16

Communication Support for Knowledge-intensive Services

J. Berghoff, J. Schuhmann, M. Matthes, O. Drobnik
Department of Computer Science, University of Frankfurt/Main
Robert-Mayer-Str. 11-15, D-60054 Frankfurt (Germany), email:
{juergen, josefine, sascha, drobnik}@tm.informatik.uni-frankfurt.de

Abstract

Knowledge-intensive services gain significant importance for the development of innovative products. Up to now they depend mainly on face-to-face communication. Technological progress in artificial intelligence, groupware, and broadband systems favour computerized support of such services. In this field we focus on the computerized communication support for human interactions. We try to enrich human communication through 3D visualization of problems and research artefacts. A prototype system is under development to assess the required network capacity. It consists of tools for knowledge modeling and group interaction supplemented by a tailor-made multicast protocol.

Keywords

Network, CSCW, Graphics, Multicast, Collaborative Learning

1 INTRODUCTION

The advent of multimedia applications combined with networked desktop computers has shown a gap between communication needs of modern applications and current communication protocols (services). With respect to the requirements of real-time multimedia applications a variety of protocols and QoS models have been developed (for example: RTP, RSVP etc.) to bridge the gap. Besides the types of data which determine the communication needs, the kind of interaction has to be taken into account: one-to-one, one-to-many, many-to-many. The concurrent interaction of more than two entities which is typical for many CSCW-applications require adequate multicast protocols.

Driven by the World Wide Web and its needs for visualization of more and more complex objects and environments, new 3D visualization and animation techniques (VRML) enter the scene. This leads to new communication requirements. Again the gap between application and communication layer demands an integrating solution (e.g. Rhyne (1997), Brutzman (1997)).

On the application level, we have focused our research on groupware to

Broadband Communications P. Kühn & R. Ulrich (Eds.)

support knowledge-intensive services. Such services are characterized by non-standardized human to human communication. We try to support this kind of communication by means of computerised 3D-visualization techniques and teleconferencing tools.

2 KNOWLEDGE-INTENSIVE SERVICES

Knowledge-intensive services gain significant importance for product development, eg., in the industry or financial enterprises to increase innovation. They aim in particular at the development of new products starting with rather rough ideas about the final product. Therefore, their degree of standardization is low and the individual processes to solve the problem are highly specialized. Often the product is developed and detailed in tight interaction between customer and service provider. Thus, solutions and innovations are the result of a collaborative learning process of both parties. The support of knowledge-intensive services is not restricted to the interaction between costumer and provider but also considers the production process itself. The spectrum of cooperation scenarios between firms (joint venture) or between developers (teamwork) is rather broad.

Non-standardized communication in all phases of a business relation is typical for such services. The complexity of a knowledge-intensive service comes from the service description on one side, and from the development and mediation of the solution on the other side.

Because of the highly interactive character of knowledge-intensive services their workflows include several face-to-face meetings. Face-to-face communication does not consist only exchanging information, which could be done by simple messaging solutions like email, but is often necessary for the process of finding general agreements, making decisions, getting a common understanding of problems and developing solutions.

At a first glance, face-to-face meetings could be supported by multimedia conferencing systems. But such systems must be enriched by tools to illustrate problems and points of discussion. These tools should support the collaborative construction of concepts for problem solving and the description of research artefacts. Since the kind of communication process immanent to knowledge-intensive services could be considered as an act of collaborative learning, solutions from this research area could be adopted. An important subject in collaborative learning is the modeling and representation of knowledge.

A widespread form of knowledge representation are so called "concept maps". A concept map is a diagrammatic representation which shows concepts (as nodes) and meaningful relationships (as links) between the concepts. Facts, evidences etc. are represented as nodes and are connected to each other via links. The link type reflects the kind of relation between the connected nodes.

Important topics are the process of creating knowledge and the representation of knowledge. Especially in collaborative knowledge construction the aggregation of individual views (from individual persons) of the problem to a commonly accepted representation is an important task. A system that addresses both problems is CLARE (Wan 1994), which stands for "Collaborative Learning and Research Environment". CLARE is a distributed learning environment and incorporates two semi-formal methods: RESRA and SECAI.

RESRA ("Representational Schema of Research Artifacts") is a semi-structured knowledge representation language designed specifically to facilitate collaborative learning from scientific text. RESRA can be seen as a special class of concept maps, where each node and link is of a certain type of a predetermined set of node types and link types. An example of an abstract RESRA-representation is shown in figure 1.

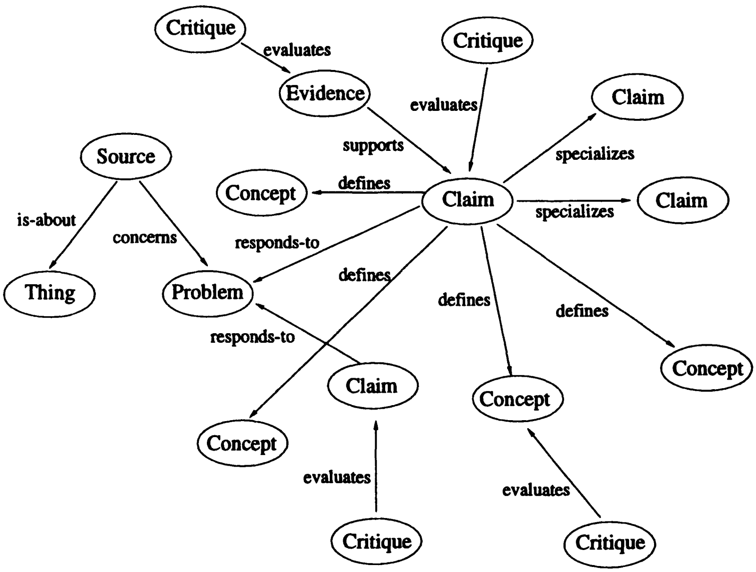

Figure 1 *RESRA example.*

SECAI ("Summarization, Evaluation, Comparison, Argumentation, and Integration") defines two phases for collaborative learning from scientific text: *exploration phase, consolidation phase* (cf. 3.1).

The concepts of our supporting tools are based on the ideas of CLARE. We are developing a system for collaborative modeling and visualization of problems and research artefacts. It will be integrated into a multimedia conferencing system to supplement or replace conventional face-to-face meetings. The overall concept of our approach is presented in more detail in the following section.

3 AN INFRASTRUCTURE FOR COLLABORATIVE LEARNING

Our infrastructure consists of a distributed meeting system for concurrent interaction of humans and incorporates video and audio capabilities for video conferencing and presentation tools for 3D-visualization of knowledge. Figure 2 shows an overview of the infrastructure. A central component is the control unit, which runs on the workstation of each participant (group member). The control unit manages the interaction between group members and controls the supporting applications. The video and audio components exist on each workstation and exchange data directly via multicast. In a first prototype, the visualization components for knowledge modeling are connected to a central server (a distributed version is planned). All actual representations of knowledge are stored in this server.

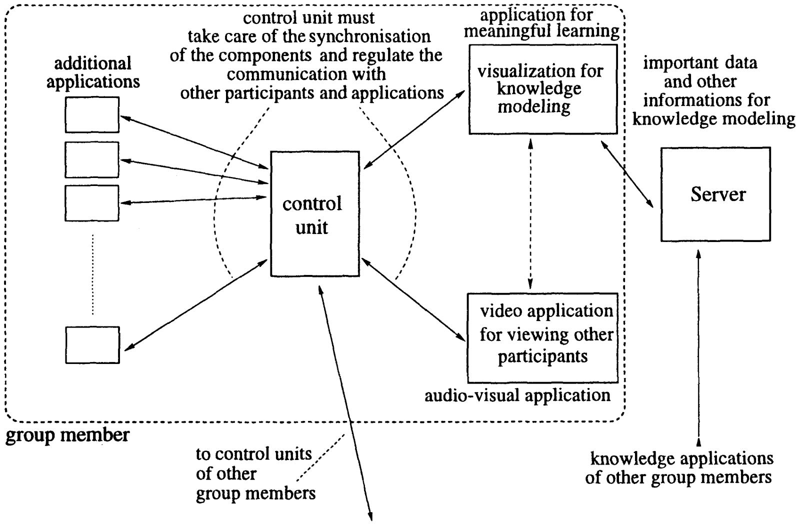

Figure 2 *Infrastructure overview.*

Several additional applications can be connected to the control unit. Most of these applications are collaboration-aware and consist of a number of replicated components (one on each local system of the participants) which communicate and interchange information via their own communication facilities. Examples of such applications are Whiteboards, Web-Browsers and other presentation components. The control unit is able to start applications and controls their behaviour in dependence of a role scheme. A participant has a certain role in a group. An example of roles and a role scheme is given in the virtual seminar section. Often data streams of different application components have to be synchronized. Synchronization information is distributed by the control units among the application components. One of the application components functions as a synchronization master and sends synchronization events to the control unit. Furthermore the control unit has the following functions: *role management and application control, member management, distribution of synchronization information.*

In the following section we concentrate on those components developed especially to support knowledge intensive services.

3.1 Knowledge modeling and representation

In conventional systems, concept maps have a 2-dimensional graphical representation. For complex problems or artifacts a map may become large and meshed and is difficult to understand. We propose a 3-dimensional interactive map representation to overcome these difficulties. Our implementation is based on VRML 2.0 (e.g. VRML96 (1996), Hartmann (1996)). VRML 2.0 enables the description of dynamic and animated worlds which responds to the actions of the user (e.g. if the user clicks on a door bell he could hear the corresponding sound). The possible actions of the user are registered by so called sensors which dispatch the events to the right nodes (In the example a *touch sensor* registeres the mouse klick and dispatches a corresponding event to a *sound node*). Another possibility to respond to a user action is to start a (small) program via *script nodes*. Script nodes contain either a URL at which the program could be found, or the source code itself. The programs can be written in *JavaScript* or *Java* (Descartes 1996). We use a VRML 2.0 viewer with support for *Java* script nodes.

(a) Exploration phase:

In this first phase all group members work on their own to build a concept map. Nodes and links can be built, modified and deleted via a graphical user interface. For each node or link a textual annotation of any length can be stored. The input of the graphical user interface is transmitted to a so called *file generator* which modifies and updates the VRML-file. The file generator stores all informations in a database. After updating the VRML-file the file generator signals the viewer to update its display. To view the information associated with a node or link, the user simply clicks on the appropiate element on the screen, and the associated information is displayed in a separate window. Figure 3 shows the architecture of the knowledge modeling system. A canonical form of a concept map is displayed to help the user in building a representation of the problem.

(b) Consolidation phase:

In a second phase all group members work together to aggregate the individual concept maps to a common representation of the problem. Differences and conflicts are discussed using multimedia conferencing capabilities. The knowledge modeling systems of the participants are connected to a central server

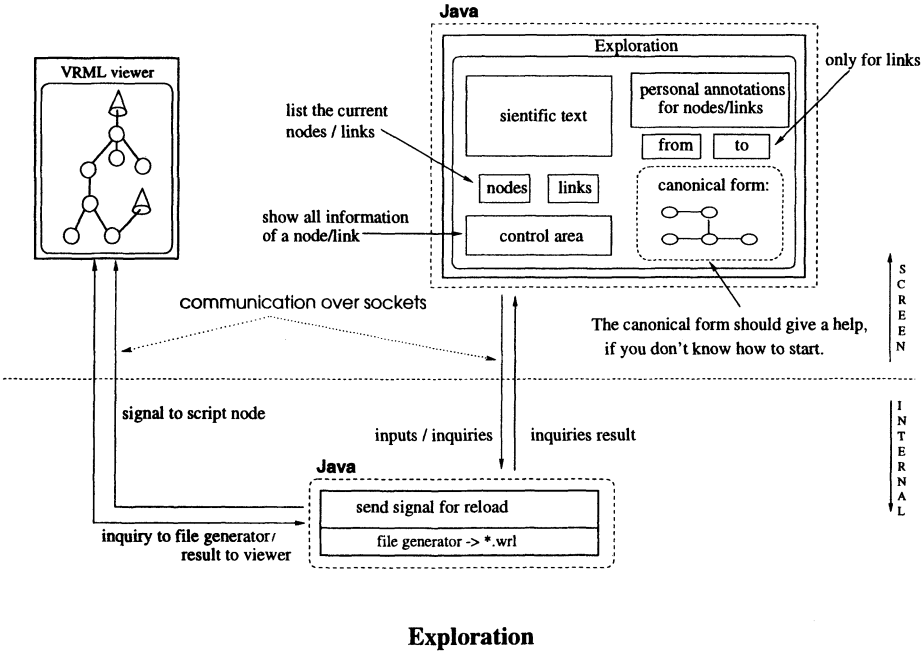

Figure 3 *Architecture of the knowledge modeling system.*

which contains a common database. The access to the various representations is synchronized by the server. Updates and modifiactions of the representations are distributed by the server to each participant (point-to-multipoint communication).

The consolidation phase consists of many interactions between group members. Besides the exchange of audio and video information through the conferencing system the concurrent manipulation of the representations requires multiple and timely updates of the displays of all involved persons. A reliable multicast protocol is necessary to ensure the consistency of all displayed representations.

4 VIRTUAL SEMINAR

We implemented the *Virtual Seminar System* to evaluate the general infrastructure for collaborative learning. It will also be used to estimate the QoS-requirements. Each participant is provided with a replicated software environment which enables him to participate in a seminar from his home terminal. The communication mechanisms required for the data transfer are based on the services provided by the Internet.

The system encompasses several components which support audio-visual communication among the participants as well as the presentation of papers and charts. The role of the participants can change in the course of the seminar and determines which possibilities of interaction a participant has at any given time during the seminar. The structure of the Virtual Seminar System is shown in figure 4.

The control unit is responsible for the administration of the participants as well as the allocation of the roles. Each component has an interface which distributes information about the current roles of the participants.

Audio data synchronize the media information flow. As the human brain is most sensitive to disruptions in the perception of audio data, it is essential that speech sequences are replicated as clearly as possible. The audio component distributes local synchronization events for synchronizing the replication of audio data with the remaining media information flow.

The cross-system data exchange between the components is realized by using the IP-Multicast protocol which reduces the bandwidth required for distributed real-time group applications.

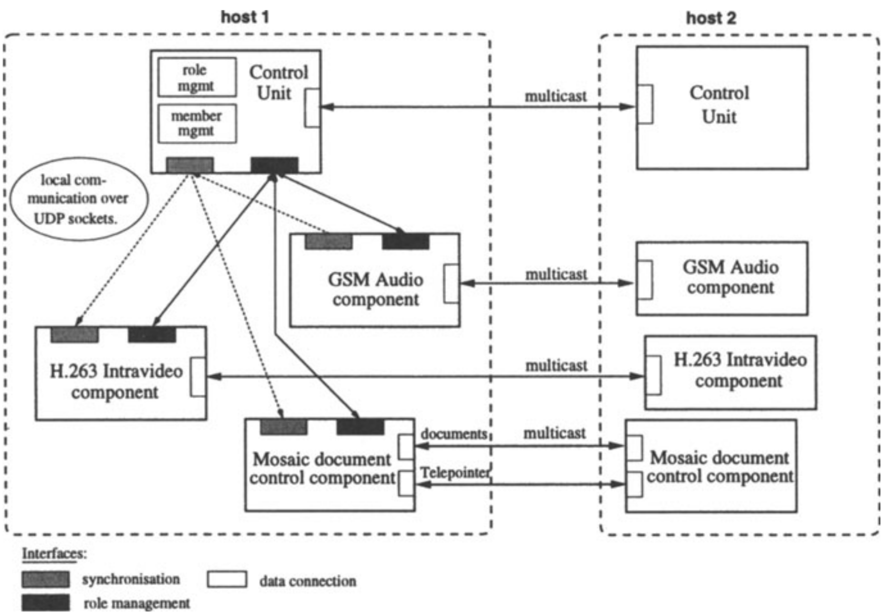

Figure 4 *Virtual Seminar Architecture.*

4.1 Role Administration

The role scheme reflects the hierarchical allocation of tasks in normal seminars. There are three active roles which can be held by a participant: ***initiator, lecturer, talker.***

The initiator starts the seminar with an introduction into the subject of the respective seminar and allocates the role of the lecturer.

The lecturer presents a paper on a certain subject. He can allocate the role of the talker temporarily to another participant who has a question or comment.

The role of the talker is held by the initiator at the start of the seminar and by the lecturer during the presentation of the paper. It can be allocated to other participants to ask questions.

Participants who do not hold any active role are listeners.

The *Control Unit* administers and allocates the three active roles and is therefore responsible for the consistency of the distributed group application. At any given time during the seminar, each of the three active roles can be held by one participant only. This requires special control mechanisms such as the use of a protocol for the allocation of active roles. This protocol has been specifically adjusted for the Virtual Seminar. It uses the IP-Multicast protocol for addressing different participants simultaneously and uses other reliable Unicast protocols where direct point-to-point connections would be sufficient. That way some of the typical problems which tend to occur in reliable multicast protocols can be avoided. The application-specific protocol meets the requirements of scalability and robustness.

4.2 Application components

The application components of the *Virtual Seminar* support audio-visual communication and the presentation of papers and charts. Data sequences are transferred by using the unreliable IP-Multicast protocol. The loss of single data packets in audio and video data flows can be tolerated as the information of the individual data packets are only needed for playback at a certain time. Delayed data can be discarded. Due to the real-time character of the *Virtual Seminar* it is most likely that data sequences which are transferred repeatedly are received after they are needed for playback. As this would imply additional load to the network, it is sufficient to use unreliable data transfer methods. In order to reduce the required network capacities the media data sequences are compressed before transfer.

(a) Video component

The video component realizes the transfer of video images of the lecturer or talker. The picture formats sub-QCIF (128x96) and QCIF (176x144) are supported. The compression is based on the intra compression method for video images which includes the Discrete Cosinus Transformation (DCT), quantisation and subsequent variable length encoding according to Huffman. As the compression is a pure software solution, there is no compensation of movement by calculating motion-vectors. For a higher compression rate a mechanism has been integrated which recognizes motion and only transmits those parts of the image which have changed. Moreover, there is a successive refresh procedure which updates the entire image periodically. Thus the refresh procedure prevents errors in the transmitted image.

(b) Audio component

For the transfer of speech sequences the data are compressed according to the European GSM 6.10 standard used for digital mobile telephone communication in Europe which is particularly suitable for the compression of speech. The audio component recognizes speech activities so that only active speech sequences are transferred and pauses do not take up unnecessary network capacities.

(c) Document component

The WWW-Browser Mosaic is used to present papers and charts in the *Virtual Seminar*. The browser can be accesed via its CCI interface. A telepointer feature has been integrated in the browser to provide higher comfort for the lecturer. The telepointer uses its current position in the document as orientation and moves the displayed part of the document in the receiving browsers accordingly. In their browsers the participants can therefore see the part of the document which is currently presented.

The procedures and components of the *Virtual Seminar* support an efficient collaboration of group members via the Internet. The specific needs of the consolidation phase have to be considered for the allocation of roles to group members.

5 MULTICAST PROTOCOL

An efficient infrastructure for groupware should make use of the multicasting capabilities of the network. Designing universal multicast protocols for a broad range of services is difficult. Diot (1997) gives a survey of multicast protocols and functions. We decided to design an application-specific multicast protocol based on IP-Multicast (Deering 1989).

The IP-Multicast protocol provides a service which is unreliable. There is no guarantee for a correct and complete data transfer. The IP-Multicast protocol can be extended by adding suitable control mechanisms to provide reliable data transfer.

When implementing reliable multicast services, the mechanisms used by IP-Unicast protocols (TCP) cannot be used as the scalability of the protocol would be poor (the protocol overhead would be too high in lage multicast groups). If every member of a group has to acknowledge the receipt of every single data packet to the sender, the result will be an acknowledgement implosion (Figure: 5).

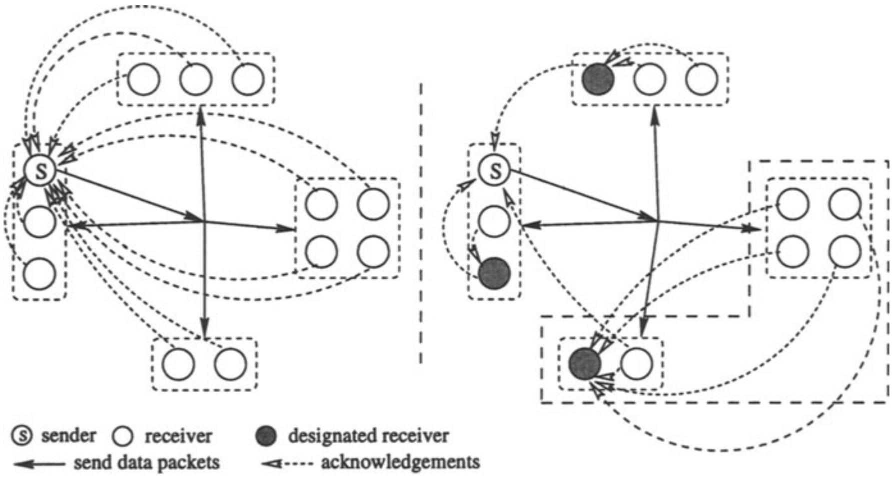

Figure 5 *Acknowledgement Implosion and sub-groups.*

One way of avoiding such situations is to define sub-groups. In this case several group members acknowledge the receipt of data packets together. Another solution is to confirm data transfers after individual delays. As a result the sender does not have to process all data transfer acknowledgements at the same time.

In addition to the extension of mechanisms for reliable data transfer, special processes need to be designed to guarantee that the transmitted data sequences are received in the correct order. This could be ensured by using sliding windows methods with numbered data packets to obtain the so-called source order. If the objective is to secure the correct order of all data sequences transmitted by different senders to the same group, i.e. the so-called total order special control structures need to be developed. This could be achieved by giving data sequences to be transferred global sequence numbers (e.g. Armstrong (1992), Bormann (1994)) or by giving already transmitted data sequences relative time marks (Birman 1991).

In the following we outline our multicast protocol, which was designed to provide high scalability and robustness. A protocol is considered robust when its performance is not affected by temporary breakdowns of systems involved in the data transfer.

Scalability: The allocation of the three active roles according to the role scheme has to be carefully synchronized. This requires a reliable data transfer.

Before allocating an active role to a person it has to be taken from the person who is presently holding it. This is realized by using a reliable point-to-point connection with the current role holder. All participants have to be informed about this change so that the components can adjust to the new allocation of roles. The allocation is therefore done by using the unreliable

IP-Multicast protocol. Only the affirmative acknowledgement of the new role holder is necessary to secure the correct allocation of the active roles.

Using the unreliable IP-Multicast protocol for the allocation of roles implies that not all participants are informed about the role changes if the data packets about the new roles are lost. However, the components are capable of recognizing that a role change has taken place as soon as they start receiving data sequences from a new sender. In this case a status request about the current role allocation is issued. This request is sent via the IP-Multicast protocol, thus all participants are informed and do not have to issue the same status requests. A status request is delayed briefly to avoid negative acknowledgement implosions just in case another participant has already sent a request.

Robustness: When the data transfer connections between the participants in a seminar are temporarily down, the protocol should reestablish the connections once the systems are back on line. The problem is the consistency of the overall status of the distributed group application, i.e., the allocation of the current roles.

If a participant who does not hold an active role is disconnected, the course of the seminar is not affected. The participant can join the seminar again as soon as the connection has been reestablished. If required, a status request can be issued to inform the participant about the current allocation of roles.

However, if a participant is disconnected who is holding an active role the seminar cannot be continued. The active role which has been lost cannot be allocated to a new person, as it has not been taken from the previous person. In that case the holder of the hierarchically higher role can give the lost role to another participant without acknowledgement so that the seminar can continue. As long as the connection to the previous holder of the role is down, the consistency of the group application is maintained. As soon as the previous holder is back on line, however, the active role is allocated twice. If the previous holder of the active role receives data sequences from the present holder or if the other participants receive data sequences from the previous holder, status requests are issued. As a result a status report on the present allocation of roles is transmitted and a consistent status is reestablished.

The application-specific protocol developed for the *Virtual Seminar* is therefore robust in case of system breakdowns and supports the continuing of the seminar in spite of breakdowns. Participants who have been disconnected can be reintegrated in the seminar. Thus the protocol does not only tolerate errors but addresses errors efficiently as well.

6 CONCLUSION

In this paper we introduce an infrastructure to support knowledge-intensive services. We use modern 3D-visualization techniques to enrich the human-to-human interface and integrate the knowledge modeling tools into a multimedia conferencing system. On the communication side we propose customized multicast protocols for efficient data exchange. A prototype system is under development; the virtual seminar and the exploration phase already exist. Experiments with the prototype are planned to assess the required network capacity. Part of this work (especially those focused on knowledge intensive services) is done in the context of the research program "Competitive Advantage by Networking" at the University Frankfurt (for more information about the research program see http://www.vernetzung.de). Further research will be done to advance the process of collaborative knowledge modeling and to support heterogeneous system environments.

REFERENCES

Descartes, Alligator (1996) Interfacing Java and VRML. *UK Java Developer's Conference Paper,* Nov. 1996.

Hartmann, J. and Wernecke, J. (1996) The VRML 2.0 Handbook. Addison Wesley Developers Press, 1996.

The Virtual Reality Modeling Language Specification (1996). Version 2.0; ISO/IEC CD 14772; August 1996.

Wan, D. Johnson, P.M. (1994) Computer Supported Collaborative Learning Using CLARE: *the Approach and Experimental Findings.* Proceedings of CSCW 94; New York 1994.

Rhyne, T-M. Brutzman, D. Macedonia, M. (1997) Internetworked Graphics and the Web; Computer, August 1997.

Bormann, C. Ott, J. *et al* (1980) MTP-2: Towards Achieving the S.E.R.O. Properties for Multicast Transport; ICCCN 94, San Francisco, Sept. 1994.

Armstrong, S. Freier, A. Marzullo, K. (1992) Multicast Transport Protocol; RFC 1301, 1992.

Birman, K. Schiper, A. (1991) Lightweight Causal and Atomic Group Multicast; ACM Transactions on Computer Systems, Vol. 9, No. 3, Aug. 1991.

Brutzman, D. Zyda, M. Watsen, K. Macedonia, M. (1997) virtual reality transfer protocol (vrtp) Design Rationale. Workshop on Enabling Technology (WET ICE), Cambridge, June 1997.

Diot, C. Dabbous, W. Crowcroft, J. (1997) Multipoint Communication: A Survey of Protocols, Functions and Mechanisms. IEEE Journal on Selected Areas in Communications, Vol. 15, No. 3, April 1997.

Deering, S. (1989) Host Extensions for IP Multicasting. RFC 1112, August 1989.