If a metaphor is used: is the metaphor appropriate and consistent?

2.3 The Step by Step Process

An analysis of the task domain is required in order to get a good understanding of the precise user needs. Therefore a short description of the expected users, and the system requirements is created, using forms to help guide the usability inspector. Based on this user context analysis, task scenarios are created, which are used in the Cognitive Walkthrough Inspection.

- **The User Context Analysis**

An analysis of the user context is required in order to get a good understanding of the specific user needs, which one needs to keep in mind whilst doing the inspection.

- **The Task Scenarios**

The major tasks in the application should be described as they are expected to occur during a typical task performed by representative users, resulting in a scenario description.

- **The Task Analysis**

Tasks, system functions, and 3D objects should be identified from the scenario description, they may be listed as general activities, and more specific activities subsumed under general ones.

- **The Interaction Cycle Analysis**

The tasks, system functions and 3D objects should be linked to the one or more of the 6 interaction cycles.

- **The Cognitive Walkthrough**

For each task, system function and 3D object a number of Inspection questions exists, which are used in the CW.

- **The Report**

For each problem found with a task, system function or 3D object, a note is made and a reference number is assigned.

3 Example Inspection – User Context Analysis

The User Context Analysis is a necessary element of the Inspection, because it helps to clarify the specific user needs and requirements which the Inspector has to keep in mind while inspection the application.

User's initial goals, anticipated user groups, user description, user group description, and a short user task description should be created in order to clarify and possibly split up the different user needs and requirements.

This user Context Analysis is created using the User Requirements Framework Handbook, Deliverable 5.1 from the RESPECT Project (RESPECT, 1997).

3.1 User's initial Goals

Individual virtual travelers can meet other virtual travelers. The traveler has the possibility to collaborate and communicate through audio and text chat with other travelers, making new friends, meeting travel agents, hotel operators, other tourist service providers (souvenir vendors, etc.), archaeologists, local experts, etc.

3.2 Anticipated User Groups

Travelers, tour guides, and travel operators. At this stage of development of the applications only travelers are supported.

3.3 User Description

A traveler is expected to be a citizen, wanting to gather information about possible holiday locations, making a virtual trip to a travel agent office.

3.4 User Group Description

There are three basic roles that we distinguish in the citizen application:

- Traveler
- Tour guide
- Travel operator

Since the group aspect and collaborative aspect of the citizen application is important, we state that each of these three basic roles may be extended with one of the following features:

- Group member
- Group leader

3.5 User Task Description

Traveler activities can be distinguished as three separate options:

- the user will travel as an individual,
- the user is one of a small group of relatives or close friends, or
- total strangers are preparing their group travel together.

In the latter two cases the group will choose a leader in a natural and implicit way. During the visit in the Virtual Travel Agency the role of the leader can be altered between the participants.

4 Example Inspection - The Task Scenarios

The major tasks in the application should be described as they are expected to occur during a typical task performed by representative users, resulting in a scenario description. In order to create a scenario description the following questions may be used:

- I. Where might the user need to explore?
- II. What information or object might the user need to find?
- III. What operations may the user need to try out?
- IV. What is the next logical action from the users' point of view?

The answers will give indications about how much of the virtual environment the user may need to explore and areas, objects and activities of particular interest therein.

An exploration path through the CVE can be specified which represents a typical pattern of exploration implied by the users information needs. An overall floor plan of the virtual world can be used and the exploration plan traced in it to assess how much is it (a) necessary to explore and (b) likely to be explored in view of the cues given in the environment. In applications where exploration is important, exploration areas of interest, and paths may be specified. These paths can be used to create exploration scenarios.

4.1.1 Example map from COVEN Platform.

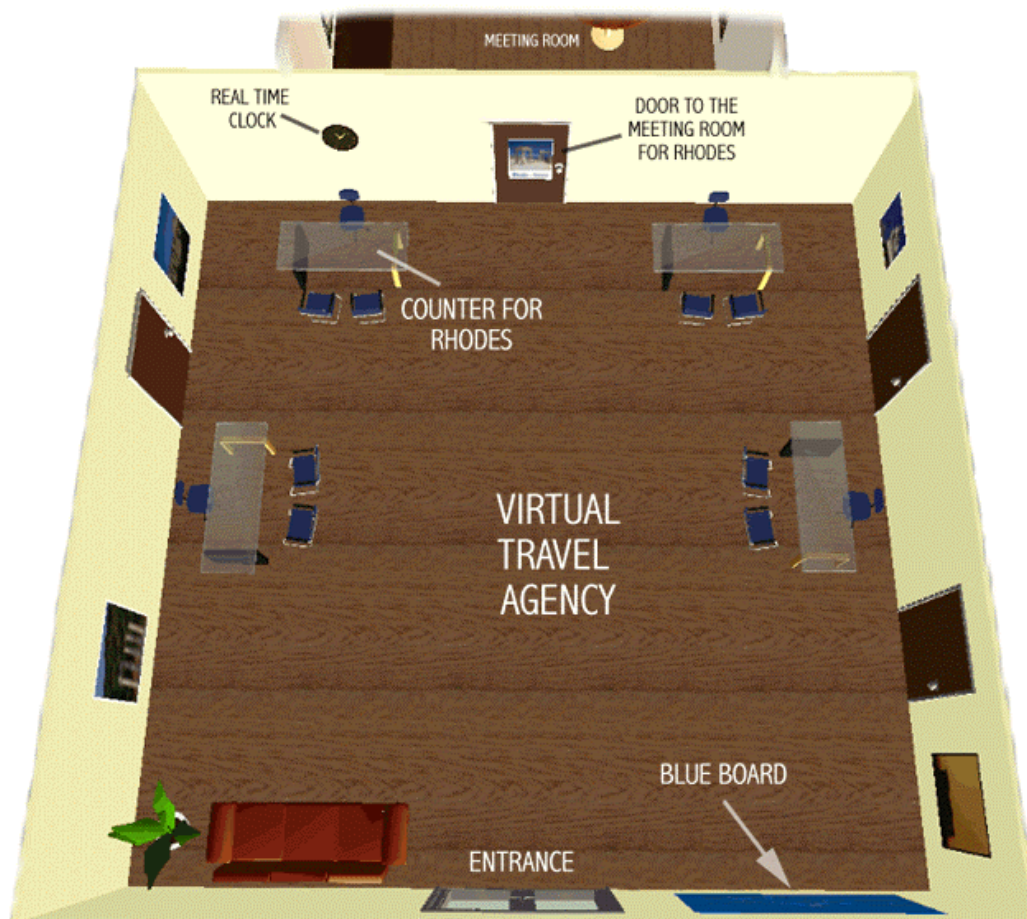


Figure 1: Virtual Travel Agency

Figure 1 shows the Virtual Travel Agency (VTA zone). One door (with a picture of Rhodes on it) leads to the meeting room for Rhodes. The other doors are reserved for other possible holiday destinations.

On the walls there are posters of holiday destinations. Next to the entrance of the VTA is a so-called *blueboard*. Blueboards are used for text based communication in the application. Next to the door leading to the Rhodes meeting room there is a clock showing the actual real-world time.

The Scenario Description is based on the text of Del. 2.7 “*Specification of the extensions to the COVEN Demonstrator*”.

4.1.2 The Virtual Travel Agency

The scenario describes the viewpoint of an individual traveler, Max, interested in meeting other travelers, and exploring the zones of the application, namely: the Virtual Travel Agency (VTA), The Rhodes Meeting Room (MR), the Rhodes Virtual Flight Zone (VFZ) and the Acropolis of Lindos Zone (LZ). The scenario exercises functionality that exists in the online application (see Deliverable 2.4) and which is proposed for the final version (see Deliverable 2.7a).

Note that the overview of functionality and the task analysis is based on both the functionality as reported in Deliverable 2.4 (On-line Citizen Service) and the design document Deliverable 2.7a (Specification to the Extensions of the COVEN Citizen Demonstrator). This is to facilitate a comparison between the existing implementation and the final implementation of the application. We have found this a useful tool when assessing the impact of design recommendations between application development cycles. However it does mean that many tasks identified are not subject to inspection in this document.

Let's meet Max. Max wants to go on vacation but he has not made up his mind where to go. He decides to login on his computer at home that is connected to the Internet. Max searches the World Wide Web for the Travel Operator and takes a look at their site. He enters a Virtual Travel Agency (VTA) (see Figure 1) which is a three-dimensional computer generated virtual world. Max is represented by his favorite avatar. A lot of people are present in this VTA, but Max does not see them all, neither can he interact with them individually. The VTA is the place where everybody in the world starts when planning a holiday and this is also the place where people actually book a trip and pay for it.

In the VTA travel operators are situated behind counters. Max walks around and speaks to one of these travel operators to obtain information. The travel operators are real human beings represented by an avatar, just like Max, who earn their living by providing on-line customer travel service. They can help Max with everything he wants with respect to planning his holiday. If Max wants to communicate with people who have been to a certain place, the travel operators can direct him to these people. If he wants to obtain information about various holiday packages or the costs of these packages the travel operators can help him. Every source of travel information that Max is being provided with, will automatically be stored in his Personal Data Container. The flyers provided by the operator provide are free of charges, but the detailed map with walking trails costs 3 ECU.

Max takes a look at the map and decides to buy it. He takes the map, turns to the operator and says he would like to buy the map. The operator automatically asks “how would you like to pay?” Max has three choices: with a credit card, with a smart card or by pre-printed giro credit slip that will be sent to his home address. Max decides to pay with a smart card. With the card numbers available in his computer system he gives permission to pay. The data will be transferred to his Personal Data Container.

The Travel Operator sells vacations to travelers. To attract people the operator offers several services. For instance, travelers can book a trip, get information and get brochures. In the Citizen Application the travel operator is located in the VTA. The travel operator needs information from the travelers themselves so he can organize direct marketing actions and arrange trips that fit better to the customer's wishes. Using the PDC, the Data Containers of the traveler and the operator they can easily exchange information. The data may contain text, images, audio and video.

4.1.3 The Meeting Room

After information gathering Max decides that he wants to tune in on one particular location, the Greek Island of Rhodes. The travel operator Max was talking to, directs him to the Rhodes Meeting Room. In the Meeting Room Max can access large quantities of data in the form of text, images, audio, and video. Other travelers have stored the photos they have taken during their holidays. Max can contact these people by sending them a message or, if they are present at the time he is on-line, Max can have a chat with them.

Max chooses a presentation and a photo-slide-show starts showing pictures of the various attractions of Rhodes (see Figure 2). The slide-show is shown on a large screen on the wall and is supported by audio explaining the slides. When no slide show is running the projection screen shows a map of Rhodes. Max admires some posters on the walls picturing the beauty of the island. He thinks these posters are very interesting, so he sends a message to the person who took the pictures. Max would like to know when these pictures were taken, and in what hotel this person stayed. Max checks the schedule of this person. It appears that this person is on-line at the moment and Max finds out the name of the hotel during a private chat.



Figure 2: Rhodes Presentation Controls and Display

While Max is chatting, two other travelers are talking to each other, several yards away from Max. They are just far enough so that Max is unable to hear or read what they say. Max knows that they are part of some sort of a group because they are standing in 'speaking' distance, close to each other. Max moves a bit closer to this group because he is interested in meeting other people, either to hear about their experiences and plans, or to make new friends who he can meet during his holidays. Max moves into close proximity of the group, and greets them. The group members enlarge their circle to allow Max to join them. The group members greet Max, and start a conversation with him. One of them has a good quality multi-purpose male-like avatar and is called John. John is logged in from Stockholm. The other avatar looks like it has been picked out of the avatar showcase of the travel agency. She is named Hillary and is from Brussels.

It appears that John and Hillary also have never met before but they can talk about all sorts of things, because they exchanged referential cards holding personal information, which will automatically be stored in their respective Personal Data Containers. Besides, they are standing in a meeting room dedicated to one particular holiday island. Max is invited to join the group. He thinks he is making a good chance of having found friends to meet during his holidays.

4.1.4 The Virtual Flight Zone

Max has seen enough of the regular information media, he wants to have an immersive experience on the actual location. John suggests traveling around being invisible for others but still see, hear and interact with everything, like a normal trip. Such trips are also called Ghost Tours. Users can choose if they do not want to be bothered by others. Max decides to join the group of John and Hillary.

Four options of making a Ghost Tour are available:

1. A group is visible for other groups and sees everybody who is visible, but cannot interact with other groups.
2. A group cannot see any other group and is not visible for other groups either.
3. A group is visible for other groups but cannot see other groups.
4. A group is not visible for other groups, but can see everybody who is visible.

John suggests the last option. He also says that he knows his virtual way in Lindos, so he offers to be their tour guide. Using a group decision support tool Max and Hillary choose to be visible to others, but they do agree John showing them around. With help of the decision support tool, option 1 is chosen. John will be the group leader, which means whatever movement he makes, Max and Hillary will stay within the group's speaking and hearing distance.

John starts to teleport himself to Rhodes (see Figure 3), resulting in the fact that Hillary and Max teleport with him and 'fly' beside him above the Island. When John activates his personal teleporting functionality he soon finds himself flying towards the island of Rhodes. He has a control panel in front of him to navigate over the island of Rhodes. Max and Hillary automatically stay close to John. However, if one of them at a certain moment prefer to go his/her own way, they break the group band and leave each other.

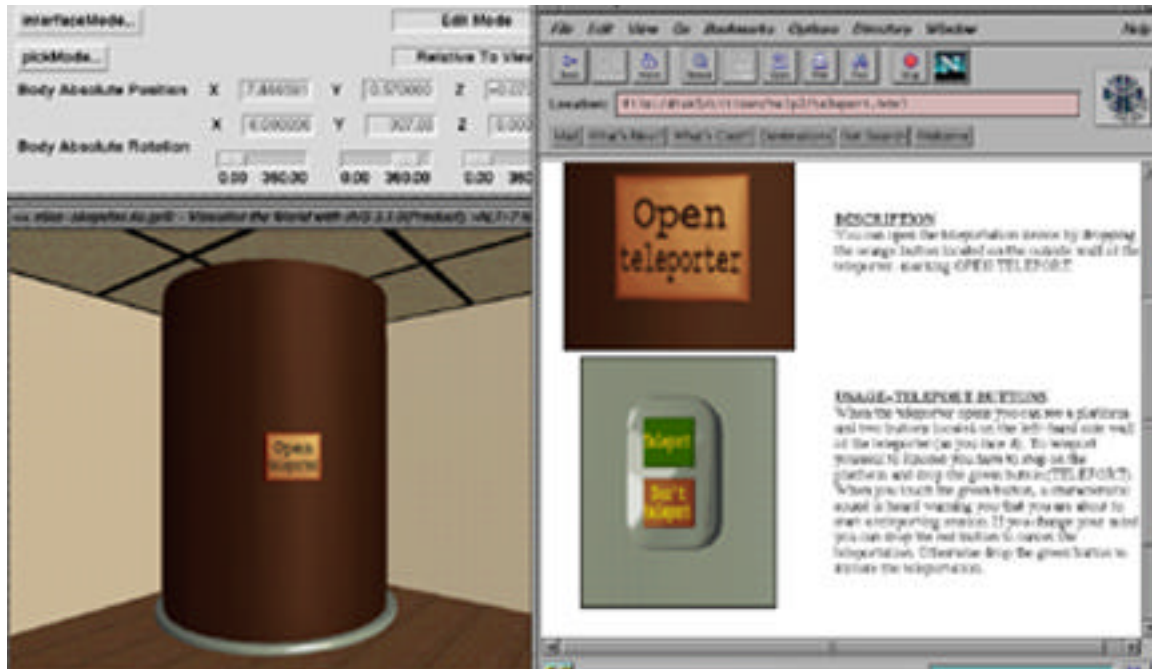


Figure 3: Teleporter and associated help screen

Rhodes is represented by a three-dimensional, thematic map that was created using a satellite height image (see Figure 4). The map is overlaid by another satellite image with a resolution of 30x30 meters showing where the woods are and the rivers run. The map provides a realistic picture of the land-cover and land-use of the destination.

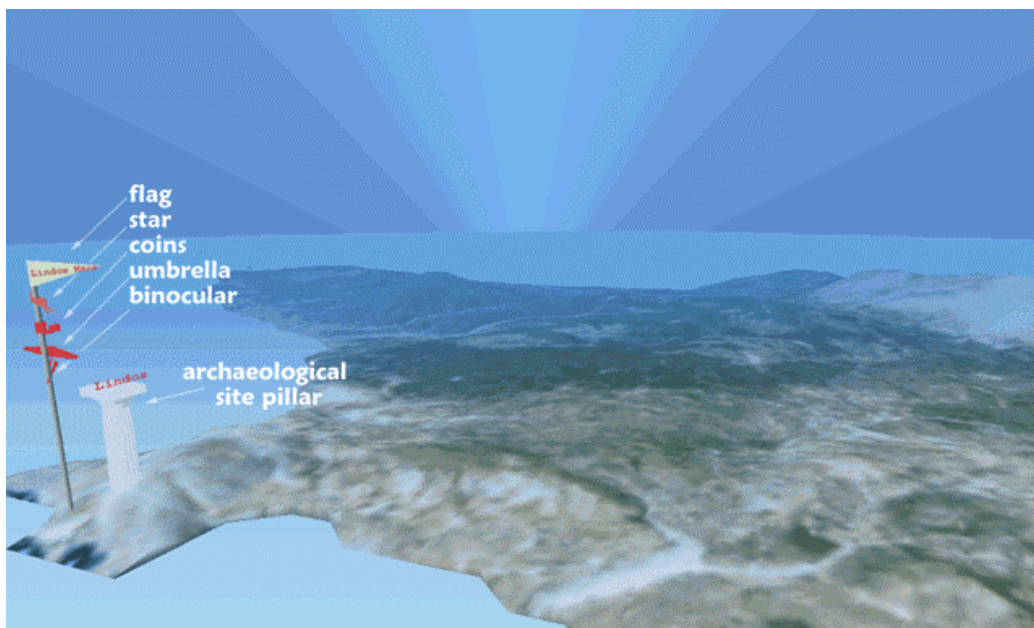


Figure 4: The Virtual Flight Zone

Among other things, Max looks for the hotel he just found out about. Important locations such as hotels or interesting archaeological sites are represented on the 3D map. A search for the name of the hotel activates a blinking red light somewhere on the horizon. Max suggests John to fly to that place. When they get there they find out that they can virtually visit the hotel the way it is built in the real world. They enter the hotel, walk through the corridor and enter a hotel room. It is a nice room with a marvelous view on the beach. Back in the lobby they find information about the hotel room such as the price, the distance to the beach and swimming pool and when it is vacant. This information is available both a visual and a textual form. They talk with the autonomous Hotel Operator who is standing behind the counter in the lobby.

Since the three enjoy their companionship, they decide to stay in the same hotel. Max likes this hotel, but is interested in entertainment in the near surroundings. Hillary likes any hotel with a swimming pool and John does not mind as long as the hotel is not too expensive. They ask a Tour Guide to give advice. Together with the Tour Guide they check out seven hotels in the Virtual Flight Zone.

The Tour Guide knows interesting places and can give advice 'on the fly'. In the Citizen Application the Tour Guide is located in the Virtual Flight Zone or the Lindos Zone. Using the PDC, the Data Containers of the traveler and the guide they can easily exchange information. The data may contain text, images, audio and video. In case the Tour Guide is busy guiding another group, the group may decide to wait till he is available, or to choose for the Automatic Tour Guide. The group leader can 'reserve' the Tour Guide by adding their names and request to the PDC of the Tour Guide. The Tour Guide will find the group as soon as he is finished guiding the other group.

Max, John and Hillary are interested in the location of the bars. Using the site/search functionality all the bars in the vicinity are visualised. Although the view on the island changes for Max, John and Hillary, this does not influence the view for any other group present in the Virtual Flight Zone. This artifact is referred to as 'subjective views'.

Two hotels seem OK. Max and Hillary would like to stay at the first hotel called 'Hotel Mare', while John prefers Hotel 'Cheapie'. Their main issue is money. Hillary argues that drinking beer is more expensive than staying at a more expensive hotel. Max interferes and makes a list of the costs. The group decision support tool presents an overview of the financial situation. Using the tool they decide to go to stay at 'Hotel Mare'. Meanwhile the Tour Guide has left the group to guide another group that was waiting for him.

4.1.5 The Lindos Zone

Max would also like to visit the most interesting and important archaeological site. By putting another query they are advised to visit the Acropolis of Lindos. A blinking light appears and some travel information is being displayed: Lindos is located 55 km south of the city of Rhodes. The Acropolis is located on a rocky hill above the village of Lindos. It has a triangular shape covering an area of 8400 m². The various buildings at the top of the hill belong to various historical periods from the 6th century B.C to the 14th century AD

Still in the Virtual Flight Zone they fly to the archaeological site pillar which represents the Acropolis of Lindos. They decide to virtually visit Ancient Lindos. The Acropolis of Lindos is modeled as it was in ancient times see Figure 5). Although nowadays there are only a few ruins remaining, the virtual world gives a good impression of how life must have been in Lindos during the 6th century BC While exploring the Acropolis, Max observes the souvenir shop and decides to leave the group using his own navigation controls. However, he stays a member of the group which means that he can still have a direct chat and audio conversations with John and Hillary.

Max enters the souvenir shop and communicates with the Souvenir Vendor which is another autonomous agent. He picks up a vase, turns it around and sees a 3D animation of what the vase was used for. Max decides to buy this souvenir and pays for it using his chipcard. The souvenir will be shipped to his home address by express mail.



Figure 5: The Lindos Zone

4.1.6 Booking the actual trip

Max, Hillary and John decide to book a trip to Rhodes together. They teleport back to the VTA and make reservations in the office. Using their PDC all necessary information can be given to the Travel Operator. They say goodbye to each other and leave the application.

During the weeks they spend on Rhodes they use a portable device which is connected to the application on the Internet. While walking on the Acropolis in real live, they can counsel the 3D image of the site, ask information of the online Tour Guides or answer questions of other travelers.

Two months after their holidays Max goes back to the VTA. During his vacation he has met John and Hillary in real life. In Greece they made an appointment to meet in the VTA again. Max meets other travelers and together they chat about their vacation. Max has taken pictures he made during his vacation with him. He is able show his pictures to others using his Personal Data Container. John and Hillary can get copies of his pictures by just selecting the picture. The pictures will be saved automatically in their PDC. Max and Hillary make another appointment.

Hillary has promised to meet him on their favorite spot on Rhodes tonight, but she has left him a message which he found as soon as he logged in, saying she would be late....

5 Example Inspection - Task Analysis

Tasks, system functions, and 3D objects should be identified from the scenario description, they may be listed as general activities, and more specific activities subsumed under general ones.

5.1 Introduction

Two types of CVE users are distinguished:

1. Users immersed in the CVE with a Head-Mounted Display and using a 3D mouse, but no keyboard;
2. Users at a desktop computer, using a regular 2D mouse and keyboard.

For the virtual travel application, we assume a desktop user and an Internet front-end. The main differences in the user interface for the immersed user and the desktop user can be described as follows:

- In an immersive virtual environment all interaction between the user and the system must be done through the manipulation of 3D objects (including buttons). The keyboard can not be used.
- In a desktop environment part of the screen may be reserved for additional 2D windows. The advantage is that 2D windows can be easily manipulated by a desktop user and do not need to clutter the view on the environment.

The functionalities that will be supported by the Virtual Travel Agency application are divided in five groups:

1. Internet front-end and avatars, for personalisation of the application.
2. Communication, to facilitate interaction with other users.
3. Group functionality and subjective views, to support group decision making.
4. Crowd control and autonomous agents, for autonomous behavior of the application.
5. Embedded application functionality.

Each of the groups of functionalities is discussed below.

5.2 Internet front-end and Avatars

5.2.1 Internet front-end

Collaborative Virtual Environment applications require some sort of front-end before the user is allowed into the actual application. The main function of such a front-end could be described as "to enable easy access to the shared application". We identify three functions for the front-end:

1. Access to the application
2. Personalisation
3. Persistence

Personalisation of Internet applications is currently a hot topic. The aim of personalisation is to provide users with an application tailored to their individual needs and which protects their privacy. In a shared virtual environment it is desirable that users could choose their own avatar. Furthermore, through personalisation a company or information provider can establish a closer relationship with a customer. It would be desirable that in case of the Virtual Travel Agency, the (automatic) agents would not ask the same questions when you meet them the second time, but instead would be prepared to deliver customised services. The Internet front-end should provide services for personalisation. In the same way, if a group of users enters the application for the second time, they should find their personal settings in the virtual environment as they have left it. This is called persistence of the application.

Below is a list of wishes a user of the shared Virtual Travel Agency might have. These wishes should be addressed in view of the limitations of the application:

- If a user has arranged a meeting with friends in the VTA, the user expects to be in the same instance of the application as his friends.
- If a user wants information on a specific holiday destination, e.g., Rhodes, the user expects to enter the VE for this area quickly, so that he needs not wander around a large environment to find his area of interest.

- If a user wants to meet other users in the CVE with similar interests, he wants those users to be present, and no other users he is not interested to meet.
- If a user comes back to the VTA after a first visit, and he has asked questions before to some agent, he would like this agent to remember him and his previous questions.
- If a user has specific questions or issues he wants to investigate in the VTA, it would be nice if the agents could be prepared for his wishes.
- The user wants to be able to have a personal avatar, and when he enters the application again, he expects his avatar to be the same as last time.

Some of these wishes do not need to be addressed by the Internet front-end, but can be addressed by the CVE application itself. The wishes addressed by the Internet front-end do not change during the course of a session, such as the choice of an avatar. The preferences of the user can be recorded in a profile. The user should preferably have the ability to change some values of his profile. The profile can then be used to retain the user's preferences for a next session, or to compare them with other profiles, which can be used to form groups, for instance.

5.2.2 Avatar choice

The user is able to customise his own avatar in terms of size, color, texture and shape. As long as the body hierarchy (number of joints) does not change, the avatar can be customised to the user's wishes.

5.3 Communication

In order to exchange information between users, some form of communication has to be supported. The most important form of communication is speech. Besides talking, users need to exchange information such as short text messages, images, video, audio, documents and perhaps software and geometries. Users should be able to download information in their own computer environments so that they can process it off-line.

As in a 2D-window environment, *icons* can be used to identify the type of information. Icons are 3D objects that contain information in the form of images, video, audio, documents, etc. Exchange of information can be realised by simply handing over objects. However, to distribute information over multiple users, and to address users who are not in the same virtual space, a communication system, which consists of messages, senders, receivers and zones, is introduced.

To initiate communication metaphors are introduced. The keyboard is general enough to allow all possible commands, but the metaphors should also show a (novice) user the possible actions, depending on the current situation and the user's role (e.g., group member, individual, traveler, operator). The Personal Data Assistant (PDA) is a metaphor which could be the platform for many functions, including group functionality and communication.

Since the Virtual Travel Application is primarily intended for desktop users, the PDA could be visualised as a set of menus and associated 2D windows. Windows represent groups of related functions, and users may choose to have several windows active, corresponding to their needs. For instance, a window for communication functions could be visible on the desktop at all times.

Besides a 2D appearance on the screen, the PDA may be presented as a 3D object that is part of the application. This object represents some kind of *toolbox* with a set of buttons. The appearance of the toolbox changes according to the functionality that is offered. For instance, if a user is member of a group, a button may appear to enable communication with the group. The PDA will possibly be realised through a combination of 2D windows on the screen and a 3D toolbox object inside the virtual environment.

For communication, we can identify the following sub-functions:

- A user must be able to *identify receivers* for the information,
- then he must be able to *compose the message*,
- next, *send the message* to the intended receivers,
- and also must be able to *receive messages* from others.

All communication facilities, including communication to the outside world, are provided by the PDA. These facilities are discussed below.

5.3.1 Identify receivers

Receivers can be other users or objects intended for display (e.g., blueboards). Receivers can be grouped, e.g., all users in a group, or all receivers in the same zone. Possible receiver objects for the Virtual Travel Agency are dependent on the medium of the message (audio, video, text) and have been discussed in earlier WP2 deliverables

All receivers in the application are labeled, i.e. a name tag is shown which can be used to address the receivers. For instance, all avatars show the name of the user. A separate window of the PDA can display the names of all receivers in the present zone or in the entire environment.

The identification of receivers can also be done on request of the user by the following means:

- Upon typing specific text commands like ?who, ?Who, ?zone, ?mygroup, ?groups, ?group=[name] the application identifies receivers, either by returning a text message to the user, or by showing the names on a separate window.
- By selecting a certain receiver object in the virtual environment, its name will be returned to the user, or highlighted in the PDA window.

To address communication, a receiver or group of receivers must be selected. This can either be done by selecting receivers in the virtual environment, or by selecting receivers in a list of possible receivers in the communication window of the PDA. Using the PDA, a group of receivers may be selected for communication within a certain group or certain zone.

5.3.2 Compose the message

Messages may contain text, audio, video, or objects. Composing text messages can be done using the keyboard. A syntax will be developed to distinguish between commands typed in at the keyboard, and the contents of messages. Commands are special key combinations (such as CTRL-X) or text preceded by certain characters (such as "?group"). In this way, commands will be used to

- select groups of receivers;
- indicate emotions or gestures.

In order to store and receive objects, the **Personal Data Container (PDC)** is introduced. A user can import data from the environment in the PDC and the other way around. The PDC represents the virtual memory of a user. It will be accessible and controllable by the user whenever he or she wants. In the beginning the PDC is empty, except, perhaps, for some personal object containing personal information. If a user encounters something in the CVE that he or she finds interesting the object may be stored in the PDC. If another user is interested in some personal information, this information may be copied from one PDC to the other. A user can set certain access rights to his information. This makes it possible, for example, to automatically exchange certain information between users to support the gathering of mutual knowledge and establish common ground for discussion.

5.3.3 Send the message

The message must be sent either *explicitly* or *implicitly* to one or more receivers. Explicit sending can be done by selecting receivers as described above. Sending messages implicitly can be done by using social conventions and "laws of nature" from the real world: only people close to you can hear you whisper, only people in the same room can hear you talk, etc. Two examples of implicit sets of receivers are all receivers in the current zone and all users within the current group. These forms of group addressing will be supported by the Virtual Travel Agency application.

5.3.4 Receive messages

If messages could contain text, audio, and video this has implications for the way it is displayed to the user.

- Text requires a special window (desktop version) or receiving area in the virtual environment. A personal text receiver could be realised using subjective views.
- Images and video can be shown to the user by a special window of the PDA, or by a special receiver within the virtual environment.
- Audio generated by objects (such as the opening of a door) can only be heard if the user is in the same zone. However, audio communication between a group of users is possible regardless of the position of the users within the application (see also spatialised sound below).

5.3.5 External communication

Communication from the application to the outside world can be realised through e-mail. E-mail messages can be sent to and from the VTA. In this way communication with off-line users can be established. E-mail can also be used to send information (such as documents) from the virtual environment to the personal computer workspace.

5.3.6 Spatialised sound

Spatialised sound means modifying the sound source to make the user aware of its position in the virtual world. For example, a user may approach a sound source and be aware of its volume growing as he approaches the object that generates the sound. If the user were to turn away from the object such that it would then be on the right of the user's head, then he would hear more audio in his right ear than in his left. The user would be able to deduce the position of the audio source from its intensity and varying audio volume across both ears.

5.4 Group functionality and subjective views

The VTA application is strongly focused on the support of groups of users in a shared virtual environment. Besides communication, also functionality such as group navigation, group decision support and subjective views are required to support group decision making in large virtual environments. Group functionality is provided by the PDA.

5.4.1 Group support tool

Users can become a member of a certain group in two ways:

1. Implicitly: E.g., a user stands within hearing distance of another avatar and thus forming implicitly a group. If a user decides to move to a different area, the user will not be a part of this group anymore. Users present in the same virtual zone form a group and are able to communicate with each other.
2. Explicitly: E.g., a user can become a member of an existing group or can assign himself and other users to a new group. This is facilitated by the Group Support Tool of the PDA.

Using the Group Support Tool all users have the ability to:

- Identify the current groups in the virtual environment.
- Identify the members in these groups.
- Obtain any public information from a certain group.

Based upon this information, a user in the virtual environment could ask to join an existing group or create a new group. Users who are already group members should be able to let a person join the group, or to disconnect from the group. Permission to let a person join the group could be restricted to the group leader or the creator of the group.

A user can be a member of more than one group at the same time and is able to communicate with both groups. However, for functionalities such as group navigation, a user can only be active within one group.

5.4.2 Group decision support tool

A group decision support tool could be used to simplify group decision making. Decisions related to group membership are whether a certain user may join the group or become the group leader. Other decisions are related to the task of the group in the application. A group decision support tool should:

- Give an overview of the different opinions;
- Give an overview of the positions (or votes) of group members for the different opinions;
- Allow individual users to express their opinions and position;
- Allow individual users to make their decision;
- Announce the group decision to the group.

The decision support tool is intrinsically related to the communication facilities. It only needs to be present when a user is part of a group, or when a user intends to join a group. The tool can be regarded as part of the PDA.

5.4.3 Group navigation

There are several ways to navigate through the virtual environment, either individually or as a group. In order to navigate, a user must be able to

- identify his location in the virtual environment (global map),
- search for locations, objects or users of interest (search functionality),
- move through the virtual environment,
- either on his own (using the map as reference),
- or automatically guided by the system (directive manipulation),
- or “jump” from his present position to the desired location (teleport).

A group of users may navigate through the virtual environment using some kind of transport vehicle (e.g., flying carpet) or another mechanism that keeps group members together, such as a virtual rubber band.

5.4.4 Personal Distancer

The personal distancer is a facility that helps avatars in a group not to stand in each other's personal space, as this is commonly perceived as rude. In present day virtual environments it is rather complicated and cumbersome to navigate back to a ‘proper’ position every time somebody else's avatar moves, or arrives. The personal distancer positions the bodies of avatars in a group in such a way that every user can easily look every other avatar in the face. In combination with group navigation functions such as ‘return to leader’ it provides a natural way for a group to come together. It can be implemented as a control mechanism, which is activated explicitly, e.g., by the PDA.

5.4.5 Global map

The global map is available for the user at each location in the virtual environment. Hence, it is part of the PDA. The global map can be used to find objects or a specific person in the citizen application (oneself, group members, receivers, interesting objects). The map is an abstraction (3D framework) of the entire application. A search facility can be combined with the global map. The result of searches can be presented on the global map.

The global map can also be used to support *teleporting*. By selecting a specific spot on the global map, the user will be placed on the location that corresponds to the selected spot. This notion of teleporting can also be used to navigate to the group leader, for instance.

5.4.6 Group teleporting

Group teleporting happens in the following context: members of the group have decided that one of the group members is the guide (or leader) of the group. If group teleporting is activated, this means that wherever the leader teleports to, the group follows.

5.4.7 Directive manipulation

Directive manipulation can be regarded as teleporting initiated by another user or by the application. For instance, directive manipulation is used by an operator to direct a traveller to a certain point of interest. A user is able to grant permission for such a teleport operation to another user. Again, the global map is used to initiate the teleport.

5.4.8 Search functionality

As opposed to group navigation and group decision support, search functionality is more application specific. A user might for example be interested in the location of the bars on Rhodes. The result of such a query can be presented in two different ways:

- 1) In case objects representing bars are not part of the current view on the virtual environment, the bars can be added to the view.
- 2) In case objects representing bars are already part of the current view, the bars will be highlighted.

In both cases *subjective views* will be applied to guarantee that only users who are interested in the information will experience a change of view on the virtual environment.

5.4.9 Subjective views

Subjective views are used to customise the view on the virtual environment to the wishes of a particular user or user group. As an example, the bars could be made visible. It should be possible to adapt the view on the environment on a personal basis or for a group of users, but this should not influence the view on the environment for users outside the group.

5.5 Crowd Control and Autonomous Agents

5.5.1 Crowd control

A crowd is defined at the level of each world. Bodies, which will be involved in the crowd, are added to the world as classical 3D objects. In order to give the capability to retrieve involved bodies at the level of the final application, they are tagged by using specific names. It is permitted to keep pointers to them in order to send or receive information to or from the external crowd controller to animate them. A set of points of interest is needed in order to determine the way the crowd will act. Crowds will go from one point to another. Finally, the number of groups with the number of bodies involved in each can be defined, which permits to have a good control of what the crowd will look like.

Crowds can be used in the Virtual Travel Agency application mainly to populate visited areas, which are supposed to be public. Adding autonomous crowds will give the opportunity to make this virtual application much more realistic than classical 3D virtual worlds which usually do not contain virtual humans.

5.5.2 Autonomous agents

An autonomous agent seems to behave by himself according to surrounding events. It is a classical body, animated by the body controller, but instead of being controlled by a real user, communication with the external process will be made through a specific part of the code of the application. In the case of the Virtual Travel Agency, according to the function of each autonomous character (tour guide, hotel operator...), a specific internal controller will manage application events in order to define autonomous agent behaviour by launching specific animation in a given situation. This can be implemented by using the communication interface described in COVEN deliverable D4.4.

The provision of autonomous agents in the VTA application is considered important for the following reasons:

- If autonomous behaviour is built into the virtual agents, then the human operator of the VTA will be required to do less work in accommodating requests by the users of the system. For example, autonomous agents can undertake the tasks of providing tour guides, providing hotel information and selling souvenir items.
- Including autonomous actors that can interact with real human participants increases the real-time interaction with the environment contributing to an increased sense of presence.

In the following paragraphs we examine in detail the tasks performed by the autonomous actors and the requirements for accomplishing these tasks.

In general, the behaviour of autonomous agents will be based on: (i) perception of the environment, and (ii) communication with the human users. Simulating autonomous virtual actors within virtual worlds is a challenging new field where a lot of research is currently performed.

5.5.3 Automatic Tour Guide

Automatic tour guides can be implemented in the Virtual Flight Zone as well as the Acropolis of Lindos Zone. In the Virtual Flight Zone, an autonomous actor will be implemented to provide a guided tour of the island. The provided tour guide could have several manifestations based on different requirements imposed by the human actor, such as:

- show me the major cities
- show me the best hotels
- show me the cheapest hotels
- show me the best beaches
- show me the archaeological sites

Similarly, in the Acropolis of Lindos Zone, an autonomous actor will be present to provide a guided tour through the archaeological site. Tasks performed by this autonomous agent include a tour of the various buildings and an exhibition of various findings such as vases, weapons, etc.

5.5.4 Hotel Operator

The Hotel Operator autonomous actor can be used to accomplish the following tasks:

- Provision of hotel information such as pricing, room accommodations, distance to beach, interesting events, etc.

- Orientation to the hotel and its services

5.5.5 Souvenir Vendor

A Souvenir Vendor autonomous actor can be used to provide information on the various souvenirs available, negotiate prices, etc.

To accomplish the tasks described above, autonomous actors should be implemented with sufficient behaviours for motion as well as appropriate mechanisms for interaction with the environment and the human users. With respect to body motion, the autonomous actors should be able to accomplish the following:

- body motion: walking and body gestures while speaking
- body deformations while moving

For interacting with the environment and the participants, the autonomous actors should be able to accomplish the following:

- Animation of the face for simulating speech and emotions
- Navigation in the virtual world. For this we need a mechanism to pass the necessary information from the environment to the autonomous actor to allow path searching and obstacle avoidance.
- Virtual audition. The autonomous actor should be able to access sound source information in a reliable way and react accordingly.
- Object manipulation. This involves reaching out for objects and grasping them. For example, the Lindos tour guide should be able to pick up an exhibit and show it to the audience.

5.6 Embedded Application Functionality

5.6.1 VTA Zone

This is where the user starts out when he enters the application. The VTA zone shows a travel agency. Four counters can be seen behind which the travel operators are situated. A traveller entering this zone can walk around and address one of these travel operators to obtain information. By interacting with one of the travel operators the virtual tourist can

- (i) obtain information about various holiday packages
- (ii) obtain information about the destination and the cost of these packages
- (iii) (in a future version) actually book a trip and pay for it.

One door (with a picture of Rhodes on it) leads to the meeting room for Rhodes. The other doors are reserved for other destinations that have not yet been included in the application.

On the walls there are posters of holiday destinations. Next to the entrance of the VTA is a so-called "blue board". Blue boards are used for text based communication in the application; for more detailed information about the blue boards see Next to the door leading to the Rhodes meeting room there is a clock showing the actual real-world time.

5.6.2 The Rhodes Meeting Room Zone

The second zone is the meeting room which corresponds to a particular travel destination. In the present application Rhodes is the only holiday destination. People who are interested in Rhodes can get all kinds of information about the island in text and audio-format. Putting a CD into the CD-ROM player on the table starts a slide-show showing pictures of the various attractions of Rhodes. The slide-show is shown on a large screen on the wall and is supported by audio explaining the slides. When no slide show is running the projection screen shows a map of Rhodes. Like in the VTA zone there is a blueboard for text based communication between travelers. Finally there are posters on the walls telling something about the beauty of the island.

An important purpose of the meeting room is offering travelers the opportunity to meet each other. Tourists that are present in the meeting room can communicate, and if they wish they can form a group and go for a virtual Rhodes-visit together. In order to go on a virtual visit they have to leave the meeting room and move to the Virtual Flight Zone. This can be done using the teleporting device, which is situated in the far corner of the meeting room. The teleporting device has the ability to move a number of users from one point in space immediately to another point in space. The teleporter is opened by pressing the button on the front of the device. When the teleporter opens users who want to form a group can enter the teleporter. Once all users are inside the teleporter one person can press the green button to start the teleporting sequence. The door of the teleporter will close and a

green light beam will show up. At this moment all group members are moved (teleported) to the Virtual Flight Zone together.

5.6.3 Virtual Flight Zone Functions

After teleportation the virtual tourists enter the Virtual Flight Zone. Here they can use a flying carpet to fly over the island of Rhodes, which is represented by a 3-dimensional, thematic map. This map was created using a satellite height image. This map was overlaid by another satellite image with a resolution of 30x30 meters showing where the woods are and the rivers run. The map provides a realistic picture of the land-cover and land-use of the destination.

Important locations on the island, e.g. where hotels or interesting archeological sites are located, are highlighted by large pillars, rising high above the surface of the island. Attached to the hotel pillars are symbols providing information about the hotel connected to the pillar:

- The coins symbol, below the star, denotes ranges of room rates
- The umbrella symbol, below the coins, denotes distance from nearest beach
- The telescope symbol, below the umbrella, denotes the quality of the room views

Each of these symbols has one of the colors yellow, orange and red. Yellow indicates low prices/quality, orange stands for medium and red is for fine (but expensive) hotels. For example, a yellow umbrella means that there is no beach near the hotel, orange binoculars indicate that the view from the hotel rooms is just nice and red coins identify an expensive hotel.

By dropping (selecting) the hotel flag, the user can see a picture of the hotel on the top right corner of the window screen. In parallel, the application invokes Netscape using the corresponding hotel URL and takes the user to the hotel WWW home page.

5.6.4 Lindos Zone

This zone presents a model of Rhodes' most important archaeological site: the Acropolis of Lindos. The user can enter this zone via the column that depicts the site on the map of the island. Lindos is located 55 km south to the city of Rhodes. The Acropolis is located on a rocky hill above the village of Lindos. It has a triangular shape covering an area of 8400 m². The various buildings at the top of the hill belong to various historical periods from the 6th century B.C to the 14th century A.D.

The Acropolis of Lindos was modeled in its ancient status. In reality, however, there are only a few ruins remaining. In this respect, VR offers an additional functionality to the user: being able to see and navigate in an archaeological site as it was in its ancient state. There are a number of audio presentations dotted around the ruins which are activated by pressing buttons on signposts.

5.6.5 Hotel Mare

Currently the citizen application provides hotel information based on an abstract representation scheme where quick information about each hotel can be obtained. A pole bearing a flag with the name of the hotel indicates the location of the hotel. A number of symbols on the pole provide information regarding the price range, the hotel category, the distance to the nearest beach, and the room view. A possible extension would be to model one of the hotels to demonstrate the capabilities offered by VR technology with respect to virtual travel and vacation planning. The model of the hotel will be implemented as a new part of the virtual world where participants will be able to enter and explore the hotel and its surroundings. They will be able to visit a room and explore its structure, find out where the room is located with respect to other hotel facilities, get an idea of the room view and communicate with an autonomous hotel operator to get information about the hotel.

5.6.6 Souvenir Shop

A souvenir shop can be implemented as a place where users will be able to find a number of virtual representations of items that they can obtain through electronic payment. A model of a souvenir shop will be created and will be populated with models of important objects related to Rhodes (e.g. archaeological findings such as small statues and vases, icons, etc.). A souvenir vendor in the form of an autonomous actor will be there to provide information on the exhibits and to negotiate prices. The ordering and payment of the actual object will be implemented in connection with the WWW.

5.6.7 Book a trip

Booking of an actual trip can be implemented in connection with the WWW. The traveler will be asked to fill in a questionnaire where he will be presented with a number of options to choose from. He will be able to answer these questions using the information gathered from the virtual trip and stored in his/her PDC. The questionnaire

will be linked to a database where the completed information will be stored and later accessed by the travel operator to make the booking.

6 Example Inspection – Interaction Cycle Analysis

The tasks, system functions and 3D objects identified in the Task Analysis (Section 5), should be linked to one or more of the 6 interaction cycles as described in Section 2.1.

N.B.: In the following table, lines that are greyed out indicate planned functionality that has not yet been fully implemented or has not been used in the on-line trials.

Class of Function	Sub-Function	Description	Indicate Cycles
Internet Front-end	Access	Logging on to service	2. Normal task action Cycle 2D
	Personalisation	From user -> Travel Service Provider: Personal profile	2. Normal task action Cycle 2D
	Persistence	AS PROVIDER, DON'T ASK THINGS FOR A SECOND TIME. FROM TRAVEL SERVICE PROVIDER -> USER	1. System Initiative Cycle
Avatar	Avatar choice	Menu based avatar selection	2. Normal task action Cycle 2D
	Avatar modification	Menu based colour selection	2. Normal task action Cycle 2D
COMMUNICATION	Identify receivers	Identify receiver from list	2. Normal task action Cycle 2D
		Select receivers in view	6. Collaboration Cycle
		Proximity sets receivers	1. System Initiative Cycle
	Audio	Push to talk control	2. Normal task action Cycle 2D
	Send message	Composition	2. Normal task action Cycle 2D
		Specify mode (whisper, thinking...)	2. Normal task action Cycle 2D
	Receive message	Automatic (in new area on screen)	1. System Initiative Cycle
		Message blocking	2. Normal task action Cycle 2D
	External communication		
	Spatialised sound		1. System Initiative Cycle
Group functionality	Group formation tool	Join group	2. Normal task action Cycle 2D
		Leave group	2. Normal task action Cycle 2D
	Interacting with Global map		2. Normal task action Cycle 3D

	Directive manipulation	User teleports other users	6. Collaboration Cycle 2. Normal task action Cycle 2D
	Search functionality		2. Normal task action Cycle 2D
Navigation	Identify a specified position		4. Goal Directed Exploration Cycle
	position overview	Global map	1. System Initiative
	move through the VE	As a group: decide destination	6. Collaboration Cycle
		As a group: travel together	4. Goal Directed Exploration Cycle
		As single: teleporting	3. Normal task action Cycle 3D
		As single: flying directed	4. Goal Directed Exploration Cycle
		As single: flying undirected	5. Exploratory Browsing Cycle
		Guided	1. System Initiative Cycle
Subjective views		Activation of subjective views	2. Normal task action Cycle 2D
Crowd control and Autonomous agent	Automatic tourguide	Agent Triggered automatically	3. task action Cycle 3D 1. System Initiative Cycle
	Hotel operator	Agent Triggered automatically	3. Normal task action Cycle 3D 1. System Initiative Cycle
	Souvenir vendor	Agent Triggered automatically	3. Normal task action Cycle 3D 1. System Initiative Cycle
Application Functions	Door	Open door	3. Normal task action Cycle 3D
	CD Player	Select and play presentations	3. Normal task action Cycle 3D
	Teleporter Door	Open teleporter	3. Normal task action Cycle 3D
	Teleporter Controls	Operate teleporter	3. Normal task action Cycle 3D
	Guide Posts	Activate presentation	3. Normal task action Cycle 3D
	Carpet Console	Control carpet	3. Normal task action Cycle 3D
	Hotel Signs	Identify icons	3. Normal task action Cycle 3D
	Book a trip		3. Normal task action Cycle 2D
	Order a souvenir	Identify purchase Order	3. Normal task action Cycle 3D 2. Normal task action Cycle 2D
	Book a hotel room		3. Normal task action Cycle 3D

			2. Normal task action Cycle 2D
--	--	--	---------------------------------------

7 Example Inspection – Cognitive Walkthrough

For each task, system function and 3D object, identified in Section 6, a number of Inspection questions exists, which are used in the CW. The Inspector steps through the task, asking the questions belonging to the interaction cycles. For each question the answer is noted down in the column 'Potential Problem and Design Solutions'. The Inspectors are encouraged to write down any design solutions they can think of in this same column. The next column 'Severity Rating', is used to note a number between 0 and 4 to indicate the severity of the problem found. The following rating scale is used for rating the severity of usability problems:

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.

The final column is used to give a reference number to the problems, so that in the subsequent report (Section 8) it will be easier to refer to specific problems.

7.1.1 System Initiative Cycle

Only one task, Receive Message, was found to belong to the System Initiative Cycle. It is reported below.

Function	Generic Task Scenario for System Initiative	Potential Problem and Design Solution	Severity Rating	Ref. Numb
Receive Message				
	i: Is it clear to the user that the system has taken control?	Yes	0	
	ii: Can the user resume control at any point and is the appropriate action clear?	Message received wipes over previous messages. No way to retrieve them. Add a HUD control to scroll through messages (makes this non-system initiative).	3	1
	iii: Are the effects of system actions visible and recognisable?	Yes	0	
	iv: Are the system actions interpretable?	Yes	0	
	v: Is the end of the system action clear?	Yes	0	

7.1.2 Normal task action Cycle 2D

Four tasks were found to belong to the Normal Task Action Cycle 2D; Identify Receivers, Send Message: Composition, Send Message: Specify Mode, Audio: Push To Talk. These results from the Inspection in these tasks are reported below.

Function	Generic Task Scenario for a Normal Task Action 2D	Potential Problems and Design Solutions	Severity Rating	Ref. Numb
Identify Receivers				
	i: Will the users be trying to produce whatever effect the action has?	Not obvious that there is a way to identify receivers. The user might assume they can only send messages to people with name tags.	2	2

		The interface could directly ask you who to want to send it to (and default to "world"). That is, if no receiver is identified explicitly, the interface offers a list of possible receivers.		
	ii: Will users be able to notice that the correct action is available?	Limited. Help is supplied only on request. (Though incorrect – say “ask me “who” not “?who””. The world is called “wereld” in the identity list. “Temporary” should not be identified.) Support for the other identification methods (view and proximity) is essential. See i as well	4	3
	iii: 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?	Yes. Command name is representative.	0	
	iv: After the action is taken, will users understand the feedback they get?	Names are a bit obscure (bb1) more elaborate names required	2	4
Send Message: Composition				
	i: Will the users be trying to produce whatever effect the action has?	Yes	0	
	ii: Will users be able to notice that the correct action is available?	Only if they have read the manual.	1	5
	iii: 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?	One should expect typing to result in a message being sent. Unclear when the message is sent (only after return is pressed).	1	6
	iv: After the action is taken, will users understand the feedback they get?	No. No notification that the message as sent was received by anyone. Prompt “OK” to sender.	4	7
Send Message: Specify Mode				
	i: Will the users be trying to produce whatever effect the action has?	Not at all obvious that private messages are possible	2	8
	ii: Will users be able to notice that the correct action is available?	No, is a modifier on the text message command “receiver:message”. Addressed by adding further receiver identifier methods.	2	9
	iii: 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?	Immediate feedback should say “sending private message to bb1” when the user types “bb1:” or indicate an error on the mode.	4	10
	iv: After the action is taken, will users understand the feedback they get?	No feedback on mode error. No feedback on success.	4	11
Audio: Push To Talk (Using RAT alongside the dVS session).				
	i: Will the users be trying to produce whatever effect the action has?	No. Not obvious.	3	12

	ii: Will users be able to notice that the correct action is available?	The button is not clearly labelled and is a separate window. (Not addressed in COVEN/dVS platform. Is addressed in COVE/Dive).	3	13
	iii: 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?	Probably	1	14
	iv: After the action is taken, will users understand the feedback they get?	Button just changes shading. Audio feedback would be useful.	3	15

7.1.3 Normal task action Cycle 3D

Seven tasks were identified as belonging to the Normal Task Action Cycle 3D; tasks Move As Single: Teleporting, Door, CD Player, Teleporter, Guide Posts, Carpet Controls, and Hotel Signs. Six of these tasks involved objects; , Door, CD Player, Teleporter, Guide Posts, Carpet Controls, and Hotel Signs. Two tasks addressed overlapping issues; Move As Single: Teleporting, and Teleporter.

Function	Generic Task Scenario for a Normal Task Action	Potential Problems and Design Solutions	Severity Rating	Ref. Numb
Move As Single: Teleporting				
	i: Can the user form or remember the task goal?	Not obvious that this is possible.	2	16
	ii: Can the user specify an intention of what to do?	Possibly not. Requires the user to press "4" before interface is activated. Should be offered on a HUD control.	3	17
	iii: Are the objects or part of the environment necessary to carry out the task-action (users new intentions) visible?	Yes once the key has been pressed	1	18
	iv: Can the objects necessary for the task action be located?	Yes once the key has been pressed.	0	
	v: Can the users approach and orient themselves to the objects so the necessary action can be carried out?	Yes	0	
	vi: Can the user decide what action to take and how?	Only if they know the names of the zones. Graphical representation would be preferred.	3	19
	vii: Can the user carry out the manipulation or action easily?	Yes	0	
	Viii: Is the consequence of the users action visible?	Yes	0	
	Ix: Can the user interpret the change?	Probably as long as they recognise the new place. Audio feedback would be useful.	1	20
	x: Is it made clear to the user what the next correct/needed action could be?	N/A.	0	
Door				

	i: Can the user form or remember the task goal?	Yes	0	
	ii: Can the user specify an intention of what to do?	Yes	0	
	lii: Are the objects or part of the environment necessary to carry out the task-action (users new intentions) visible?	Yes	0	
	iv: Can the objects necessary for the task action be located?	Yes	0	
	v: Can the users approach and orient themselves to the objects so the necessary action can be carried out?	Yes	0	
	vi: Can the user decide what action to take and how?	No. The door should react to proximity since it is not obvious that the handle need to be selected and picked. Especially since the door itself can be selected but this does nothing.	3	21
	vii: Can the user carry out the manipulation or action easily?	No. See above	3	22
	viii: Is the consequence of the users action visible?	Yes	0	
	ix: Can the user interpret the change?	Yes	0	
	x: Is it made clear to the user what the next correct/needed action could be?	N/A	0	
CD Player				
	i: Can the user form or remember the task goal?	Not totally obvious that there are presentations, though the controls don invite exploration. Titles could be "Show General Info". Map on screen could be replaced by an introductory slide. E.g. "Press slide show controls to see presentations".	3	23
	ii: Can the user specify an intention of what to do?	Once they have discovered the interface then yes.	1	24
	lii: Are the objects or part of the environment necessary to carry out the task-action (users new intentions) visible?	Yes though the buttons could be larger and labelled.	1	25
	iv: Can the objects necessary for the task action be located?	Yes since they are in front of the presentation screen.	0	
	v: Can the users approach and orient themselves to the objects so the necessary action can be carried out?	Yes	0	
	vi: Can the user decide what action to take and how?	Yes	0	
	vii: Can the user carry out the manipulation or action easily?	Objects are tricky to click since they are quite small.	1	26
	viii: Is the consequence of the users	There is a considerable time lag.	2	27

	action visible?	Immediate audio or text feedback should be given.		
	ix: Can the user interpret the change?	Yes	0	
	x: Is it made clear to the user what the next correct/needed action could be?	Yes	0	
Teleporter				
	i: Can the user form or remember the task goal?	They will probably know they can go to Rhodes, but they do not necessarily know how to get there. Each slide show should finish by telling the user about the teleporter. The teleporter should be labelled with a sign "Rhodes Teleport".	3	28
	ii: Can the user specify an intention of what to do?	Yes	0	
	lii: Are the objects or part of the environment necessary to carry out the task-action (users new intentions) visible?	Yes the object is big enough to be seen.	0	
	iv: Can the objects necessary for the task action be located?	They are hidden behind the door. Remove the door!	2	29
	v: Can the users approach and orient themselves to the objects so the necessary action can be carried out?	This tricky since the teleporter is quite small. Instant teleportation on collision would fix this since you would just have to pass through the space.	2	30
	vi: Can the user decide what action to take and how?	Yes.	0	
	vii: Can the user carry out the manipulation or action easily?	No – the buttons are hard to see when inside the teleporter. Maybe teleportation should be automatic when entering teleport (or on a short countdown).	3	31
	viii: Is the consequence of the users action visible?	Yes	0	
	ix: Can the user interpret the change?	Yes, though the island is difficult to see.	1	32
	x: Is it made clear to the user what the next correct/needed action could be?	N/A	0	
Guide Posts				
	i: Can the user form or remember the task goal?	Yes	0	
	ii: Can the user specify an intention of what to do?	Yes	0	
	lii: Are the objects or part of the environment necessary to carry out	Yes	0	

	the task-action (users new intentions) visible?			
	iv: Can the objects necessary for the task action be located?	Yes	0	
	v: Can the users approach and orient themselves to the objects so the necessary action can be carried out?	Yes	0	
	vi: Can the user decide what action to take and how?	No. The only button is "Stop". This should toggle between "Play" in green and "Stop".	3	33
	vii: Can the user carry out the manipulation or action easily?	Yes	0	
	viii: Is the consequence of the users action visible?	Button should react in some way. (Colour/Raise)	2	34
	ix: Can the user interpret the change?	Yes	0	
	x: Is it made clear to the user what the next correct/needed action could be?	No (the button state does not change).	1	35
Carpet Controls				
	i: Can the user form or remember the task goal?	Some explanation of the carpet is required. "Drop" should read "Press". Not sure why you need to approach first. Suggest removing that function.	2	36
	ii: Can the user specify an intention of what to do?	Yes, though "Approach" isn't very specific (it's a big island).	1	37
	lii: Are the objects or part of the environment necessary to carry out the task-action (users new intentions) visible?	Yes	0	
	iv: Can the objects necessary for the task action be located?	Yes	0	
	v: Can the users approach and orient themselves to the objects so the necessary action can be carried out?	Yes	0	
	vi: Can the user decide what action to take and how?	Yes – but only since nothing else works until "Approach Island" is pressed. Not sure what the "3D Map" button does. Surely "Hotels" and "Sites" should be independent toggles, and labelled "Show Hotels"/"Hide Hotels" etc. In particular you can not turn off one of Hotels or Sites if they are both on/	4	38
	vii: Can the user carry out the manipulation or action easily?	Yes	0	
	viii: Is the consequence of the users action visible?	More feedback message are required on button presses. "Hotels are on", "You are now over Rhodes".	3	39
	ix: Can the user interpret the	Yes.	0	

	change?			
	x: Is it made clear to the user what the next correct/needed action could be?	Yes	0	
Hotel Signs				
	i: Can the user form or remember the task goal?	Yes	0	
	ii: Can the user specify an intention of what to do?	Not really – it is not obvious that the flags are active. Some feedback in the text box, or a picture on the HUD showing where to click.	2	40
	lii: Are the objects or part of the environment necessary to carry out the task-action (users new intentions) visible?	Yes	0	
	iv: Can the objects necessary for the task action be located?	Yes	0	
	v: Can the users approach and orient themselves to the objects so the necessary action can be carried out?	Yes, except that some of the picture boards are end on to the user from some positions. Maybe billboards would be better.	2	41
	vi: Can the user decide what action to take and how?	Yes	0	
	vii: Can the user carry out the manipulation or action easily?	The coin icons are rather small.	1	42
	viii: Is the consequence of the users action visible?	Yes	0	
	ix: Can the user interpret the change?	Yes	0	
	x: Is it made clear to the user what the next correct/needed action could be?	Yes	0	

7.1.4 Goal Directed Exploration Cycle

Three tasks have been identified as belonging to the Goal Directed Exploration Cycle; Navigation: Identify A Specified Position, Navigation: MoveThrough VE as a Group: Travel Together , Navigation: Move Through VE as Single: Flying Directed.

Function	Generic Task Scenario for Goal Directed Exploration	Potential Problems and Design Solutions	Severity Rating	Ref. Numb
	i: Does the user know where to start looking?	Yes	0	
	ii: Can the user determine a pathway towards the search target?	Yes	0	
	iii: Can the user execute movement and navigation actions?	Yes since it is a simple turn	0	
	iv: Can the user recognise the search target?	Most things are labelled, though it is possible to get lost in the mist. Constrain the participant to stay within a visible distance of the island.	1	43

	v: Can the user approach and orient themselves to the objects so the necessary action can be carried out?	Yes	0	
	vi: Can the user decide what action to take and how?	Yes	0	
	vii: Can the user carry out the manipulation or action easily?	Yes, we assume they have been tutored on basic navigation with the middle mouse button.	1	44
	viii: Is the consequence of the users action visible?	Yes	0	
	ix: Can the user interpret the change?	Yes	0	
	x: Is it made clear to the user what the next correct/needed action could be?	N/A	0	
Navigation: Move Through VE as a Group: Travel Together				
	i: Does the user know where to start looking?	Yes		
	ii: Can the user determine a pathway towards the search target?	Avoidance of other avatars can be a problem. Collision detection with other avatars would be useful.	2	45
	iii: Can the user execute movement and navigation actions?	Yes		
	iv: Can the user recognise the search target?	Target can be obscured by others.	0	
	v: Can the user approach and orient themselves to the objects so the necessary action can be carried out?	With a 1 st person view it can be hard to see where the other people are situated. A toggle to a group overview 3 rd person viewpoint would be useful.	1	46
	vi: Can the user decide what action to take and how?	Yes	0	
	vii: Can the user carry out the manipulation or action easily?	Yes	0	
	viii: Is the consequence of the users action visible?	Yes	0	
	ix: Can the user interpret the change?	Yes	0	
	x: Is it made clear to the user what the next correct/needed action could be?	N/A	0	
Navigation: Move Through VE as Single: Flying Directed				
	i: Does the user know where to start looking?	Yes	0	
	ii: Can the user determine a pathway towards the search target?	Yes	0	
	iii: Can the user execute movement and navigation actions?	Yes	0	
	iv: Can the user recognise the search target?	Yes	0	

	v: Can the user approach and orient themselves to the objects so the necessary action can be carried out?	Yes	0	
	vi: Can the user decide what action to take and how?	Yes	0	
	vii: Can the user carry out the manipulation or action easily?	Yes	0	
	viii: Is the consequence of the users action visible?	Yes	0	
	ix: Can the user interpret the change?	Yes	0	
	x: Is it made clear to the user what the next correct/needed action could be?	Yes	0	

7.1.5 Exploratory Browsing Cycle

Function	Generic Task Scenario for Exploratory Browsing	Potential Problems and Design Solutions	Severity Rating	Ref. Numb
Navigation: Move Through VE as Single: Flying Undirected				
	i: The user determines a pathway for movement.	Yes	0	
	ii: The user executes movement and navigation actions.	Yes	0	
	iii: The user recognises objects in the environment.	Yes	0	
	iv: The user interprets identity, role and behaviours of objects.	Yes	0	
	v: The user remembers important objects or locations.	Yes	0	
	vi: The user forms a mental map of the explored environment	Yes	0	

7.1.6 Collaboration Cycle

Only one tasks has been found for the Collaboration Cycle; Navigation: Move Through VE as a Group: Decide Destination. It is reported below.

	Generic Task Scenario for Collaboration	Potential Problems and Design Solutions	Severity Rating	Ref. Numb
Navigation: Move Through VE as a Group: Decide Destination				
	i: Can the user locate the other user(s)?	Yes.	0	
	ii: Can the user recognize the identity of the other user(s), tell the other users apart?	The name tags don't always face the user.	1	47
	iii: Is there an indication of mutual awareness?	No, text and audio messages are not confirmed. Sound can be very bad and no indication of the loss is given.	3	48
	iv: Are the actions of the other user(s) visible and recognisable?	No. Selection and picking are not indicated explicitly e.g. if an object is selected it is not possible to tell who	3	49

		selected it if several people are looking at that object. No pointing gestures except for looking at something		
	v: Can the user act on a shared object while keeping the other user(s) in view?	Only in certain situations.	0	
	vi: Can the user easily switch views between the shared object, other locations/object of interest and the other user(s) (sweep from one to the other)?	No since the navigation sweep must be specified manually. Maybe a viewpoint "hotspot" feature could be added, with spots corresponding to nearby users and active objects.	2	50
	vii: Can the user get an overview of the total shared space and all other users in it?	No, no overview map provided,	2	51
	viii: Can the user tell when there are interruptions in the attention of the other user(s) to the CVE?	No, avatar posture can convey the wrong impression.	2	52

8 Example Inspection – Report

In order to create a report of the Usability problems found by the Inspection, the original table created for the Inspection (Section 7), can be reordered to emphasize several aspects which are of concern to the designers, the inspectors, and the managers of the project respectively.

The table directly below (Table 1), lists the cycles, the associated functions of the application, the type of problem found (that is: which particular step of the interaction cycle the problem belongs to) with a short summary. Column 4 states shortly whether a design solution has been given by the Inspector or whether the entry is merely a problem description. It has to be noted that sometimes alternative design solution are incorporated in the design problem description. Finally, the severity rating and the problem reference numbers are stated. This table is of most direct concern for the designers of the application.

Cycle	Function	Type of Problem	Design Solution	Severity	Reference Number
System Initiative	Receive Message	ii) resume control	Yes	3	1
Normal Task Action 2D	Identify Receivers	i) trying to produce the action	Yes	2	2
		ii) correct action noticeable	Yes	4	3
		iv) feedback	Yes	2	4
	Send Message: Composition	ii) correct action noticeable	No	1	5
		iii) correct action identifiable	Yes	1	6
		iv) feedback	Yes	4	7
	Send Message: Specify Mode	i) trying to produce the action	No	2	8
		ii) correct action noticeable	Yes	2	9
		iii) correct action identifiable	Yes	4	10
		iv) feedback	No	4	11
	Audio: Push to talk	i) trying to produce the action	No	3	12
		ii) correct action noticeable	Sort of	3	13
		iii) correct action identifiable	No	1	14
		iv) feedback	Yes	3	15
Normal Task Action 3D	Move as a Single: Teleporting	i) task goal clear	No	2	16

		ii) specify intention	Yes	3	17
		iii) correct action visible	No	1	18
		vi) correct action identifiable	Yes	3	19
		ix) feedback	Yes	1	20
	Door	vi) which action and how	Yes	3	21
		vii) carry out action	Yes	3	22
	CD Player	i) task goal clear	Yes	3	23
		ii) specify intention	No	1	24
		iii) correct action visible	Yes	1	25
		vii) action diff	Sort of	1	26
		viii) feedback	Yes	2	27
	Teleporter	i) task goal clear	Yes	3	28
		iv) object not obvious	Yes	2	29
		v) approach	Yes	2	30
		vii) action diff	Yes	3	31
		ix) feedback	No	1	32
	Guide Posts	vi) which action and how	Yes	3	33
		viii) feedback	Yes	2	34
		x) next action clear	No	1	35
	Carpet Controls	i) task goal clear	Yes	2	36
		ii) specify intention	Sort of	1	37
		vi) which action and how	Yes	4	38
		viii) feedback	Yes	3	39
	Hotel Signs	ii) specify intention	Yes	2	40
		v) approach and orient	Yes	2	41
		vii) action diff	Sort of	1	42
Goal Directed Exploration	Navigation: Identify a specified	iv) recognise search target	Yes	1	43

	position				
		vii) carry out action	Sort of	1	44
	Navigation: Move through VE as group: Travel together	ii) determine path	Yes	2	45
		v) approach	Yes	1	46
Collaboration	Navigation: Move through VE as a group: Decide destination	ii) recognize others	Sort of	1	47
		iii) mutual awareness	Sort of	3	48
		iv) actions other users visible	No	3	49
		vi) switch view from A to B	Yes	2	50
		vii) overview	No	2	51
		viii) attention focus of other users	No	2	52

Table 1 Designers feedback report

In total 52 problems have been found.

The next table (Table 2) shows what severity ratings have been given to which functions of the application. First the cycle name is listed, then the function associated with the cycle. Next the severity's in descending order are given, and the final column contains the total of problems found within this particular function Inspection. At the bottom of the table - the final row of totals - describes how often problems of this particular severity have been found during this Inspection. N.B. The numbers in the cells refer the reference numbers of the problems found. This is of importance when identifying priority areas for redesign.

Cycle	Function	Severity 4	Severity 3	Severity 2	Severity 1	Totals
System Initiative	Receive Message		1			1
Normal Task Action 2D	Identify Receivers	3		2 4		3
	Send Message: Composition	7			5 6	3
	Send Message: Specifify mode	10 11		8 9		4
	Audio: Push tot talk		12 13 15		14	4
Normal task	Move as		17	16	18	5

Action 3D	Single: Teleporting		19		20	
	Door		21 22			2
	CD Player		23	27	24 25 26	5
	Teleporter		28 31	29 30	32	5
	Guide Posts		33	34	35	3
	Carpet controls	38	39	36	37	4
	Hotel Signs			40 41	42	3
Goal Directed Exploration	Navigation: Identify specified position				43 44	2
	Navigation: Move through Ve as group: Travel together			45	46	2
Collaboration	Navigation: Move Through VE as a group: Decide destination		48 49	50 51 52	47	6
Totals		5	15	16	16	52

Table 2: Usability and Designers feedback report.

The final table (Table 3) describes the type of problem found (which particular step within the cycle created a problem), the frequency with which this problem was found, and the severity rating which was given to this particular problem. N.B.: The first number in the cells refers to the problem reference number, the second number after the colon refers to the severity rating. The three final rows contain the total number of times this particular problem was found, below that the total score on the severity ratings (all severity rates added up), and the last row contains an average problem quotient, by deviding the total score on the severity ratings with the total number of times the problem was found. This average problem quotient indicates the impact of each problem found at that particular step in the Inspection Cycle. N.B. Not all cycles have 10 steps in total. And not all problems steps have completely corresponding problem descriptions, although they are largely structurally similar.

Cycle	Function	P i: Sev	P ii: Sev	P iii: Sev	P iv: Sev	P v: Sev	P vi: Sev	P vii: Sev	P viii: Sev	P ix: Sev	P x: Sev
System Initiative	Receive Message		1: 3								

Normal Task Action 2D	Identify Receivers	2:2	3:4		4:2						
	Send Message: Composition		5:1	6:1	7:4						
	Send Message: Specify mode	8:2	9:2	10:4	11:4						
	Audio: Push to talk	12:3	13:3	14:1	15:3						
Normal task Action 3D	Move as Single: Teleporting	16:2	17:3	18:1			19:3			20:1	
	Door						21:3	22:3			
	CD Player	23:3	24:1	25:1				26:1	27:2		
	Teleporter	28:3			29:2	30:2		31:3		32:1	
	Guide Posts						33:3		34:2		35:1
	Carpet controls	36:2	37:1				38:4		39:3		
	Hotel Signs		40:2			41:2		42:1			
Goal Directed Exploration	Navigation: Identify specified position				43:1			44:1			
	Navigation: Move through VEs as group: Travel together		45:2			46:1					
Collaboration	Navigation: Move Through VE as a group: Decide destination		47:1	48:3	49:3		50:2	51:2	52:2		
A: Total number of problems found		7	11	6	7	3	5	6	4	2	1
B: Total score on severity ratings associated with the problems		17	23	11	19	5	15	11	9	2	1
B/A: Total score on B divided by total score		2.43	2.09	1.83	2.71	1.67	3.00	1.83	2.25	1.00	1.00

on A.										
-------	--	--	--	--	--	--	--	--	--	--

Table 3: Feedback report for CVE Project managers, CVE usability evaluators and CVE designers

From the numbers in the lowest row of the Table 3 can be derived that, problem vi: "Can the user recognize what action to take and how", (similarly for the Collaboration Cycle: "Can the user switch from view A to B) causes relatively most concern. This could mean that this particular aspect of the design is generally problematic and might need special attention. However, for this calculation to work properly the steps in the cycles need to be synchronised more.

Additionally, an indication of the success of the design can be derived by comparing scores on 'Total row B'. By putting the scores from this row for several iterations of design and Inspection in comparison one can see whether the design is improving (scores must be going down), whether there are serious usability problems to be fixed (and thus sufficient time needs to be allocated), and whether the re-designs are successful.

9 Method Development

This method is still under development. The new Inspection method has now been applied once. It seems to have generated a substantial number of usability problems, which are easily referenced and re-read. The most difficult part of applying the Inspection to the application was allocating the functions, tasks, and objects to the most appropriate cycle. It remains to be seen how the designers find this improved method. In order to answer to the request of more specific re-design recommendations, and the need to educate the designers about good design choices (see Section 9.2 below) we have tried to integrate the Inspection questions at the level of design problem listings, and we have tried to give a design solution where possible. Clear indications of the availability of design solutions have also been attempted.

9.1 Influence of Previous Inspection Results

During the first iteration of the usability studies in the COVEN Project (Act 3.2) it was decided to explore the usefulness of the 2D CW and HE methods for 3D interfaces in order to extend the method to encompass the usability of CVEs. The standard CW and HE inspection based on Nielsen & Mack (1994), has been applied to the COVEN Platform by four independent usability inspectors. Collaboration was added as an item to the HE list of usability criteria, but no other specific items were added. See the Combined Inspection Report (COVEN, 1997e) for details. The Inspection was found useful in that it yielded many usability problems on system, application and interface level. The usability of the method itself has been evaluated (Tromp, 1997b), including the reception of the Inspection report by the designers themselves (Tromp, 1998), both are summarized in the next section (Section 9.2).

9.2 Results of Interviews with CVE Designers

In order to find out how to provide CVE design guidance we have 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 has been used to draw conclusions from the interviews. We looked at themes that recurred in all five interviews. This document outlines the themes, which need to be considered when trying to draw up any kind of guidance for VR designers who develop CVEs.

9.2.1 Introduction

Collaborative Virtual Environments (CVEs) are a novel application area of computing technology, demanding an understanding of human-computer interaction and computer mediated human interaction in virtual spaces. Standard usability engineering techniques and HCI design guidelines do not directly address the usability problems introduced by these new applications. The technology on which VEs are built is in its early stages, with in particular the human factors impact of its specific features still poorly explored (Durlach & Mavor, 1995), illustrated by the fact that significant usability problems with VEs have been found (Kaur, Maiden & Sutcliffe, 1998). VR designers have been found to lack a coherent approach to design, they also seem to lack an understanding of usability concepts underlying VEs and avoid using conventional HCI methods and guidelines (Kaur, 1997). From this we conclude that there is little knowledge available to VR designers on how to design VEs from a usability point of view, which can be traced back to the fact that there are no ready-to-use guidelines for VE design.

9.2.2 Methodology

In order to come to thorough interview questions first a designer was observed while working on a particular design. Interview questions were developed based on background knowledge of CVE design issues and on the observations of the CVE designer in action. The interview questions were tested in two iterations of pilot studies on CVE designers. Based on the results of these pilots the interview questions were refined and restructured.

The interview followed the following structure:

- Definition of task and role in design process
- Design choices made for CVEs
- Approaching an Assignment
- Making a Collaborative Virtual World
- Sources of Inspiration

The interview has been conducted on five individual CVE designers in total, who all have a background in computer science, are all involved in building CVE demonstrators, using raw C-code, the AC3D-editor, and MASSIVE-2. Their experience ranges from work on 1 project for half a year, to work on 8+ projects for the past 6 years, building pioneering mixed reality VEs to pioneering stand-alone Internet based CVE teleconferencing software.

All but the pilot interviews were conducted by two interviewers, were one, a sociology student trained in conducting interviews, took an active role in asking the questions and the other, a psychologist with expertise in CVE usability problems, guided the follow-up questions in the direction of interesting topics mentioned by the interviewees. 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.

9.2.3 Results

The results from the interviews gave an extensive amount of qualitative information about CVE design practices. A summary and interpretation of the results is provided below. Generalisation of the results can only carefully be made, whilst bearing in mind the relatively small number (5) of interviewees.

9.2.3.1 Is there a CVE designer in the house?

During the course of the interviews the respondents were reluctant to define themselves as CVE designers. This analytical category had been created by us to describe someone who is involved in creating CVEs. The respondents, however were not keen on what seemed to be a fair descriptive term.

The view varied according to the interests of the respondent. It generally seemed to be anything that they were not involved in. R2 considered a CVE designer in these terms "there's a CVE designer in terms of the system and there's the designer in terms of the person who puts something together". R4, R5 and R6 were of a similar opinion: a CVE designer is someone who creates the content of a world. In contrast R3 did think that CVE designer was a reasonable category to be put in, but s/he also said that s/he felt s/he had not done any actual CVE design yet (although in fact s/he had been working in the general CVE development area for almost a year).

This contrasts with the perspective of HCI which sees interface design as a multidisciplinary activity. In their own eyes, the Respondents contribution to the development of CVE is not seen as design; which they see as a separate process. They distinguish design from their own work. This leads them to identify more with their own research area than the overall process which their work is ultimately contributing towards. This can mean that they do not fully appreciate the extent to which their work contributes to and shapes the design of CVEs. By identifying other processes as CVE design, the responsibility for its development can be easily shifted.

9.2.3.2 Social discourses; technical processes

CVE designers use a number of terms that are recognizable to a usability researcher. Terms like interaction, collaboration, sociability and functionality are words familiar in their work. However the way these words are used are very different from computer science, the background of the interviewees.

For instance, one of the important concept to CVEs is that of interaction. When a sociologist discusses interaction, they look at the normative and cultural relationships that people have to create links between themselves and society. For usability researchers interaction has a more technical meaning, it refers to the way that humans react to computers; how people use interfaces in achieving their job, this knowledge allows designers to improve the way that interfaces are designed to help people get their jobs done more effectively.

In CVEs the term interaction seems to refer to the mechanical process of communication. The focus on interaction is technical, the emphasis is on making it work, or happen (Bullock and Tromp 1997). Interaction is dissolved into a number of computer operations. The job of interaction is mainly a mechanical operation. The vehicles of interaction are embodiments and objects. This involves the implementation of hardware and software. It would be overly simplistic to argue that designers reduce interaction to hardware and software operations issues only. On the other hand, it obscures the role of the users in these processes.

On the other hand, this kind of reductionism - where words are turned into mechanical references - does create a potentially increasing 'language-gap' when CVE designers come to work in an interdisciplinary team with members from usability research, the social sciences, and the art community. The ideas, values, and use of language differ widely between the different disciplines. This shows up most clearly in the use and meaning of words used during the course of everyday 'business'. These words are misleading understood in the broader framework of common English. Words expressing vague concepts and ideas refer to much more definite ideas and items. Collaborative virtual environments are designed to support a number of users participating in an activity. It is a social environment in the sense that its aim is to support group activity. Designing for CVEs is thus a multi disciplinary process.

9.2.3.3 Which User?

The end-user is not typically considered when designing. R3: “you have ideas and you want to test them out and so you write an application and then it is deciding which is the best way of transforming your ideas into something more tangible in terms of the application.”. R4: “the approach has always been how can we improve it technically and not how can we improve it for users. The assumption is that technical improvements cause user improvements.”. This issue is mostly caused by the fact that these designers are developing prototypes, so that it is not easy to enable representative end-users to use the application, due to the high risk of serious bugs.

9.2.3.4 What Books?

Areas of sociology and cognitive psychology have been picked up and used by CVE designers. CVE designers are using ideas from the social sciences. However, they do not always feel that this information is very accessible. R6: “stuff like aesthetics and world building, if there is a literature it is outside the scope of things that I monitor. The sort of books that come out about VR I find to be extremely unsatisfactory containing 450% of gratuitous waffle with 0% insight. Presumably there are resources that exist within the areas of art, aesthetics, architecture and so on that one might attempt to translate into this arena. I have never yet seen it done well.”. this issue is mostly caused by the fact that VR technology is still in its early stages, so that books or guideline documents that specifically address CVE usability design are not available yet, because they haven't been published yet.

9.2.3.5 Collaborative Aspects of CVE Design

The interviewees seemed to have problems defining the design issues to support collaborative behavior for CVEs. For some it referred to communication issues between participants, for some it referred to the ability for multiple users to interact on the same object without creating confusion. None of the interviewees had a clear definition of what makes a CVE collaborative. For example, R4: “kind of manipulative collaborative malleable I suppose”.

9.2.3.6 Iterative Design

All interviewees basically have a universally iterative prototyping working style. R6: “I always like to have something that works no matter how trivial and incrementally extended. I would probably create a rough outer shape for it which would probably be a cuboid and assuming that it is going to have enough functionality in it to require its own programming behaviors, then code the skeleton class for it, instantiate it in the world, check that it is subject to the behaviours that I am encoding, then it such a vague problem it is difficult to say more then. I add some more features one at a time until I get the desired functionality.”

They use a kind of informal interaction analysis, by trying things out themselves or with a few colleagues. Through this process they aim to understand the social dynamics of CVE interaction. This has a few consequences for the effectiveness of the iterative design process. Currently the design is being modeled on other professionals of the same discipline, as this is where they are being tested. This can result in closed professional culture; drawing together professionals of the same discipline. This means that it is very difficult for outsiders to gain a detailed understanding. This issue is of particular importance to the design of CVEs, as they are ultimately interfaces to social environments. The way CVEs are made today, influences our expectations and thoughts about the CVE technology of the future.

9.2.4 Discussion

The interviewees seem to interpret ‘design’ in two different ways; but often the same word is used to talk about two significantly different tasks:

- A) design in terms of computer code (assuming a designer with a computer science background)
- B) design in terms of the form of objects (assuming a designer with an artistic background)

Many CVE designers are performing both design tasks simultaneously, and the interviewees make clear that a lot of the frustration in their work originates from the fact that they feel they are not sufficiently qualified to do either kind of design. Both design tasks need to take into account how to make things usable for (multiple, collaborative) users. Each design task (A + B) belong to a different discipline, with different associated skills.

The interviewees also seem to interpret ‘guidelines’ in two different ways, and again the same word is used to talk about two significantly different kinds of information:

- A) guidelines on how to work with the software to build objects/worlds
- B) guidelines on how to design usable objects/worlds

A strong need is expressed for guidelines A (however, this may be situation dependent, and only apply to these particular interviewees), and a strong preference is expressed for a certain kind of guidelines B. To be specific they believe they would not use guidelines for design if this stifled their creativity. They would rather see design

tools created which allow designers (specifically type B) to design, with build in guidelines (type A) which allow designers (both types) to build useable objects/worlds.

9.2.5 Common Themes

Below we outline themes which were found to be common to the design practice for CVEs.

- People working in CVE design do not really see themselves as designers because the whole process is broad and they have a specific role and specialism to bring to the process unfortunately this means that CVE design is not a self conscious process.
- The problem with defining collaborative behaviour in CVEs. Does it refer to communication or the ability to act on objects with the environment along with other participants.
- Hyperbole: the designers imagination, mission and purpose is not actually reflected in the results of their work. this leads to frustration but also an ideological rather than tangible perception of the value of the work.
- There is confusion between the natural world and natural law and the familiar world created by social conventions and expectations.
- There seems to be a lack of clear realistic design direction amongst designers. Designing starts with the trivial and microscopic building into the wider environment. They take a guerrilla raid attack approach to programming.
- Individualistic work cultures are common working practices. Although designers work by themselves relying largely in their on ideas seeking help only in practical cases. the ways they work and their are approaches are strikingly similar despite the different focus each designer has due to their own specialisms.
- Virtual worlds are paralleled with real worlds, they are defined in this way so that they can make sense with the minimum of jarring. This brings with it a continuous trade-off decision making process between the amount of visual detail used and the effect this has on run-time performance.
- Dissatisfaction with the visual impact. The interviewees feel that they lack the artistic capabilities to design the contents of the world.
- Testing is informal, often conducted by the designer themselves, or with a few colleagues. Typically no representative end-users are used.

9.2.6 Conclusions

Any set of design guidelines need to take into account that CVE designers have to consider various issues on different levels and aspects. There are also discrepancies in the different things that designers bring to their work according to their own interests and expertise. The guidelines could be split into the different issues and provide a basic breakdown of the things that CVE designers need to consider.

The interviewed designers have no clear strategies for designing CVEs. They have no clear understanding on the usability topics to address in order to design for the support of collaboration in CVEs. None of the interviewees work according to the standard design method of requirements specification, conceptual specification, building prototype, testing, and rebuilding. Their primary concern seems to be to demonstrate functionality, and a second concern seems to be to generate publishable papers from these demonstrations. It has to be emphasized that this is not a shortcoming of the CVE designers. These work practices are shaped by the fact that these designers are working in an academic environment, where CVE development takes place at the level of basic development of new technologies. However, their work practices do show the need for the development of a more systematic guidance for usability design.

A questionnaire addressed the reception of the Usability Evaluation deliverable by the designers community in COVEN (Del 3.3). This deliverable identified many usability problems in great detail. Twelve designers answered the questionnaire. Six designers confessed not having read the document yet, because it had no direct bearing on their task. Three designers, who were directly involved in creating the design under evaluation, had read 80% of the document or more. They found the document reasonably useful, but would have liked more specific advice on the solutions to the found usability design problems. They expressed the need for a step-by-step process description of the relevant design principles. Indeed, the fixes made to the interface by the designers, after reading the information in Del 3.3, removed the identified problems while at the same time introducing similar new problems (Del 3.5c). It seems therefor that there is a need to educate the designers themselves about the correct design issues.

Improvements to the lack of understanding and knowledge about usability design for CVEs should involve a number of issues. A better understanding of what constitutes collaboration in CVEs is needed. A better understanding of the level of detail, and degree of realism necessary to support users in their task is needed. A better way of presenting the usability design principles is needed. A general direction for such design support could be in the shape of light weight automated hints in world construction tools.

9.3 Future Work on the Design and Inspection Method

The main problem with the design of CVEs seems to be in the design of the flow of interactions. Although the Inspection method does address the design of the flow of interaction for *separate* actions, and objects in its current state it does not emphasize the flow of action *between* actions. A short description of the problem is provided below.

9.3.1 Design Flow of Interaction

For most Virtual Environments (VEs) the choice is made to represent the environment in a realistic way in order to allow users to transfer their daily knowledge of operating in the real environment to the virtual one. However, this real-time image generation puts a high load on machine processing time, and in the case of networked VEs (CVEs) on the network traffic generated by the real-time image update. To reduce this load the functionality and appearance of the environment and objects is reduced to the bare minimum. Because of this simplification it is often difficult for users to predict which operations are available and which are not. Users have been shown to struggle with the interface for these reasons (Kaur, 1997; Steed & Tromp, 1998).

VEs are intended to provide three dimensional graphical representations of spaces for people to interact with. Users are enabled to freely roam through the spaces, interact with objects and other users present in the same space. Because of this freedom of navigation and interaction it is difficult to predict what actions users will take, and in what order they will perform their 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 navigating into precise positions (Kaur, 1997; Tromp & Snowden, 1997).

The correct performance of actions and sequences of actions in the VE greatly relies on the design of the interface of the VE and its objects. User actions in VEs oscillate between user created 'story-lines' (i.e. the successful performance of action sequences) and user interface struggles (i.e. the inability to perceive or perform the correct action, the absence of feedback, or both).

Standard HCI design methods, developed for 2D interfaces, are using the notions of 'direct manipulation', 'perceptual affordances' and 'sequential affordances' to address issues of representation and user action guidance (Preece et al., 1994). Direct manipulation is concerned with allowing the user to directly act out their goals by providing them with icons which represent objects traditionally used in the real world to perform those actions (for example a paint brush icon to represent to action of painting). Affordances are "the properties of an object which determine how an object can be used" (Norman, 1992). Perceptual affordances are visible controls which suggest functionality. Sequential affordances refer to the notion that acting on one perceptual affordance can lead to the perception of new information, leading to information indicating new affordances. The issue is how we can use these notions to inform the design of 3D objects and spaces for VEs.

9.3.2 The Need for Structure

In VE's we want to create easily understandable environments and guide users in their expectations. We want to supply users with intuitively understood affordances of the environment (Gaver, 1991). During the COVEN Project a virtual environment has been build on top of dVS from Division Ltd. This platform provides 3D information in the general area of travel. An Inspection (Cognitive Walkthrough) of the COVEN platform (Steed et al, 1997) was made to establish whether the design supported users, and if not how to improve this. Numerous remarks made by the Inspectors illustrate the need for more structure in the design of the interactions. Below a few examples are given:

"The overall design of the rooms is not consistent in terms of realism and details. The 'semi-real' metaphor is not fully consistent: there is a real world feeling but still some objects are floating in the air. This can sometimes be confusing. Also, some task-unrelated objects are represented (typically, radiators) - why these, why not others (lamp, plants, etc.)?"

Figure 1: View of the Tourist Information room for the island of Rhodes, Greece. On the left wall is the slide-show, in front a table, with the CDs and the CD player.

Another example is the CD player which was implemented to provide a metaphor for a slide-show accompanied with a voice-over of tourist information. These are some of the comments on the design:

"CD player is not obviously visible unless you are close to it", "I find it very hard to read the CDs. I hardly can speak of a 'selection process'; as user I just try", "not clear what the choices are unless you get right in front + look down (too difficult to navigate such a move), only then can one read names of the CD's", "It is not obvious to me that playing a CD would involve a slides show. When I activate a CD, I am expecting sound, not images. This is because the CD player looks like the audio CD player I have at home, not like a multimedia CD-ROM appliance", "The option for playing a CD-ROM is not obvious. You come upon it when playing with the mouse. It actually comes as a surprise that a simple CD selection activates the player (not consistent with the usual behaviour of selections within the application). Risk for errors", "Anyway, the show is fine. But once started, how do I stop this lady telling me about Rhodos? I cannot find a stop button."

An example of the need for guidance for the sequence of tasks is the lay-out of the main tourist information office room and the task-related or functional objects in it. The user first has to go through the door.

"Opening the door of the Meeting Room wasn't as easy as I thought. The door was slammed in my face for three times. And what a sound was accompanying it: for a moment I thought I was in Mickael Jacksons thriller! I think a user prefers that the doors opens after touched the handle and will place him in the middle of the next room."

Next, the user is expected to find the CD slideshow, but the table with the CDs and CD-player is not placed in a very prominent position. After having negotiated the CD slideshow the user is expected to take a virtual flight over the holiday destination by means of entering a teleporter, which is located in a corner of the room. However, again the object is not placed in a prominent position, and there is not a lot to guide the user's interest towards this other main functional object in the room.

"It is not clear what the function of the teleporter is, unless you know what it is. The outside does not suggest anything about it's functionality" and "Maybe a textual tag would help, e.g. "to virtual Rhodos". Or is it part of the fun to entertain mystery? Teleporter needs a label."

Finally, one ends up with problems on the level of structure in the object interaction again:

"The controls for the teleporter are not obvious since they are unlabelled. Indeed one has to open the teleporter first before these controls become apparent."

9.3.3 Sequential Affordances for 3D Direct Manipulation

Designing the affordances of objects in the VE is a trade-off between realism and simplification - between user needs and utilized computing resources. A balance needs to be found between the essential and non-essential elements of the object, so that the user can still perceive the correct actions and functions, while the machine load is kept to the minimum. Often this simplification results in a more or less cartoon-like representation of the VE and the objects. The objects represented should be caricatures, which ideally act as a form of amplification through simplification (McCloud, 1993). If we can reduce the representation of the object to its most salient features and functions, we can guide the user more easily to the next correct action.

In addition to exploring the caricatures of representations, we also need to explore the caricatures of situations. Live-action films are stripped-down versions of reality, to increase the intensity of the story, thereby guiding the viewers in their anticipation of the next action (Straczynski, 1996). If we can simplify the situation in which the user finds herself at a particular moment in the VE interactions, we can predict the next user action more easily, thereby clearing the path of the user story in the direction we would like the user to go. Effectively, each VE user creates her own story-line in a VE - of all possible actions, the user will have to select one, which could lead to the next, etc. The designer will have to help the user identify the actions and objects necessary to perform their tasks, especially the order in which they are to be used. Some chunks of information have to be interpolated by the user, some have to be attached to the objects, before the user can make sense of the environment. The sequence of appearance is extremely important. The items of information function as elements of a story, and while the arrangement may be flexible and open, the elements have to be assembled in a particular order to make sense of the story. And that order should be designed to guide the users through their tasks. Here we can learn from the art of exhibition design. Exhibitions are ideally regarded as a form of sculpture: "They are three-dimensional compositions which recognize the importance of solids and voids and strive for satisfactory spatial relationships." (Lawson, 1981).

9.3.4 Conclusions

The perceptual affordances of 3D objects in VEs need to be improved by choosing simplification of the objects so that the available functions on the objects are amplified as much as possible. The sequential affordances of a task which involves interaction with multiple objects in a certain, specific order needs to be designed with more care by directing the users attention from one object to the next as desired. Generally, partial tasks can be automated, and guidance for sequential affordances can be provided by structuring the lay-out of the rooms and position of the objects more deliberately. Instead of arbitrarily positioning objects and rooms, they can be grouped and ordered into meaningful parts which intentionally draw the user from action to action. Standard HCI alerting techniques for guiding user attention to the next action, such as the use of colour, flashing, and reverse video are

not very elegant solutions in a VE, especially when it concerns the design of multi-user VEs. However, the use of spatial and temporal cues and audio warnings may be much more effective. Designing the spatial lay-out of rooms and objects more carefully, and providing more carefully designed object affordances could improve the usability of VEs. Simplifying the VE by deliberately designing caricatures of objects and situations may be a more effective way of keeping machine load down, without losing usability points.

10 Final Conclusions

The Inspection method as it is presented here, has now been tested once. It needs more testing, and refinement of the method. The reception of the report by the designers - the primary recipients of the Inspection report - will be accessed in order to gather feedback on the effectiveness of the reporting style of the results. A number of necessary refinements can already be specified:

- Flow of actions part of Inspection
- Steps of cycles synchronised as much as possible.
- Formulation of steps of cycles more precise.
- Allocation of cycles to functions, tasks and objects made more easy.

In addition, attention should be given to the fact that the interaction cycles provide excellent guidance for the design of actions and objects, *while they are being designed*. It is for this reason that designers are encouraged to use the interaction cycles as a first form of guidance, when they are re-designing the problems with the application as reported in Section 8. One of the designers in the COVEN project (KPN) has already started using the cycles as design guidance. A report of these activities is still pending. The authors of this document would appreciate anyone using this method to provide them with feedback on the effectiveness and any problems occurring.

11 References

- Durlach, N.I., Mavor, A.S., (1995). Virtual Reality: Scientific and Technological Challenges, National Research Council, National Academy Press, USA.
- Gaver, W.W., (1991). Technology Affordances, in: (eds.) Robertson, S.P., Olson, G.M. & Olson, J.S., Proceedings of CHI'91: Reaching through Technology, ACM, N.Y., pp. 79-84.
- Kaur, K., (1997). Designing Virtual Environments for Usability, Doctoral Consortium paper in: Proceedings of INTERACT'97, Sydney, July 1997, IFIP.
- Kaur, K., Maiden, N., Sutcliffe, A., (1997). Interacting with Virtual Environments: an evaluation of a model of interaction, in: Interacting with Computers, Special Issue on VR.
- Lawson, F. (1981). Conference, convention and exhibition facilities : a handbook of planning, design and management..
- McCloud, S., 1993. Understanding Comics, the invisible art, Kitchen Sink Press, NY.
- Nielsen, J, Mack, R.L., (1994). Usability inspection methods, John Wiley and Sons, New York, NY.
- Norman, D., (1992). The Psychology of Everyday Things, Basic Books, New York.
- Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., Carey, T., (1994). Human-Computer Interaction, Addison-Wesley, New York.
- RESPECT, (1997). User Requirements Framework Handbook, Deliverable 5.1.
- Steed, A., Tromp, J., Normand, V., Dijkhuis, J., (1997). Combined Usability Inspection Report, Anthony Steed (ed.), Public Deliverable COVEN ACTS Project N. AC040.
- Steed, A., Tromp, J.G., (1998). Experiences with the Evaluation of CVE Applications, Proc. of Collaborative Virtual Environments 98 (CVE'98), University of Manchester, 17-19 June, 1998, D. Snowden and E. Churchill editors, pp. 123-130.
- Straczynski, J.M., (1996). The Complete Book of Scriptwriting, Titan Books, London. .
- Sutcliffe, A., Kaur, K., (1998). Evaluating the Usability of Virtual reality User Interfaces, submitted for publication.
- Tromp, (1997a). Collaborative Actions in CVEs: a task breakdown for evaluation in Activity 3.4.
- Tromp, (1997b). Results Usability Questionnaire, Internal COVEN Document.
- Tromp, J. (1998). Assessing Criteria of Credibility for Collaborative Virtual Environments, 1st year Viva report, internal report, Dep. of manufacturing Engineering & Dep. of Computer Science, University of Nottingham.
- Tromp, J.G., Snowden, D., (1997). Virtual Body Language: Providing Appropriate User Interface in Collaborative Virtual Environments, in Proceedings of ACM Symposium on Virtual Reality Software and Technology, Lausanne, Switzerland.
- Tromp, J. & Steed, A., (1998). Cognitive Walkthrough and Heuristic Evaluation: 3D and Collaboration Additions for CVEs, Internal COVEN document.