Chapter 5 Research Approach

> "The man of science characteristically processes his experiences of the phenomena encountered in his specific sector in one or more of the following ways: he endeavours to describe, to order, to record (measure) them, to understand and to explain them; in these activities he is motivated particularly by a desire to be able to predict new phenomena, so that their predictability shall enable him to control his sector by influencing the phenomena. [...] Uncompromisingly he seeks after truth, that is, factual truth concerning his sector of reality – in the same way as a philosopher seeks for wisdom."
>
> Prof. Adriaan de Groot, (1969).

5 Research Approach

This chapter consists of: four distinct sections: a description of the research approach that is used in this thesis, a description of the methods used for the observational analysis presented in this thesis, a description of the experiments, and a description of the type of data collected.

5.1 Introduction

This chapter describes the frameworks used for the CVE evaluation presented in this thesis, the methodology used, and the experimental set-up. It places the activities performed for this thesis into a general framework derived from four tentative hypothesis about usability for CVEs, and in a specific framework derived from the methodological approach chosen for this research. The arguments for the choice of this particular methodological approach are formulated in Chapter 2, section 2.4, and the conclusions are used to explain the frameworks presented in this chapter. The frameworks are presented in order to clearly delineate the systematic scientific approach used for the research presented in this thesis. Furthermore, it describes and discusses the measurement tool for the observations, the choice of experiments, and the experimental set-up. The aim of this chapter is to ground the results from the critical analyses of the previous chapters in a clear manner, so that the research findings of the different strands of exploration can be interpreted in relation to each other, thus gaining maximum insight in usability for CVEs from the research.

The next section presents a general framework for the analysis (section 5.2); the observation method (section 5.3), followed by a description of the experiments and the experimental set-up (section 5.4), and some early conclusions about the data gathering process (section 5.5).

5.2 Framework for the Analyses

The critical analysis of usability as applied to CVE (Chapter 2, section 2.4) lead to the formulation of the goal of the CVE user and the goal of the CVE application. In summary:

- The goal of the CVE user is to be provided with information about who does what, where and to whom, sufficient to understand and contribute to the goings on, compared to real life collaboration.

- The goal of a CVE is to afford the perception of the functionality, the space, the objects, the users' own virtual embodiment, the other participants, and their activities, sufficiently for a typical user to achieve effective interaction and communication with and within the CVE for collaboration between participants.

As evoked in Chapter 2 (section 2.4.3), in order to assess whether these two goals are satisfied, it is essential to observe CVE users in action, since there is insufficient knowledge of collaboration in CVEs to define validation criteria more precisely. In order to capture the design, evaluation and usability problems of CVE as a whole, the research in this thesis starts from a general framework of four tentative hypotheses about collaboration design and usability. These hypotheses were derived from work by Kaur (1998), who used a similar broad approach to assess the design and usability of single user human-computer interaction in VEs. The author adapted these hypotheses to address collaboration for CVEs:

Hypothesis 1: Interface design guidelines are needed specifically for collaboration in CVEs.

Hypothesis 2: General patterns of collaboration can be predicted.

Hypothesis 3: Collaboration design properties can be predicted.

Hypothesis 4: New design properties support CVE collaboration.

Table 5.1 shows how the author expected to test these tentative hypotheses, the anticipated results, and the place of the results in this thesis. The shaded cells describe a test that has not been performed due to lack of availability (described in more detail below). Hypothesis 1 has been tested by looking at two complementary sets of information. Firstly, the results from the Usability Inspection have been analysed. The Inspection method, including the development and refinements, are presented in Chapter 9, as it does not constitute the main thread of research presented here. However, the findings of the Inspection have been analysed and a summary of the usability problems found is presented as part of the answer to H1. Secondly, interviews with CVE designers were conducted, and the analysis, interpretation and summary are presented in Chapter 6, as the other part of the answer to H1.

Hypothesis 2 has been tested by developing two sets of information. On the one hand a model of collaboration was developed, which in turn has been used to develop a task cycle for collaboration as an addition to the Inspection method, to enquire into collaboration activities as well as the other tasks that were distinguished as CVE tasks for inspection. The collaborative task cycle is presented in Chapter 9, section 9.3.2. This task cycle was found useful and it was further refined during the development of the CVE Inspection method. However, it was not used for the main work presented in this thesis, and does not come back in the analysis directly. On the other hand, the observation method was developed and refined, and used to make predictions of user behaviours, which was subsequently statistically analysed to see how much these predictions differ from the observed behaviours. The results of that analysis are presented in Chapter 8.

To test hypothesis 3 a hierarchical task analysis of collaboration was created, and used to predict design properties which are essential to avoid usability problems. This information is presented in Chapter 7 and 10. Additionally, a summary of typical collaborative behaviours during real world collaboration derived from the literature review of collaborative behaviours (Chapter 3) is used to analyse the observational data. This information is presented in Chapter 10, section 10.2.1.

In order to test hypothesis 4, two experiments were planned. In the first experiment, being the control condition, user behaviour during representative collaborative tasks was to be observed, in order yield insight into what the user problems are, and which design improvements could improve the usability. For the second experiment, the experimental condition, a new version of the CVE with improvements for multi-user collaboration was used. Again user behaviour during the same representative collaborative tasks had to be observed, which was expected to yield insight into whether the design improvements actually improved the usability. Unfortunately, implementation of new design properties to improve the usability of the COVEN CVE did not take place at a level that would improve multi-user collaboration. This meant that the planned test to ascertain the impact of the proposed new design properties could not be performed.

Hypothesis	**Test**	**Activities**	**Expected Results**	**Place in Thesis**
H1: Need for CVE design guidelines.	Exploration of CVE usability problems.	Development of Inspection method for CVEs; Model of collaboration in CVEs.	Analysis of usability problems and design issues.	6.4 9.3.2 10.2.3
	Exploration of CVE design problems.	Interview CVE designers	Analysis of type of design guidance needed.	6.2
		Feedback questionnaire on COVEN usability report.		6.3
H2: Prediction of general patterns of collaboration.	Predictions of observed behaviors based on expected behaviors.	Literature review of collaborative behaviours.	Indication of the type of collaborative activities that should occur, when, and why.	3

		Observational method to analyze group collaboration in CVEs created.	Measurement tool for the type of collaborative activities that occur, and their properties.	5.3
H3: Prediction of desired design properties	Predictions of usability problems during CVE collaboration activities made from Hierarchical Task Analysis	Hierarchical Task Analysis for CVE collaboration created.	Interpretation of the type of collaborative activities that must be supported.	7.3
			Predictions of design properties needed for collaboration support in CVEs.	7.4
H4: New design properties support collaboration.	Experiment I: Control condition using CVE prior to improvements.	Observation of user behavior during representative collaborative tasks, using the group collaboration analysis method.	Indication of usability problems, and the type of collaborative activities which do occur, when, and how often.	5.4.4 8 10
	Design recommendations	Detailed suggestions on how to improve the CVE by increasing support for collaboration.	Indication of the type of usability improvements needed, and the effort involved.	10
	Experiment II: Experimental condition using improved CVE.	Improvements in collaboration are defined as fewer usability problems, better task performance or lower task completion times, and general improved satisfaction.	Comparison of results for conditions I & II: Empirical evidence about which particular design properties do improve CVE collaboration and how.	

Table 5.1: Hypotheses, tests and anticipated results, place in thesis where the results are presented.

Although during the COVEN project, each iteration of the design of the CVE was an improvement on the previous version, it was not possible within the scope of the project to introduce major changes in the CVE, which were the type of improvements hypothesized to increase the success of multi-user collaboration. The focus of the analysis was therefore adapted accordingly, using the following research questions:

- What behaviours are exhibited, how often, in what order?
- Is observed behaviour similar to predicted behaviours?
- Do novice users perform differently from experts?

In order to perform the observations of CVE users whilst they are collaborating on a representative task, an interaction process analysis method (Bales, 1951; Forsyth, 1990) was developed by the author of this thesis, which for measurement and quantification of human interaction in CVEs. Additionally, two new conditions for comparison were created (condition I: Experts; condition II: Novices). The main part of results presented in this thesis, are based on the observation method, and the other activities are where needed to support and better understand usability for CVEs.

The research presented in this thesis can be depicted as a diagram, to illustrate how the different activities feed of each other. The author proposes this structure as a specific framework for the exploration of collaborative behaviours in CVEs (see Figure 5.1). Figure 5.1 shows the various elements of the research presented in this thesis (in rectangles), their purposes (in ovals), showing the interconnections between them, and the relationship of the various elements to the final results. The literature review of collaborative behaviours was made, from which a hierarchical task analysis of collaboration in CVEs was created. Observations of CVE users collaborating on a representative task were made, from which an index and sequence analysis of collaborative behaviours was created. From the hierarchical task analysis predictions about usability problems were made, and compared to the usability problems, which were found through the observation of CVE users. This comparison was used to create an overview of the type of usability problems that exist for collaboration support in CVEs. And finally, this overview is used to create usability design guidelines for CVEs.

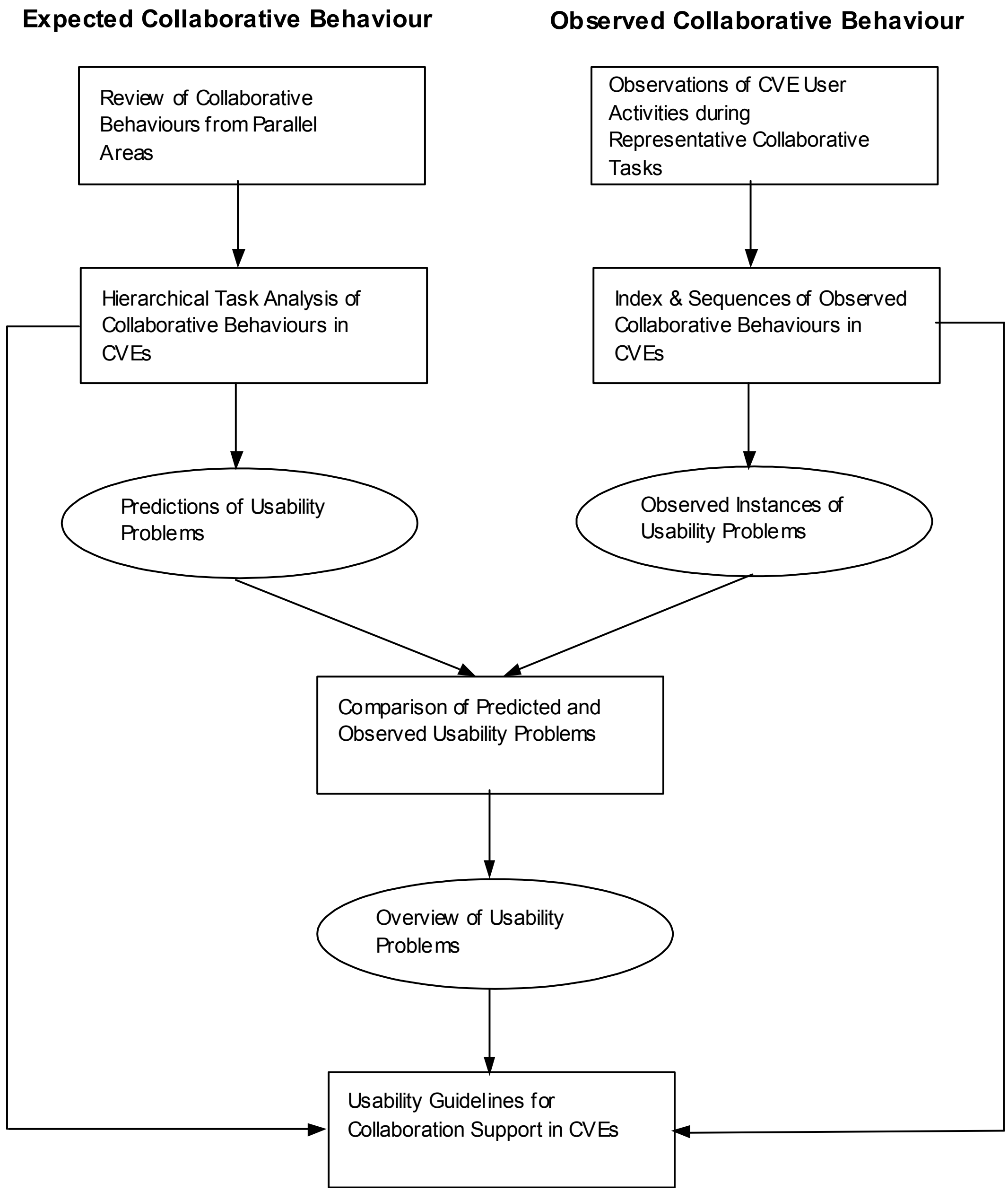

Figure 5.1: Research presented in this thesis in a framework.

Additionally, the longitudinal approach of the COVEN usability evaluations (described in Chapter 4) allowed for the collection qualitative data through a repeatedly administered questionnaire at the end of each network trial (see Appendix D). The qualitative data derived from this questionnaire are used to illustrate the usability problems that were found during the analyses.

5.3 The Observation Method

The observation method as created by the author and presented here, allows for the quantitative analysis of the interactions of CVE participants. The method is based on a general-purpose framework for observation called Interaction Process Analysis, a method described by Bales (1951), which can be used to obtain a series of standard indices regarding the structure and dynamics of interaction in any small group. The heart of the method is a way of classifying direct face-to-face interaction as it takes place, act by act, and a series of ways of summarizing, analysing the resulting data so that they yield useful information (Bales, ibid). Basically, observers score the occurrences of small group behaviors as they occur over time, based on a system of pre-defined categories. The number of times the categories are scored and the order in which this occurs gives us information about the behavioral patterns of the members of the small group under observation. The categories that are used to score the behaviours have to be created, developed and tested by the researcher employing the method, based on the particular research question under focus. For instance, Bales' categories (ibid), were developed to analyse small group behaviours in meetings, discussion groups, and therapy groups. Although Bales' method may be appropriate, the categories Bales used for analysis are not directly suited for scoring collaborative behaviours in CVEs. To analyse the interaction process of multiple users collaborating in a CVE, the author of this thesis has adapted the method by creating new categories. These categories were necessary in order to score the observations of collaborative behaviours of CVE users as they could be observed on the video-recordings of the longitudinal COVEN network trials during the CVE interaction experiments the author designed. The categories and how they were developed are presented in section 5.3.2.

Some other VR researchers have used this method, with their own set of categories. For instance, Neale et al. (1999), used a multiple activity chart scoring method to observe interaction with a virtual environment. She applied the method to several Virtual Learning environments for children with learning disabilities in six case studies. The scoring categories were derived from a theory of learning. She coded language and actions of two simultaneous users of the software; a teacher and the child with learning disabilities. Not all seven principles of the theory of learning are visible behaviours, so only observations of behaviours gleaned from the interactions between the teacher and the child and the VE could be scored and Neale had to drop two of the seven principles. The remaining five categories were extended based on pilot studies, including subcategories within the mean categories, and coded for observations of negative occurrences of the principles.

With regards to the applicability of the method Neale reports that she does not know of previous investigations using this method applied to VEs. With regards to using the method she reports that some behaviours were difficult and very subjective to score, some categories were not specific enough, and some categories were extremely difficult to infer. On the whole the method assisted in uncovering a number of theoretical relationships between certain aspects of the design of the VEs and their subsequent use.

Kanuritch et.al. (1997) including members of the same research group as the work of Neale et al. (ibid), used a multiple activity scoring chart to score teleoperator behaviours with a remote handling system for nuclear power station maintenance and repair activities, as part of a usability study of two different handling systems. Performance data such as time to complete tasks, errors made, subjects' movement of joystick, display viewed, speech taking place and activity of robot manipulator were charted. No record exists of analysis of the multiple activities charted, nor are any remarks on the usefulness of the method available.

There are other ways of analyzing sequential interactions. One such method is the network traffic analysis (Greenhalgh et al., 1997), also performed during the COVEN Project (Deliverables 3.2, 3.4, and 3.7A). Time-stamped logs of key events of network traffic generated by user activities are analyzed to explore the type of network traffic generated, and to predict what volumes and patterns of network traffic might be generally expected. The key events that can be recorded are the ones that the computer system can detect, such as position, movement, speech, manipulations of objects or interface controls, etc. Generalizations and predictions about user behavior can only be reliably made based on large amounts of quantitative data. Because the recording, and analysis of key events are largely performed by the computer, network traffic analysis is a cost effective statistical method to analyze and predict user and network activity. However the key events which the computer records, have no immediate bearing on the context in which the user activities take place. Explanations as to the meaningful relations between sequences of events are not automatically available.

Ethnographic observation is another method to analyze sequential interactions, which overcomes the limitations of network traffic analysis in the interpretation of the recorded interactions. However, ethnography is typically a time-consuming activity. Ethnography is a standard practice in anthropology, which generates quantitative data, based on the observers' extensive exposure to a certain community, workplace, or work practice by means of intensive observations, interviews, and participation. From this exposure the ethnographer gradually extracts and makes sense of the key events under exploration. Generally, ethnographic observations lead to rich, detailed understanding of the meaningful relations between otherwise seemingly unrelated acts. The great advantage of this method is that it gives insight into real acts in the real activity context in which they take place, thus greatly increasing the quality of the data. The disadvantage of this method is that it leads to anecdotal evidence only and has little quantitative data to support its claims.

The observation method presented here aims to combine the best aspects of both methods described above. It allows for the collection of large amounts of quantitative data, with a direct bearing on the context in which the user activities take place. Additionally, by collecting quantifiable data in the context in which they take place this method avoids the limitations of other methods in that the key events under analysis are precisely the meaningful relationships between activities. This method provides a tool to accomplished the following main goals:

- Generate an insight into the interaction dynamics of CVE users through space and over time.

- Check user interaction in a CVE in relation to other users, objects, and spaces.
- Establish how well task-design and task-flow meet representative user needs.
- Recommend redesign solutions for specific areas of potentially serious user interface struggles.

5.3.1 The Observers' Role

The observer has a list of categories into which she classifies every item of behaviour she can observe and interpret. The classification she makes, involves the imputation of meaning, the "reading in" of content, the inference that the behaviour has function(s), either by intent or effect. The observer has to be familiar with the categories. She should know the central theoretical meanings of the categories and also the range of variations of concrete behaviour included within each of the categories. As the people in the observed group talk to each other, the observer breaks down their behaviour into the smallest meaningful units she can distinguish, and records the scores by putting down beside the proper category, the number of the person speaking, and the number of the person spoken to. The observer follows the interaction continuously in this microscopic manner, attempting to keep the scores in the sequence in which they occur, and to ensure that no item of behaviour is omitted. All kinds of behaviour can be included, provided that the observer can assign meaning to the behaviour in terms of the categories.

5.3.2 The Interaction Categories

The set of categories used for scoring interaction must be concerned with aspects of interaction so general that they will appear in communication between members of any small group, regardless of the idiosyncratic content of the topic of their discussion or the kind of concrete problems or subjects with which they may be dealing. In addition to the formulation of behaviour that always appears, the list must be concerned with certain variations of behaviour which may not be frequent in certain groups but which can and do appear under certain conditions, regardless of idiosyncratic content. All of the categories included should assume essentially the same time span; that is, they should all refer to single acts of communication or expression. The unit to be scored is the smallest discernable segment of verbal or non verbal behaviour to which the observer, can assign a classification based on the categories.

The categories that the author of this thesis has created are based on several incremental stages of observation, modelling, testing and refinement. Eight basic interaction categories have been created: communicate, external, gesture, manipulate, navigate, position, scan, and verify. These categories were developed in the following way. First, a pilot observation of a recording of a representative CVE interaction session was made, noting down all single units of behaviors once. The observation material consisted of video footage of two participants collaborating on a virtual moving task in MASSIVE-2 (Hindmarch et al., 1998). This list of observed behaviors was grouped and condensed into 26 categories. A focus group (Jordan, 1998) of 6 participants (consisting of psychologists, sociologists, and ethnographers) assessed the categories based on their professional expertise. These categories were subsequently tested on another recording of a representative CVE interaction session. The frequency and fit of the categories was analyzed and further reduced to 8 basic categories. These 8 categories were assessed by another focus group of 6 participants (consisting of VR researchers, evaluators and developers) by applying them to another representative CVE interaction recording, followed by a discussion of the technique afterwards.

Category	Definition	Comments
Communicate (C)	An act of communication is defined as any auditory or textual ‘utterance’.	Actors communicate for many reasons, and if the observation is concerned with specific types of communication (for instance verbal and non-verbal) this category needs expanding.
External (E)	An external act is any situation where an actor may be observed to switch attention from the virtual environment to the physical environment.	Attention may be given either to the room or to a window external to the CVE application.
Gesture (G)	A gesture is defined as any non-verbal, non-auditory, non-textual act, including facial expressions.	Most CVEs have a limited set of gestures available, but we include creative use of the virtual embodiment to express emotions in this category (e.g. twirling the virtual embodiment around its axis; sometimes used as an indicator of presence, or joy).
Manipulate (M)	An act of manipulation is defined as the selection, displacement or operation on any object in the virtual environment.	Depending on the object being manipulated the goal of their manipulations may be more or less clearly determined.
Navigate (N)	Navigation is defined as directed movement of the virtual embodiment from one spot to another, with a clear start and finish of this movement.	Navigation can be differentiated from ‘Scanning’ behavior by its movement from A to B, where scanning involves looking from A to B, without obvious relocation of the VB.
Position (P)	Positioning is defined as the precise act of alignment of the virtual embodiment or the object in relation to something	It may sometimes be difficult to distinguish a positioning act from a navigation act, but generally

	else in the virtual space.	positioning takes place after arrival near an object.
Scan (S)	Scanning is defined by a sweeping movement from A to B, not with the intention of moving the whole virtual embodiment, but with the intention of viewing (part of) the virtual space.	Scanning movements are difficult to perform and actors may be observed trying to sweep from A to B several times before they can actually perform a smooth sweep.
Verify (V)	Verifying is defined as all acts which have the goal of letting other users know that things are still as they should be, or checking with other users or programs running in other windows.	Verifications are sometimes expressed as verbal messages and sometimes as physical acts. However, this type of activity occurred so often that it was given its own category.

Tab. 5.5: The list of interaction categories.

The author of this thesis presents the 8 categories as the fundamental acts a user or actor can be observed to perform at some time during interaction with other users in a CVE. Table 5.5 presents the categories, their code (in brackets), their definitions, and comments for the observers benefit.

This list of interaction categories can easily be changed or expanded for more detailed observations or for a CVE with more complicated interaction mechanisms. For instance, when we want to capture all different types of communication that can be observed, and not just each act of communication, we need to extend the category. In order to score all different type of communication that can be observed, new scoring categories have to be made up on the fly. For instance, the category ‘communicate’ can be extended with all observed sub-categories, such as possibly ‘communicate-meta level‘, ‘communicate-verify’ or ‘communicate-socialize’. If, through analysis, it turns out that one or more of these sub-categories are basic interaction acts we can decide to add these sub-category to the list of basic interaction categories during further statistical analysis.

5.3.2.1 Contextual Interpretation

The most salient insight from ethnographical observations is that a collaborators' activities are part of the larger goings on in the shared space. Their actions are initiated and adjusted based on their awareness of the other collaborators and the shared objects. In order to capture the meanings of each observed act in the CVE, the categories were extended according to a basic formula: each category represents an act that the actor is performing to satisfy a certain goal (c.f. Suchman, 1987).

	Communicate	**External**	**Gesture**	**Manipulate**	**Navigate**	**Position**	**Scan**	**Verify**
Communicate		CE	CG	CM	CN	CP	CS	CV
External	EC		EG	EM	EN	EP	ES	EV
Gesture	GC	GE		GM	GN	GP	GS	GV
Manipulate	MC	ME	MG		MN	MP	MS	MV
Navigate	NC	NE	NG	NM		NP	NS	NV
Position	PC	PE	PG	PM	PN		PS	PV
Scan	SC	SE	SG	SM	SN	SP		SV
Verify	VC	VE	VG	VM	VN	VP	VS	

Tab. 5.6: Intentional acts and their respective categories.

The categories are subdivided to express the type of result that the actor seems to be intending to achieve. For instance, an actor may navigate (N) towards a certain position in order to be able to communicate (NC), or an actor may communicate (C) in order to verify (V) that the other actor has perceived his/her action (CV). The total list of categories is simply put in a matrix, which increases the total number of categories to 8 * 8 = 64 categories. The categories can now be structured and displayed in the following summarized way (see table 5.6).

Any additional categories that were made up on the fly during the data collection process were simply added by creating an incremental number for the original category and with a clearly defined additional meaning. For instance: CC is the category for 'communicating' and a new category to capture the different types of communications was added during the analysis: CC1 is the category for 'communicating about task at hand; contributing to progress'. It cannot be expected that all categories will get a similar number of hits, but this is precisely one of the things we would like to know.

The advantage of this approach is that firstly the observed activities can be interpreted on their own by looking at the top categories only (the frequency of single types such as C, E, M, G, etc.). This will reduce the data set to create maximum overview. Secondly, the observed behaviours can be interpreted in conjunction, using the intentional matrix (the frequencies of double types such as CC, CE, CM, CG, etc.). This allows for an analysis of what the users are trying to achieve through their actions. Additionally, the data is analysed to verify what type of acts tend to follow each other (the frequencies of act type CC to follow other types of intentional acts such as CC, CE, CM, etc.). This allows for the analysis of a third level in the sequence of behaviours. Finally, using the scores for each type of precisely specified observed act (the frequencies of type CC1, CC2, EE1, EE2, etc.), a rich source of insight is generated by interpreting what kinds of behaviours are exhibited, when.

Communicate (C)

Actors communicate for many reasons. Communication is an essential element in collaboration and many aspects of collaboration can be found with repetition in the communications observed.

Communicate-External (CE)

Actors may be observed communicating to the other CVE participants about something happening in their physical environment. For instance, an actor may receive a phone call and announce this to the other CVE participants.

Communicate-Gesture(CG)

Actors may be observed to accompany their gestures with speech to emphasize or illustrate the fact that they made the gesture.

Communicate-Manipulate (CM)

Actors communicate in order to manipulate an object. They may be observed discussing an object in their shared space in order to find out who is, or who will manipulate the object (first).

Communicate-Position (CP)

Actors communicate in order to position their VB in relation to the position of something else in the shared space. They may be observed discussing the precise alignment of their VB or that of the other actor(s).

Communicate-Scan (CS)

Actors communicate in order to scan the environment. They may be observed discussing the relative position of objects or other VBs in relation to other objects in the shared space or in relation to their own VB or objects near them.

Communicate-Verify (CV)

Actors communicate in order to verify what they have heard, seen, done or said. Making sure the other actors have perceived the same thing is an essential element of collaboration. Many different kinds of verification behaviours can be observed. It is recommended to make a record of the kinds of verification that takes place.

External (E)

Actors perform 'external' acts during their CVE interactions. Sometimes they can be observed checking another window on their screen with information that may or may not have something to do with the CVE. Any act that is not made inside the CVE is interpreted as external.

External-Communicate (EC)

Actors may be observed to intermittently having to deal with other tasks which are part of their physical surroundings. They may be observed to carry on a conversation with somebody in their physical location.

External-Gesture (EG)

Actors may be observed to make gestures to themselves, other people in their physical surroundings. Sometimes they can be observed to point at something on the screen as if the other participants in the CVE could perceive this gesture.

External-Manipulate (EM)

Actors may be observed to manipulate something in their physical surroundings or something other than CVE windows and programs on their screen.

External-Navigate (EN)

Actors may be observed to move around in their physical space, whilst being idle in the CVE.

External-Position (EP)

Actors may be observed physically moving into a different position for any number of reasons.

External-Scan (ES)

Actors may be observed to quickly review activity in other windows on their screen or in their physical surroundings, thus temporarily suspending their attention from the CVE.

External-Verify (EV)

Actors may be observed checking other windows on their screen to verify that something has happened either to the CVE or to the network or something different.

Gesture (G)

Actors perform gestures in order to communicate non-verbally, and to support their verbal communications. Although often only a limited set of gestures is available in CVEs, actors can be observed creating their own ‘gestures’ by using their virtual embodiment in creative ways. For instance, actors can sometimes be observed twirling their virtual embodiments around on the spot to indicate their wish to speak to the group.

Gesture-Communicate (GC)

Actors can be observed to use the set of available gestures inside the CVE to support their actions and speech.

Gesture-External (GE)

There is one act that is typically employed to signify being eternally (to the CVE) occupied. In some CVEs a gesture has been pre-programmed for this; the ‘sleep-position’, where actors put their virtual embodiment into a horizontal position to signify the fact that they are temporarily not ‘manning’ their embodiment to the other CVE participants. In other CVEs such a gesture may not be pre-programmed, but actors can be observed manipulating the position and location of their embodiment to signify the same.

Gesture-Manipulate (GM)

Actors may be observed shaking objects or other CVE participants to emphasize or illustrate their speech.

Gesture-Navigate (GN)

Actors may be observed moving their virtual embodiment in the direction they want others to follow.

Gesture-Position (GP)

Actors may be observed standing themselves or an object in a particular spot to emphasize or illustrate their speech.

Gesture-Scan (GS)

Actors may be observed scanning their surroundings as a gesture to emphasize or illustrate their speech.

Gesture-Verify (GV)

Actors may be observed using gestures instead of or in addition to speech to verify something.

Manipulate (M)

Actors manipulate objects in their environment for a variety of reasons. Depending on the object being manipulated the goal of their manipulations may be more or less clearly determined. Objects may also simply be manipulated in the space in order to move them from A to B, or in order to move them out of the view or path of the actor, or in order to test the effects and mechanisms of the manipulation act.

Manipulate-Communicate (MC)

Objects may be manipulated in order to communicate something about them, such as the position, the person who controls the manipulation, to signify the particular object, or to draw attention to the object.

Manipulate-External (ME)

Actors may be observed to manipulate something in the CVE which causes something to happen in the external environment, either on their computer, the network, or in their physical surroundings.

Manipulate-Gesture (MG)

Actors may be observed manipulate something in the CVE in order to perform a gesture or as part of a gesture.

Manipulate-Navigate (MN)

Objects may be manipulated in order to navigate them through the environment. The actor picks up the object and carries it from A to B.

Manipulate-Operate (MO)

Objects may be manipulated in order to operate the available functions on them.

Manipulate-Position (MP)

An object may be manipulated in order to position it carefully in an intended position or in order to get it out of its original position.

Manipulate-Scan (MS)

Objects may be manipulated in order to scan them or the space the object was obstructing.

Manipulate-Test (MT)

Objects may be manipulated in order to test the actors' manipulation skills and abilities or in order to test the particular manipulation actions, which can or must be performed on the object in order to operate it.

Navigate (N)

Actors navigate for many reasons through the CVE. Navigation is defined as directed movement from one spot to another, with a clear start and finish. Navigation can be differentiated from 'Scanning' behaviour by its movement from A to B, where scanning involves looking from A to B, without obvious relocation of the VB.

Navigate-Communicate (NC)

Actors navigate in order to communicate. They may specifically try to find other actors, and position their VB to initiate communication.

Navigate-Explore (NE)

Actors navigate in order to explore the environment. They may be seen moving from A to B and back again, to gain an understanding of the space and their position in it in relation to other actors and objects.

Navigate-External (NE)

Actors may be observed using their input device in other windows than the CVE.

Navigate-Gesture (NG)

Actors may be observed to perform navigational acts in order to exhibit a certain gesture. For instance, in order to start the meeting the chairperson moves their virtual embodiment in front of the blackboard.

Navigate-Find (NF)

Actors navigate in order to find a certain object. They should know approximately where the object is, but they still have to navigate their VB around the space in order to locate it.

Navigate-Manipulate (NM)

Actors navigate in order to manipulate an object in the environment. This could be a door, which the actor needs to go through, or an object on which the actor wishes to act.

Navigate-Position (NP)

Actors navigate in order to position either themselves or an object in their possession. They position their VB in order to act onto something, and the positioning may have to be performed very precise.

Navigate-Scan (NS)

Actors navigate the environment in order to scan from A to B. In order to ba able to scan effectively they may have to navigate to a comfortable position first, but without actually staying in this position very precisely. They may be seen navigating to locate A and B into one view, or test a sweep from A to B, or back up to the furthest corner for the largest overview of the space.

Navigate-Verify (NV)

Actors can be observed navigating in order to verify their position in relation to other CVE participants or the CVE and the objects.

Position (P)

Actors position themselves or objects in the environment for a variety of reasons. Positioning is defined as the precise act of alignment of the VB or the object in relation to something else in the space.

Position-Communicate (PC)

Actors may position their VB in order to communicate with the other actor(s). They can usually be observed positioning themselves facing the other actor(s) they are communicating with, but they may also try to face the object they are communicating about.

Position-Explore (PE)

Actors may position themselves in order to explore the environment. They may be observed trying to encompass as much of the environment in their view as possible in order to explore the relationship between objects in the space.

Position-External (PE)

Actors may be observed moving windows around on their screen in order to put all windows in their most convenient position.

Position-Gesture (PG)

Actors may be observed to position themselves in a particular place in order to signify the expression of a certain gesture.

Position-Manipulate (PM)

Actors may position themselves in relation to an object in order to reach the most advantageous position to manipulate the object.

Position-Navigate (PN)

Actors may position themselves in order to start a navigation act from the most advantageous position.

Position-Scan (PS)

Actors may position themselves in order to scan the environment. They may be observed positioning their VB in order to scan from A to B and back.

Position-Verify (PV)

Actors may position themselves in order to verify their position in relation to others or their own location in relation to objects and the space to themselves. They may be observed to use certain objects to fix their position.

Scan (S)

Actors scan the environment for many different reasons. Scanning is defined by a sweeping movement from A to B, not with the intention of moving the whole VB but with the intention of viewing that (part of) space. Actors may be observed trying to sweep from A to B several times before the can actually perform a smooth sweep.

Scan-Communicate (SC)

Actors scan the environment in order to communicate a location or intention in relation to the location of something else. They may be observed using their viewpoint and a sweep from A to B as an aid in the referencing of locations A and B even if the other actor(s) are not capable to seeing this scanning motion.

Scan-External (SE)

Actors may be observed to scan the external surroundings, that is other than the CVE.

Scan-Gesture (SG)

Actors may be observed to scan the CVE environment as a gesture.

Scan-Find (SF)

Actors scan the environment in order to find something or somebody. They may be see trying to understand their position in relation to other objects in the space, or trying to locate something in relation to themselves or something else.

Scan-Manipulate (SM)

Actors scan the environment in order to manipulate an object. They may be observed trying to scan the space in order to determine to see what they need to see to manipulate on the object successfully.

Scan-Navigate (SN)

Actors scan the environment in order to navigate to a certain location. They may be observed trying to scan the environment in order to determine the shortest path to the desired goal.

Scan-Position (SP)

Actors scan the environment in order to position themselves in relation to an object or other actor in the space, in order to determine the location of themselves or an object in the space in relation to themselves.

Scan-Sweep A-B, B-A, (SS)

Actors scan the environment in order to make a clean sweep from A to B, and back again from B to A (where A or B may be an object, an other actor or a location). They may be observed making this sweeping motion several times during a discussion of A or B. Performing a clean sweep takes an intuitive knowledge of the distance between A and B and the actors can be seen to practice the sweep several times before getting it down to a perfect motion. These sweeps can be full-circle, 180 degrees, a quarter of a circle, and 3/4s.

Scan-Verify (SV)

Actors may be observed to perform a scan of the environment in order to verify that they can see what they expected to see.

Verify (V)

Possibly due to the fact that there is no direct feedback about or visibility and audibility inside the CVE until another participant or object reacts to us. CVE actors can be seen and heard to perform many acts that serve to verify what is happening or has happened.

Verify-Communicate (VC)

Actors may be observed going round asking all other participants whether they can be heard ok.

Verify-External (VE)

Actors may be observed verifying to somebody in their physical surroundings that they are engaged in the CVE happenings.

Verify-Gesture (VG)

Actors may be observed to try out the different gestures that are available to them, and they can be seen experimenting with their virtual embodiment in general.

Verify-Manipulate (VM)

Actors may be observed trying out the effects of manipulating something in the CVE.

Verify-Navigate (VN)

Actors may be observed trying out the effects of navigating towards or around something in the CVE.

Verify-Position (VP)

Actors may be observed trying out the effects of positioning themselves or something else in the CVE.

Verify-Scan (VS)

Actors may be observed trying out the effects of scanning the environment in the CVE.

5.3.3 Scoring Procedure

Firstly, all the participants in the group under observation are given a number. Secondly, who acts in reaction to whom is scored. Whenever the act seems to be addressed to no particular other, or to more than one other, the symbol "0" is used to designate the target. In cases where more than two persons act at once, as when they all laugh together, or shout together, the symbol "0" is used to designate the actor. Thus a general laugh would be scored 0-0 (all to all). A general nodding of heads to actor 1 would be scored 0-1 (all to one). This then, is noted down with the appropriate category for the observed behaviour and the time at which it occurred, and how long the act lasted. The frequency with which classification dilemmas arise is in part a function of the fundamental soundness of the underlying rationale from which the dimensions are derived. Bales (ibid) notes that the question of reconciling interpretation dilemmas is a question of: (1) what does the manual say? and (2) what modifications of the manual are required to anticipate issues of this nature in the future?

The following table (5.7) is an example of a simple scoring sheet for the way in which data was collected for this thesis. 'Time' is the time at which the act occurs. 'Actor' is the actor who performed the act. 'Category' is the act that was performed. This will give an overview of the type of acts that occur, the frequency with which these acts are performed, the time between acts of one individual, and the time between acts of different individuals.

Time	Actor	Category
15:09:00	1	E

15:09:00	1	C
15:09:04	3	S
15:09:09	2	C

Table 5.7: Actor 1 is in Nottingham, Actor 2 in London, and Actor 3 in Stockholm.

In this example it can be seen that Actor 1 is performing two simultaneous acts: 'E', checking an external window, and 'C', communicating to the other participants in the CVE. Actor 3 is performing 'S', a scan act to look around the CVE. Actor 2 is communicating. To gain a larger insight into whom is reacting to whom, and how long a certain act takes, extra information is gathered from the same video-footage. In the next table (5.8), 'Time 1' is the time at which the act starts, 'Time 2' is the time at which the act finishes, 'Time 3' is the time at which an act by somebody else occurred to which the act was a 'reply'. 'Actor A' is the person who performed the act, 'Category' the act which was observed, 'Actor B' is the person or group to whom it was aimed if any, and 'Actor C' is the person who performed an act that triggered the act of Actor A.

Time 1	Actor A	Category	Time 2	Actor B	Actor C	Time 3
15:09:09	2	C	15:09:10	0	.	.
15:09:11	2	C	15:09:14	0	.	.
15:09:11	3	S	15:09:12	0	.	.
15:09:13	3	S	15:09:14	0	.	.
15:09:14	2	C	15:09:16	.	.	.
15:09:14	3	C	15:09:18	0	2	15:09:09
15:09:15	1	E	15:09:28	.	3	15:09:14
15:09:16	3	S	15:09:17	0	.	.
15:09:17	3	S	15:09:18	0	3	15:09:16

Table 5.8: WhoDo Experiment 30-9-98, 15:09:00-15:09:17.

In this example it can be seen that Actor 3, at Time 1: 15:09:14 is communicating to the group (marked '0' in the Actor B column), in reaction to something actor 2 communicated to the group at time 15:09:09. This communication of actor 3 lasts four seconds (from 15:09:14 to 15:09:18). The time lapsed between the meaningfully connected utterances of actor 2 and actor 3 is five seconds (15:09:14 - 15:09:09 = 5 sec.) Actor 1 is in Nottingham (UK), Actor 2 in London (UK), and Actor 3 in Stockholm (Sweden).

5.4 The Experiments

During the COVEN project three different types of CVE network trials took place, using ISDN, ATM, and the Internet to establish connections between users. Figure 5.2 shows the geographic locations for all project partners.

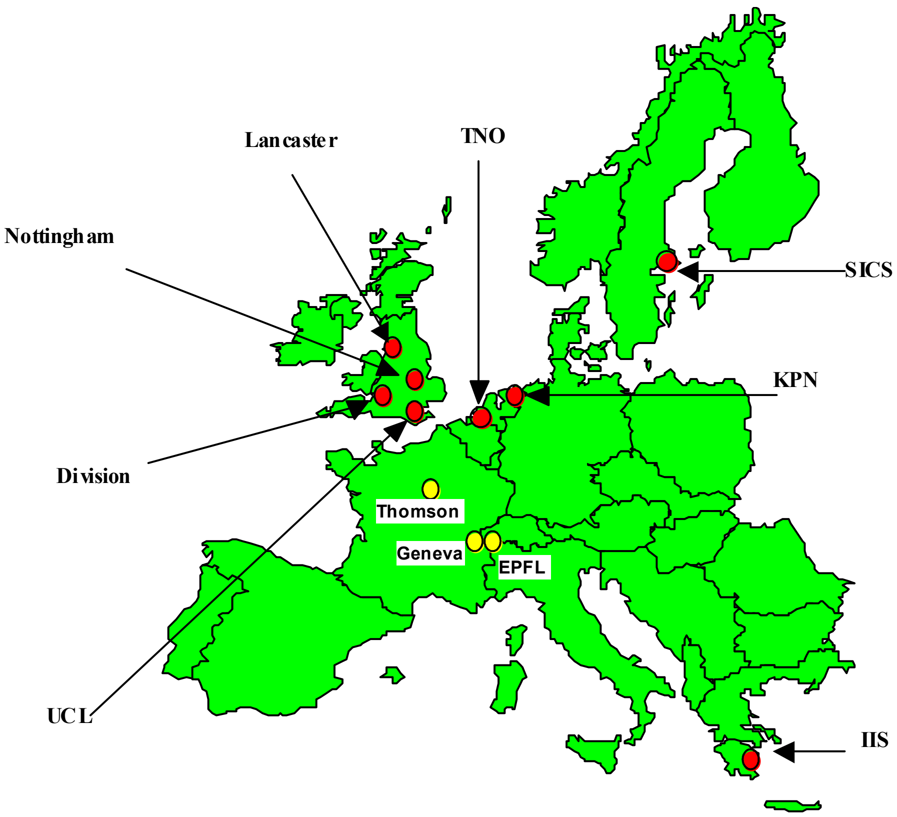

Figure 5.2: The partners represented by a red dot were involved in the experiments described in this thesis.

The initialisation period for each iteration of the network and usability test required much co-ordination; because for the topology as a whole to work, each piece needs to work. The team started out using a telephone conference call to coordinate each trial, because there were initial problems with audio connectivity. To rectify this the participants in the trials were recommended to use either a cardioid stage microphone, or a Gamma headset combination, and for each trial audio levels were adjusted so that everyone could hear each other clearly. A final addition to the infrastructure was the addition of a text chat capability.

5.4.1 The Observation Material

To measure usability and network activity of the COVEN platform, a number of experimental tasks were designed for the network trials of the COVEN platform, one of these tasks was a business trading game used when creating the categories for the observation method. To measure collaboration in CVEs, the author of this thesis suggested to the COVEN team to develop a virtual version of a murder mystery board game called Cluedo™. The virtual game was considered to be sufficiently collaborative to be suitable for analysis of collaborative behaviour, and provided a continuing test-bed for CVE functionality. It was built by Anthony Steed of University College London (UCL), and called "the WhoDo game" (further explained below), and it was used during a large number of the network trials.

During each trial video recordings were made of the video-out signal of a user's CVE interactions, visible through the CVE window on the desktop computer screen, which were subsequently used for the observational analysis. Care had to be taken for the on-screen clock to be visible at all times, because it was used to pinpoint the temporal aspect of the analysis.

5.4.2 The Hardware and Software

The four sites that took part in the WhoDo trials, and the hardware that they used are shown in figure 5.2. To start with, one person per site was engaged in the trials, rising to four per site by the final trial. Standard Internet networking was used, with the Dive software being supported by the used of an application level multicast network, the DiveBone (Frecon, Greenhalgh, Stenius, 1999). Dive is designed from the outset to support Internet networking and the set-up phase and the trials proceeded smoothly. The fact that in the final trial, 16 participants were engrossed in an hour long trial without major incidents was considered to be a major achievement (Del3.6).

Development of the WhoDo game was suggested by the author, to provide an interesting task that could repeatedly involve up to 16 people for an hour. It was created to run on top of DIVE. The game is based on the original board game Cluedo™. Our version exercises within-team collaboration and between-team competition and cooperation. It has one major difference from the original game in that the board game uses a predetermined order for each player's turn and the dice as a randomising factor in the continuation of the turn. The WhoDo game leaves the turn taking completely to its participants to determine. It is also different in that the 'players' neither have a luck-, nor a turn-taking mechanism such as the dice in the original board game version, which means that they have to move through the mansion guided by their own initiative, to find each other, before they can 'play a turn' in the game. Team-members can collaborate via a group text window. Otherwise, it is effectively a 3D version of the 2D board mansion layout. The aim was to evoke as many interaction and collaboration styles as possible, by introducing the aspects of team play, and one-on-one collaboration, not unlike realistic business settings. Figure 5.3 shows a snapshot from birds' eye view of the WhoDo world.

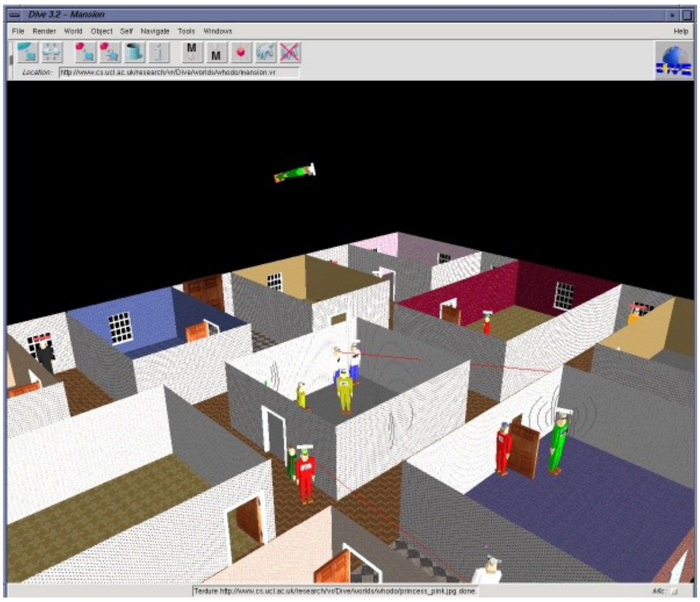

Figure 5.3: An overview of the WhoDo mansion

5.4.3 Experimental Set Up

The WhoDo world was developed, and initially tested with fifteen users. The users at each site formed a team, and were associated with one of the suspects. Each team was then given a set of cards relating details about who committed the murder, where and with what weapon. Members of each team were then free to roam and question other people in order to try and ascertain the information relating to the murder. A number of world and game problems were uncovered, and subsequent redevelopment occurred. One suggestion was some means of private communication for members of a particular group. SICS provided an implementation of some private text communication windows. An overview of The WhoDo usability trials that took place is shown in table 5.9.

Date	Activity	Participants	Usability Questionnaire	Fit for Analysis
16-09-1998	WhoDo game	15	5	
30-09-1998	WhoDo game	16	0	Yes

05-05-1999	Testing new version		1	
06-05-1999	Testing new version		1	
16-06-1999	WhoDo game	14	2	
22-06-1999	Nott host site test		0	
30-06-1999	WhoDo fiasco	9	4	
07-07-1999	WhoDo 4 novice users	4	2	Yes
21-07-1999	WhoDo 3 sites/2 players per team	6	0	

Table 5.9: Usability trials undertaken with WhoDo game

After every trial each participant was asked to complete the same collaboration usability questionnaire, thus capturing longitudinal qualitative data from the participants about their attempts to collaborate (see Appendix D). Quite often a trial would have such a long set-up period that the whole trial would last much longer than two hours before the game could commence, which caused participants to run out of work time. This meant that not all participants were equally faithful after every trial, in answering the questionnaire. Since the quality of the responses depended on the immediacy of the experience these additional data have to be considered 'lost'.

5.4.4 The Experimental Task

The task description could be considered to exist in several parts. Firstly, the persons involved in the trial at each site received instructions by email. These instructions detailed how to behave towards experimental subjects (such as during the 7-7-99 trial). Secondly, the game rules were placed on the WWW and the local coordinators were asked to provide the URL plus a printed out version of these rules to any new comers. The task was explained as follows:

"The aim of the game is to guess who killed the "Dead Dude" who is lying in the central room. Participants play in teams and must collaborate with their team mates and interact with the other teams to be successful.

You must guess who did the killing, what item they used to do the murder and where the dastardly deed took place. There are six suspects in the mansion, six possible weapons and nine rooms.

You progress through the game, by moving around the mansion asking questions to players from other teams. They are not obliged to answer you (or even acknowledge), but of course you can set up "deals" where you will only answer a question if they will. Of course if everyone is being obstinate, the game might go on for a while.

You may submit a form with your suspicions about the crime at any point for checking, though you don't get a second chance if you get it wrong. That is, if you guess incorrectly, you have immediately lost but you may still answer questions for other people. Of course, if there is more than one in your team, you can try more than one solution.

The winners are the team to first guess the correct solution. You may play on to determine rankings or decide to re-start straight away."

The main tasks in the game were to collaborate with the other teams by asking questions, and to communicate privately with the team members to form a strategy to solve the murder mystery. Figure 5.4 gives a snapshot of the start of a game.

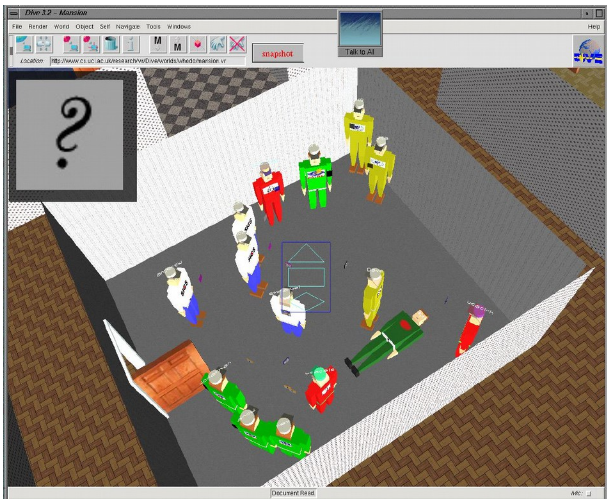

Figure 5.4: Participants gathering at the start of a game.

5.4.5 The Subjects

The subjects in the initial trials were programmers and developers of CVE technology and computer science students. Because of the small number of available subjects, the fact that they are obviously not representative end-users, and the exploratory nature of prototype development, the subjects were approached in their capacity of CVE experts, and their behaviours and questionnaire responses have to be analysed from this perspective. Developers of CVEs are pioneers of the technology and as such have very informed and detailed experiences with this new technology. Every effort has been made to elicit this knowledge during the trials. Because of the relatively small number of available subjects, repeated measures were made, using a longitudinal usability questionnaire so that a large number of observations are obtained from each person in order to more reliable results. However, these subjects had to be repeatedly reminded that even though the questions remained the same throughout the trials it still remained necessary to answer the questionnaire after each trial.

5.5 Conclusions

The continuous, unpredictable stability of the COVEN platform due to changes and additions did not allow easy testing with more representative end-users. However, the ultimate goal always was to reach a stable enough state to test the application with novice users only, which was eventually reached on the 7th of July 1999. These novice users received a 10-minute training on how to use the interface to DIVE in order to allow them to get on with the actual task of collaboration.