

This is a translation of a paper accepted to the conference: "Interaktion im Web: Innovative Kommunikationsformen" in Marburg, Germany. The German version is entitled: "Verschränkung von Perspektiven durch Aushandlung."

The Sharing of Perspectives by Means of Negotiation

Thomas Herrmann¹ and Gerry Stahl²

¹ Informatik und Gesellschaft, Universitaet Dortmund

² Center for LifeLong Learning and Design, University of Colorado

ABSTRACT

Collaborative work typically involves both individual and group activities. Individual efforts may be combined through group negotiation processes. Personal perspectives on shared information are thereby intertwined and merged into group understanding.

Computer support for personal and group perspectives allows people to view and work on a shared information repository in different contexts. The challenge for perspective mechanisms is how to manage the merging of results from individual perspectives into a consensus. Computer support for negotiation addresses the problem of merging individual results through voting and discussion. The challenge for negotiation mechanisms is how to allow work to continue while issues are being negotiated. WEBGUIDE is a software prototype that combines support for perspectives and negotiation so that individual results can be systematically merged while work continues within personal perspectives.

A WEBGUIDE interface has been designed in detail to work out the many issues involved in intertwining perspectives and negotiation mechanisms. The application domain for illustrating WEBGUIDE is the support of web research by teams of middle school students. The WEBGUIDE system makes explicit and scaffolds for these students the structure of web research, team collaboration, personal perspectives, and group negotiation that they may be experiencing for the first time in their lives.

TABLE OF CONTENTS

Abstract	1
Table of Contents.....	1
Introduction	2
Approach.....	2
Previous Work	3
Related Work	4
Hypertext and Hypermedia.	4
Database Views	5
Context Mechanisms.....	5
Computer Supported Collaborative Learning	5
Organizational Memories.....	5
GroupLens.....	6
Conflict Management.....	6

Decision and Meeting Support	6
Agent Systems	6
Examples of the Need for Intertwining Perspectives and Negotiation	7
Maintaining a Bibliography	7
Developing a Glossary	7
A Student Research Project	8
The WebGuide Prototype	9
System Goals	9
A Use Scenario	10
A Use Scenario	Error! Bookmark not defined.
Support of Individual Work	12
Support of Proposals for Negotiation	14
Conclusion	16
References	17
Acknowledgements	19

INTRODUCTION

Approach

The World Wide Web (the web) provides an obvious medium for supporting collaborative work. It makes available a rich, open-ended source of information that can be used in collaborations. Each person engaged in collaborative activities adopts his or her own perspective within which this information is used. The problem is how to create a collaborative group view from these individual perspectives.

The term *perspective* means that a particular, restricted segment of the information source is being considered, stored, categorized, and annotated. In the realm of collaborative work, it is important to intertwine these personal perspectives with each other. An individual can do this by taking into account the perspectives of others or by adopting part of another person's perspective within one's own. It is also possible to merge several perspectives into one common one. In this paper, we explore the possibility of providing computer support for intertwining perspectives for use in collaborative work.

In the case of either professional or student research teams, it is likely that information from the web on a given topic would first be gathered through the personal work of individual team members. At some point, the team members would then have to bring this information together into a coherent whole. They would have to decide which items of information should be made part of their shared work. Such a decision typically requires a process involving many steps of establishing mutual understanding and negotiating agreement. Certain of these steps can be supported by appropriate groupware mechanisms. Where the negotiation involves perspectives related to information on the web, it may make sense to integrate support for the negotiation into a web-based medium.

We propose an approach to intertwining perspectives by means of web-based computer-supported negotiation. This helps members to merge their web-based information sources into a team perspective, so that the information serves group needs as well as individual purposes.

Our approach combines previous research we conducted individually on computer support for perspectives (Stahl, 1993) and for negotiation (Herrmann, 1995). Computer support for perspectives allows people in a group to interact with a shared information source; each person views and maintains their own perspective on the information without interfering with content displayed in the perspectives

of other group members. The problem is that perspectives of group members tend to diverge instead of converging as work proceeds. Computer support for negotiation, on the other hand, allows members of a group to communicate about what information to include as mutually acceptable. The problem here is that negotiation support does not help people to work on the information while potentially lengthy negotiations are underway concerning the information. Here, perspectives provide a solution, allowing work to continue within personal perspectives while the contents of shared perspectives are being negotiated.

We believe that perspectives and negotiation are each important CSCW concepts in their own right, but that when combined they can offset each other's major weaknesses and provide powerful support for shared information sources. We propose an approach to intertwining the mechanisms of perspectives and negotiation to help collaborative groups intertwine the personal perspectives of their members into an effective shared network of perspectives on task-relevant information.

Previous Work

This paper combines two previous approaches: Stahl's (1993) collaboration using perspectives, and Herrmann's (1994, 1995) negotiation of shared information. The most important characteristics of Stahl's perspectives are:

1. Individual team members always have access to what appears to be their own information source. This is called their *personal perspective*. It consists of items from a shared central information repository that are tagged as being visible within that particular perspective (or in any perspective inherited by that perspective).
2. Team member A can integrate a node into her perspective by creating a *virtual copy* of the node from B's perspective. If B modifies the original node, then it changes in A's perspective as well. However, if A modifies the node, a new node is actually created for A, so that B's perspective is not changed. This arrangement generally makes sense because A wants to view (or inherit) B's node, even if it evolves. However, B should not be affected by the actions of someone who copied one of B's nodes.
3. Alternatively, team member A can *physically copy* the contents of a node from B's perspective. In this case, the nodes are not linked to each other in any way. Since A and B are viewing physically distinct nodes now, either can make changes without affecting the other's perspective.
4. When A creates a virtual copy of a node from B's perspective, A can decide if she will also get virtual copies of nodes related to that node, or if she will create her own subnetwork for her copy of that node. Arbitrarily large subnetworks of information can be inherited with no overhead in time or memory using the virtual copy mechanism.
5. Nodes and links can be created, edited, or deleted by users within the current working perspective.
6. New perspectives can be created by users. Perspectives can inherit from existing perspectives. Thus, a team perspective can be created that includes virtual copies of all contents of the inherited perspectives of the team members. There is an inheritance tree of perspectives; descendants inherit the contents of their ancestor perspectives. Changes (additions, edits, deletions) in the ancestor are seen in descendent perspectives, but not vice versa. A hierarchy of team, sub-team and individual perspectives can be built to match the needs of a particular application.

This model of perspectives has the important advantage of letting team members inherit the content of their team's perspective and other information sources without having to generate it from

scratch. They can then experiment with this content on their own without worrying about affecting what others see. This is advantageous as long as one only wants to use someone else's information to develop one's own perspective. However, if one wants to influence the content of team members' perspectives, then this approach is limited because one cannot change someone else's content directly. It is of course important for supporting cooperative work that the perspectives maintain at least a partial overlap of their contents in order to reach successful mutual understanding and coordination. The underlying subjective opinions must be intertwined to produce an intersubjectivity (Habermas, 1981, p. 28). It has frequently been noted in computer science literature (e.g., Floyd, 1992, p. 91) that different stakeholders engaged in the development and use of a system (e.g., designers, testers, marketing, management, end-users) always think about and judge issues from different perspectives and that these differences must be taken into account.

The concept of computer-mediated negotiation addresses the problem of making changes to a system design or an information source when the changes will conflict with the interests of others. Such a change must first be proposed by someone. The same software that is used to prepare and propose the change should also inform the people affected and help them to respond to the proposal. According to Herrmann (1995), the following options should be offered:

1. Accept the proposal.
2. Reject the proposal.
3. Modify the proposal.
4. Accept the proposal until it is withdrawn.
5. Interrupt the computer-supported negotiation process in order to discuss the matter face-to-face.
6. Each of the above options can be accompanied by commenting on the choice.

This concept of negotiation was originally developed for situations in which two users of a computer system discuss whether a system feature should be implemented or not. This approach was intended to support "controllability" and "suitability for individualization" (cf. ISO 9234, Part 10) for groupware. Such negotiation can take place in multiple cycles of the proposer and the responder reacting to each other. Negotiation rules must be established to define how many negotiation cycles can take place, how much time is allowed to pass before a decision must be reached, what happens when a time limit is reached, etc. The goal of this negotiation mechanism is to get through routine cases of agreement, abstention, or simple modifications of proposals as quickly as possible and to determine which proposals require a more intensive communication process as efficiently as possible. This provides a common starting point from which cooperation can proceed. A disadvantage of this negotiation mechanism is that it was designed for just two people. If applied to several participants, the time period for arriving at a common starting point stretches out too much. Herrmann's (1995) original negotiation concept assumed that a modified item would not be worked on further until the negotiation process was complete. This might make sense in the case of a change of system functionality, but it seems unduly restrictive for modifications of information. By contrast, the approach of intertwining multiple perspectives into a common one has the advantage that participants can continue to work in their own perspective while awaiting the results of negotiations.

Related Work

Hypertext and Hypermedia.

Hypertext and hypermedia structures provide an important mechanism for supporting collaborative work with shared materials. To some extent, this is provided by the web itself, although many hypertext mechanisms have been explored that go beyond the web's simple model (Bieber, et al., 1997). The original perspectives mechanism of Stahl (1993) is a hypermedia implementation, based on

a node and link structure; relationships among contents in different perspectives are defined by links. Internal manipulation of nodes and links allows multiple perspectives to share large information sources without unnecessary duplication. This approach to “virtual copying” or “delta storage” is well-known in system software (e.g., Fitzgerald & Rastrich, 1986), but has not previously been used in CSCW hypermedia systems.

Database Views

Perspectives seem similar in spirit to database views. Both mechanisms specify a restricted presentation of information from a larger source. However, there are important differences. A database view specifies a selection of data characteristics (table columns) to display, whereas a perspective determines which items (table rows) are visible. Furthermore, the same data item can have different content in different perspectives. Perhaps the most significant difference is that perspectives can inherit their definition or content from other perspectives.

Context Mechanisms

The importance of perspectives in collaborative work has been recognized at a theoretical level by Boland (1992, 1995), Snodgrass & Coyne (1990) and others, primarily based on the hermeneutic tradition in philosophy: Heidegger and Gadamer (see Stahl, 1993). The application of virtual copying to perspectives was explored at Xerox PARC (Bobrow & Goldstein, 1980), but abandoned as too complicated for users at that time. A related mechanism of *transclusion* was proposed by Nelson (1981) for hypertext. McCall applied a similar approach for organizing hypertext information by domain and version in PHIDIAS (McCall, et al. 1990; Stahl, et al., 1992). Stahl extended McCall’s approach in HERMES (Stahl, 1993), implementing a hypertext version of virtual copying in a productivity tool for professional design teams. He subsequently adapted this mechanism in CIE, a collaborative information environment for supporting group maintenance of ISO 9000 documentation (Stahl, 1996).

Computer Supported Collaborative Learning

A number of software systems have been developed to support collaboration of research teams in schools, and CSCL (Koschmann, 1996) has become an important new research direction within CSCW. CSILE (Scardamalia & Bereiter, 1991), for instance, is a threaded discussion system customized to scaffold classroom research. Each entry in these systems is labeled with the contributor’s name and an indication of whether the item is intended as a hypothesis (my theory), observation (I saw), or request for information (I need to know). Systems like CoVis (Pea, 1993) and CaMILLE (Soloway, et al., 1994) also provide a workspace or notebook area for collecting research results. Rather than supporting negotiation through the system they rely on face-to-face interactions to make choices about what materials get entered into the shared repository. When such systems talk about perspectives, they mean different views of the same information, rather than views of information related to different individuals or groups.

Organizational Memories

By organizational memories we mean an approach to building a structured digital library of various forms of information that can be shared by community members through computer supported collaboration and communication mechanisms (e.g., Ackerman, 1994; Lindstaedt & Schneider, 1997). It would make sense to build support for the organizational memory into the CSCW system if this could be done without disrupting the collaboration or imposing burdensome new tasks. It is important that the organizational memory evolve a structure that facilitates future retrieval of relevant stored information. Intertwined perspectives can help to structure an organizational memory. For instance,

when a group of community members undertakes a new project they can create a new perspective and negotiate which items from existing perspectives should be included for use in the new project.

GroupLens

GROUPLENS (Resnick, et al., 1996) is a system that displays available information in accordance with individual or team preferences. Statistical analyses are used to automatically determine which members of a group are interested in similar topics. Items of information that are of interest to one member are then sent to other group members based on how highly correlated they are to the first member with respect to the topic of information. In this way, a shared perspective on selected items of information is created. This approach is particularly helpful for new information that appears. In Stahl's (1993) approach, information is automatically assigned to perspectives only in the sense that a perspective can inherit entire other perspectives and that when one makes a virtual copy of an individual node one can inherit the subnetwork of nodes related to that node. By relying entirely on automated mechanisms, GROUPLENS does not allow active selection or rejection of information based on user choice.

Conflict Management

The above approaches lack any computer supported negotiation mechanisms. Wulf (1997) adopted Herrmann's approach and developed it for conflict management in groupware. Wulf distinguishes three ways in which a groupware user can avoid or reduce the effects of another user's actions:

- a. the groupware lets the user suppress the influences of others' actions;
- b. if the influence cannot be limited, it is at least made comprehensible; or
- c. the action is subjected to negotiation.

However, we believe that it should always be possible for the users to react to each other as necessary, at least by commenting. Ideally, these reactions back and forth should take place with support from the same system that presents the content related to the responses.

Decision and Meeting Support

The clearest parallel to computer-supported negotiation is decision support and meeting support systems (Geibel, 1993). In these systems, one can respond to proposals from others by extending them with one's own proposals or amendments. One can also annotate the proposals. In more elaborate systems, such as those derived from ARGNOTER, (Stefik, et al., 1988, p. 345ff) one can classify one's annotations as pro or con the argument. Although pro and con arguments can be taken as votes for or against a proposal, they do not automatically have the same consequences. Rather, there are several systems that count votes for or against a proposal by, for instance, allowing one to assign a specified number of points among a number of alternative proposals. Those proposals that garner the most points are then considered the group's choice. Various conditions must be met to organize this kind of voting properly; Ephrati (1994) provides a nice overview and discussion of this. Sen, et al. (1997) describe an application of this for meeting scheduling. A problem with such voting procedures is that one may have to deal with a number of perspectives without ever really comprehending their content, particularly when no annotations or discussion are attached to the individual points. Our approach to intertwining perspectives and negotiation tries to connect each with the other to make the process more substantive.

Agent Systems

Approaches to negotiation can also be found implicitly in agent systems. Martial (1996) proposed such an approach for solving conflicts in resource allocation. In addition to the above noted reaction possibilities within negotiation processes, Martial distinguished between modifying a proposal and presenting a counter-proposal. We try to get away with ignoring this distinction. Bayer (1995)

developed an approach in which part of the negotiation effort is taken over by software agents, in order to relieve the user of some work under ordinary conditions.

The related work reviewed here always deals with just part of what we wish to address. On the other hand, this work contains valuable suggestions that should be remembered in developing the potential of intertwining perspectives and negotiation.

EXAMPLES OF THE NEED FOR INTERTWINING PERSPECTIVES AND NEGOTIATION

In this section we present three typical situations in which intertwining perspectives and negotiation might prove helpful. In each case we will see how one has to manage multiple viewpoints when no computer support is available for this. That will make clear some of the problems that CSCW perspective and negotiation mechanisms will have to address. These examples contain suggestions of how to proceed in developing our concept. We assume from the start that our concept will have to be implemented differently in every application domain.

Maintaining a Bibliography

Our first example involves managing a list of keywords used for classifying entries in a bibliographic database. Professional research teams frequently use a shared database of literature references. Often, part of the entry for a reference in such a database is only of interest to a particular person, while other parts are important for everyone or for subgroups. While the basic reference information is relatively standardized, the evaluation or annotation of a citation can involve different opinions. One can solve this problem by defining a different view for each user. This view displays evaluations or annotations in accordance with the user's choices. Of course, this form of customization makes little sense for information that is part of collaborative work.

The problem is even clearer when it comes to keywords used to categorize bibliographic entries. The purpose of keywords is not just to jolt one's memory, but also to help other people find references. Therefore, the catalog of keywords used within a team needs to be frequently reviewed and voted on. If one maintains a simple list of keywords, then the keywords can be combined or distinguished, deleted or replaced, and amended or revised. Whoever undertakes to make such a change must decide if the change is only relevant to him or if he should propose to the whole group that they accept the change and work with the new or altered keyword. If the change is proposed for the group then either the group manager can decide whether or not to adopt the proposed change or the group can vote round-robin. Both of these decision procedures have the disadvantage of taking considerable time. In general, substantive discussion is critical for building a shared understanding of the keyword. One solution would be to make the decision about a proposed change within the context of a group discussion. However, this procedure has the distinct drawback of spending too much time discussing trivialities.

It is important to develop a computer supported mechanism for efficiently sharing comments, voting, or discovering which changes really require extended discussion. A web-based prototype for this is currently under development at the University of Dortmund.

Developing a Glossary

For working in teams it is often useful to produce a list of the essential concepts in the domain. The list should include definitions of the terms; it may be helpful to structure the list hierarchically; and it is important to keep the list up-to-date. A glossary of terms can serve as a communication medium for a group, facilitating the creation of shared meanings and documenting the group language.

Glossary changes – additions, textual edits, reorganization – require a consensus of the whole group. Of course, trivial changes like the correction of typos should be able to proceed with minimal bureaucracy. Round-robin voting is not appropriate for these cases; it would be better to have one person authorized to determine whether a change can just be made or whether it requires a more intensive substantive discussion. At the other extreme, such as in cases of interdisciplinary research teams, it should be possible for someone looking up a term to tell if some group members hold a different definition of the term than others. A system of intertwining perspectives and negotiation should be able to support this by maintaining some negotiation history and by displaying different definitions in different perspectives. The DYNAGLOSS prototype under development at the University of Colorado maintains a history of glossary evolution.

A Student Research Project

This example is the basis for the WEBGUIDE system discussed below. In summer 1997 we developed an interface design for a web-based prototype that would integrate perspective and negotiation mechanisms to support collaborative learning in a middle school (6th grade) classroom. Students work in teams, but do much of their web research individually and asynchronously. For example, the teacher proposes the general theme, classical cultures of Latin America, and the students form teams to research the Aztecs, Incas, or Mayas. The team goal is to investigate different aspects of a culture through research on the web and then produce a team web site that presents the findings. A number of special web pages have been created in advance to start the students off with basic background information and with pointers to grade-appropriate web sites.

WEBGUIDE is a glorified web bookmark manager (cf. Keller et al., 1997) and electronic notebook application (cf. Torrance, 1995), enhanced with perspective and negotiation mechanisms as described below. Students can conduct web searches, collect, annotate, categorize, and organize bookmarks for sites they like. They can summarize or excerpt the web page contents – there is no need to copy the full contents because it is already easily available through the bookmark. Students are encouraged to use the facilities of WEBGUIDE to make the results of their research more self-explanatory for viewers by defining a hierarchy of headings or categories, arranging bookmarks under these, and adding concise summaries of the content or importance of the bookmarked sites.

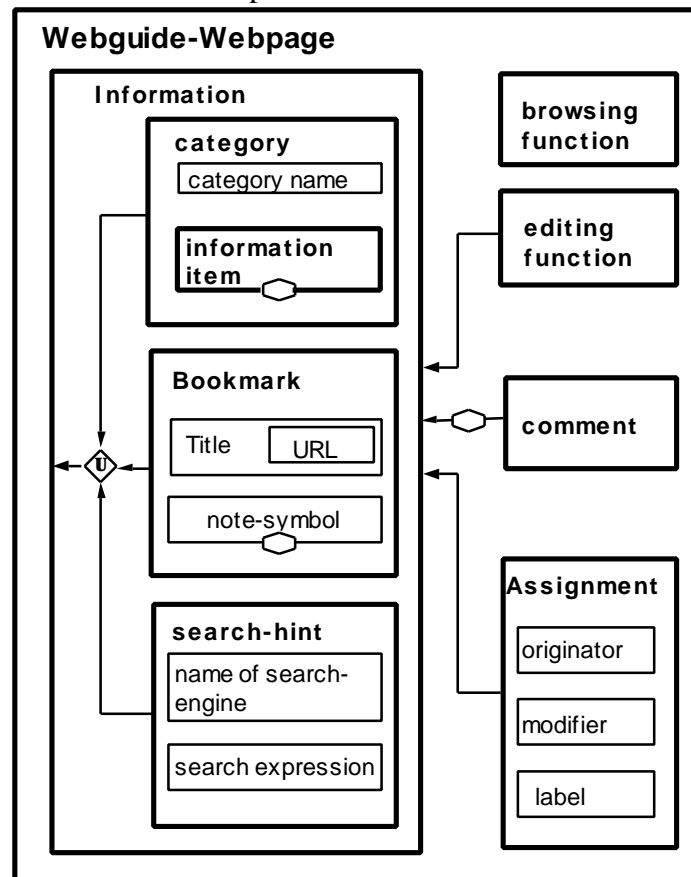
Most of the initial research work is done by students working on their own. When it comes time for team members to integrate their work, they have a problem. The organizing structures they have built are mostly incompatible. Twelve year old students are not yet experienced in negotiating equitable compromises, and many students see much of their work discarded in the process of merging the work of all team members. This results in poor motivation. Students need guidance in structuring a healthy, democratic discussion about how to merge individual accomplishments into a team product.

A system is needed that can support the merging of individual results. Such support should begin early and continue throughout the research process. It should structure and facilitate the decision-making process so that students can learn how to build a consensus. We have designed WEBGUIDE to do just this. The example of this student research project is well suited to illustrating the power of intertwining perspectives and negotiation: it shows the level of complexity that our approach can handle.

THE WEBGUIDE PROTOTYPE

System Goals

Students should each have their own perspective within which they can conduct, annotate, categorize and organize their own web discoveries. Their personal perspectives should inherit from a team perspective, so that their continuing work takes place within the context of what has already been agreed to. Each student should be able to view the work of other team members. Students should be able to adopt individual items from the work of other students into their own perspective, in order to start the collaboration and integration process. From early on, they should be able to make proposals for moving specific items from their personal perspective (or from the perspective of another) into the team perspective, which will eventually represent their team product, the integration of all their work. This requirement presupposes that information can be presented and collected in small pieces. This is also necessary for negotiating which pieces should be accepted, modified, or deleted. Figure 1 presents a schematic data structure of the information pieces that are relevant for WEBGUIDE.



[Figure 1. Data structure for a basic WEBGUIDE page.]

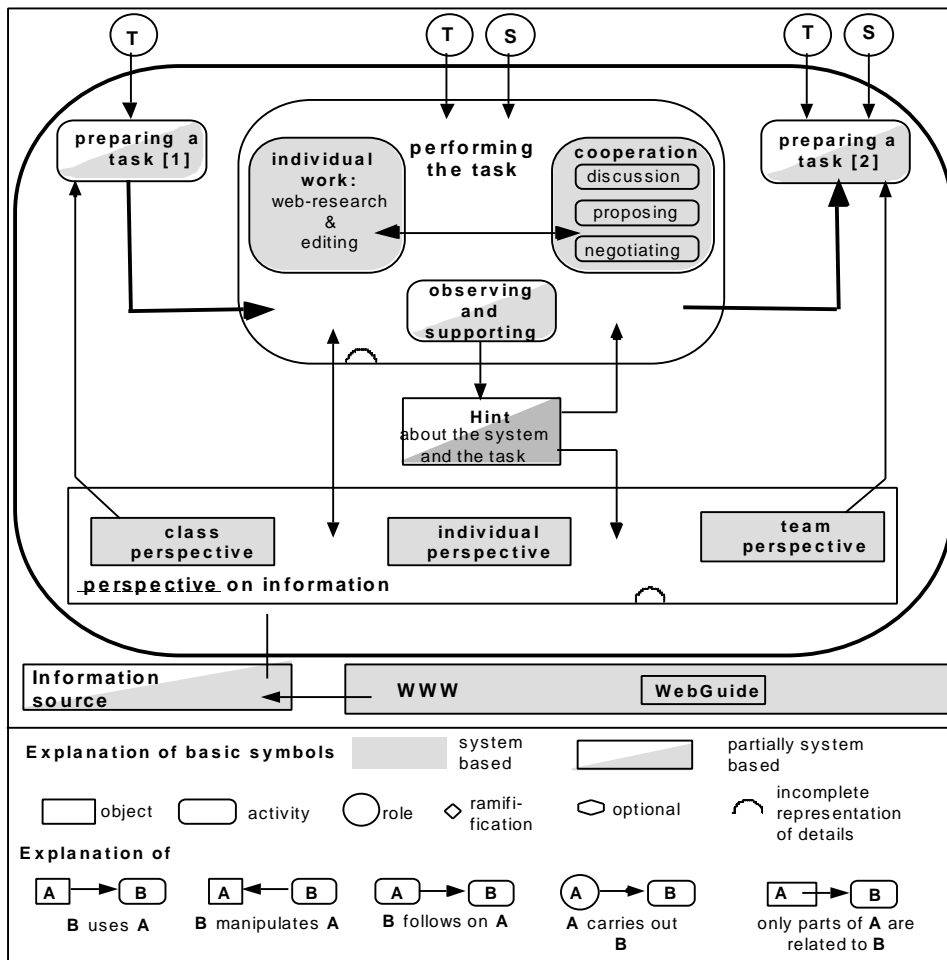
An important requirement of WEBGUIDE is that it serve as a digital notebook in which students can store the results of their research. The web pages of a student's personal perspective should not only contain clickable bookmarks and search queries, but also categories, comments and summaries authored by the student.

As indicated in Figure 1, a bookmark can have a title as well as the link to a URL. A note icon can optionally be attached as well (The hexagonal symbol indicates an option.) This icon represents a link to another web page that might contain an extract, summary or commentary to the bookmarked site, authored by the student and possibly incorporating information from readings or discussions. In addition to bookmarks, the WEBGUIDE page can contain web search queries for finding current sites on a given topic. WEBGUIDE is designed to help students learn to do web research and the sharing of successful query formulations is important for that. Finally, WEBGUIDE pages can contain categories for organizing the bookmarks and queries. These categories can initially be created without any bookmarks or queries as preparation for looking for relevant information. The categories can be structured hierarchically to create an information outline, as indicated by the recursive embedding of information item in Figure 1.

Comments can optionally be attached to any information item. Every item is tagged with the name of the person who created or last modified it. Items are also labeled with perspective information. Furthermore, there is a set of functions which can be used to edit the information items. Because of the hierarchical nature of items, something that appears as a unit of information that can be proposed for negotiation may actually consist of many parts, some of which appear differently in different perspectives. The possibility of information items having a complex but hidden internal structure is required for the intertwining of perspectives and negotiation. The internal complexity must be handled by the perspective and negotiation mechanisms behind the scenes of the WEBGUIDE interface.

A Use Scenario

To design software for collaborative learning means to design curriculum and classroom process as well. Computer support has to be matched with appropriate content on the web and with a constructivist pedagogy (Brown & Campione, 1994; Greeno, 1993; Scardamalia & Bereiter, 1991). The design of the WEBGUIDE interface and the perspective and negotiation mechanisms is accompanied by the design of informative web pages and of a use scenario.



[Figure 2. The basic model of the WEBGUIDE approach.]

We envision the teacher introducing the topic and the general goal. (The teacher's role is indicated by a "T" in Figure 2.) This can include a brief introduction to the Aztecs, Incas, and Mayas. Then the students divide into teams to research each of these civilizations and to report on them. The teacher can suggest themes such as science, religion, art, astronomy, agriculture, architecture for students to explore. Categories corresponding to these themes have already been defined within the so-called *class perspective*, which includes bookmarks of web pages and search engine queries that make good starting points for research.

Students ("S" in Figure 2) begin by copying information relevant to their research topic from the class perspective into their *personal perspective* and then starting to explore. They use their personal perspectives as workspaces or research notebooks for their individual work. As they make their own contributions, they are encouraged by the teacher or by face-to-face interactions with team mates to look at what others have discovered.

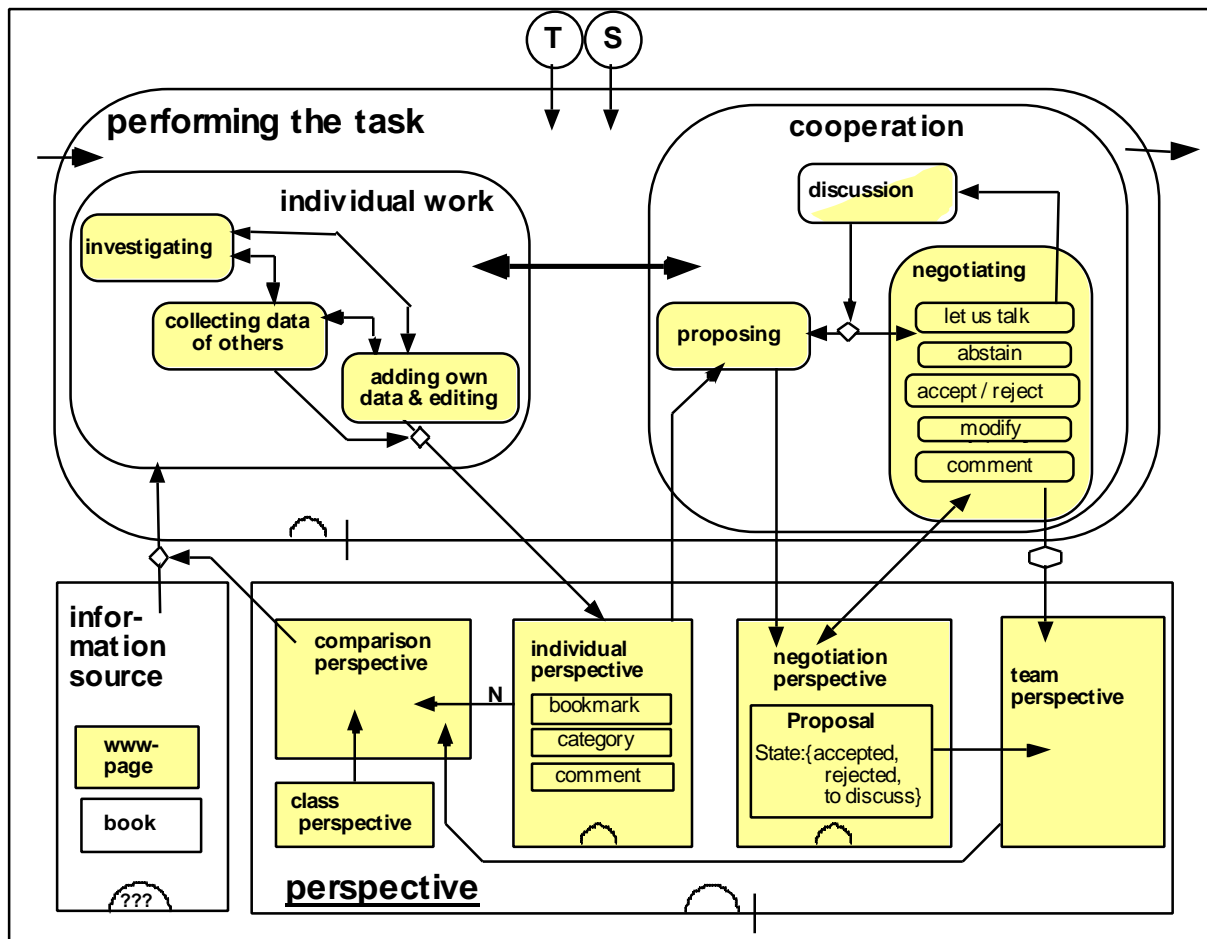
Collaboration proceeds asynchronously through WEBGUIDE, although team members can also meet together in the classroom and discuss their ideas and progress. A student can copy another's information item, modify it, and propose it or her own item for adoption by the team. Proposals can be made at any time; they then appear in the *negotiation perspective* of the team, a temporary halfway house on the way to being accepted into the *team perspective*. Students can work on their ideas in their

personal perspectives as long as they like and then decide to propose large or small items. They soon learn that well-developed information is more likely to be approved by their team mates than ideas that are poorly researched, structured, or annotated. Once an idea is proposed, team members have an opportunity to negotiate its acceptance and to subject both the proposal and its negotiation to computer-supported discussion. During the entire research project the teacher and the students can reflect upon events and provide hints for improving the process.

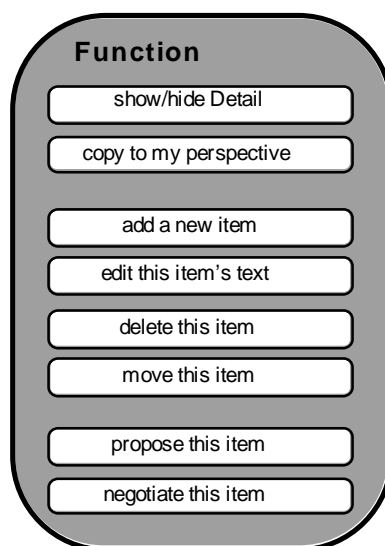
The project ends with each team producing an organized web site about one of the civilizations. These web sites can be used by members of the other teams to learn about the civilizations that they did not personally research. The sites can also provide a basis for additional class projects, like narrative reports. Finally, this year's research products can be used to create next year's class perspective starting point, so new researchers can pick up where the previous generation left off – within a web information space that will have evolved substantially in the meantime.

Support of Individual Work

Figure 3 shows how the task is performed in detail. Students research using sources available to them: the web, books, encyclopedia, CD-ROM, discussions, or other sources that we cannot anticipate (indicated by ??? in the figure). Students can review the contents of the class perspective, their team perspective and the personal perspectives of their team mates. All of these contents are collected in the *comparison perspective*, where they are color-coded and labeled by their perspective of origin. Students extract from the research those items which are of interest to them. Then the students organize and develop the data they have collected by categorizing, summarizing, labeling, or annotating. The three stages – investigating, collecting, editing – can be repeated as many times as necessary. To support these steps of the individual work, WEBGUIDE must provide a menu of functionality for each information item. The complete set of options currently planned is shown in Figure 4. Of course, not all of these options are meaningful for every type of item in every perspective.



[Figure 3. Details of the task performance and perspectives.]



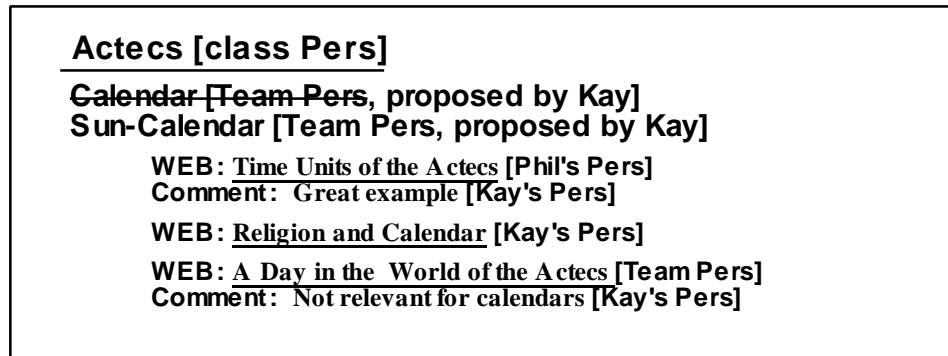
[Figure 4. Menu of options for information items.]

The menu item `show/hide detail` applies to most items within any perspective. It means that all items below the given item in the information hierarchy should be displayed/hidden. For instance, if one selects a bookmark and chooses the menu item `show detail`, then any notes or comments associated with that bookmark will appear below it on the web page. Conversely, if one chooses `hide detail` for a category on a page then all information under the category will be hidden. This should facilitate understanding and reorganizing the structure of WEBGUIDE pages.

Within the comparison perspective, editing is not allowed. One is only permitted to copy items into one's own perspective. When one selects this option, a dialog box offers the choice of copying just a single item, such as the name of a category, or also copying all the items under that category as a group.

Various editing functions are available within one's personal perspective. One can move an item to another position, modify text, or delete an item. Each of these functions involves a dialog box that accepts further specifications.

Figure 5 shows an excerpt from Kay's personal perspective. Here it is clear that Kay always sees what she has copied from other perspectives into her own perspective as well as what she has added to that or what she has proposed for deletion.



[Figure 5. Information items in Kay's perspective.]

Support of Proposals for Negotiation

Each student can make proposals for the team perspective from the `propose` option within his or her personal perspective. This is how new items get introduced into the team perspective. A student can also propose an item from someone else's perspective by locating it in the comparison perspective and copying it into her own perspective. If someone wishes to modify an item that is already in the team perspective, she must copy it into her own perspective, make the modifications there, and propose the modified item. The personal perspective always provides the workspace for individual efforts.

As shown with the category heading in Figure 5, when an item copied from the team perspective is modified, the change remains visible by having both the official team version and the modified personal version displayed and labeled with perspective indicators. This somewhat awkward approach is useful because it allows the modified item to be used by the proposer and by others while the results of negotiation are being determined.

It should be possible when proposing – just as with copying – to treat a set of related items in one step. It is also important to be able to treat a set of proposed changes together even if they are not related. For example, if a student deletes a bookmark at one spot in order to replace it with a better,

richer bookmark elsewhere, then the deletion and the replacement should both be proposed and negotiated together. Of course, students should be discouraged from putting too many items together.

When a student selects the proposal function, a dialog box opens. The student can decide whether the new proposal item should be combined with a previous or future proposal. The proposer also sees a list of all the other students who will be involved in the negotiation of the item. The determination of who should be involved is a matter for installation settings that define a local negotiation policy. Ideally, these settings should be system parameters of WEBGUIDE, so they can be easily varied by teachers or other user communities. For example, one might want to establish a rule that all new items must be negotiated by all team members – or alternatively that they do not require negotiation at all – while modified items require just those people to participate who either originally created the item or subsequently modified it. Another plausible rule would be to accept all annotations without negotiation.

As soon as an item is proposed, it appears in the negotiation perspective. Through perspective inheritance, it also appears in the team perspective and in the personal perspective of all team members, labeled as *Proposed by name-of-proposer*. At first no editing of the proposal is allowed in the negotiation perspective. Rather, there are two windows. One displays the proposed item or items within the context they would have in the team perspective once accepted. This window includes buttons corresponding to the negotiation options: *Accept*, *Reject*, *Abstain*, and *Let's talk*. The second window displays the results of negotiation decisions already made on the proposal and the commentary of team members concerning these decisions. (See Figure 6.)

Name	Reaction	Statistic
Que:	acceptance of Kay's proposal	1 of 3
	<i>Comment:</i> I think this makes sense	
Bea	rejection of Kay's proposal and modification of Kay's proposal	1 of 1
Phil	rejection of Bea's modification	1 of 1
Jay	rejection of Bea's modification	1 of 2
	<i>Comment:</i> I prefer the original	
Bea	comment on Jay's comment	1 of 1
	<i>Comment:</i> why do you prefer it?	

[Figure 6. Example of the WEBGUIDE display of negotiation responses and comments by four students.]

Several negotiation responses to the proposal are possible at this point, as shown in Figure 3. A negotiator can indicate that she abstains, that she does not care to participate in the negotiation. Alternatively, she could indicate with *Let's talk* that she would like to discuss the proposal face-to-face in the team. In the later case, the label on the proposed item would change to *Proposed by name-of-proposer, Let's talk*. In addition, an automatically maintained agenda of points for group discussion would be extended to include this proposal.

Of course, the primary options are to *Accept* or *Reject* the proposal. It should be noted that a proposal can have been modified by other group members so that there may be several versions of the

same proposal. If `Accept` is selected for one alternative, then all the others are assigned `Reject` and the negotiation is over for that student. If `Reject` is selected, then the next version of the proposal is displayed. When several versions are available, a student can either accept precisely one or reject them all.

After making a negotiation decision, a student can comment on her response. All students who view the negotiation perspective after that can see her response with her comment. Although it may not be sensible in a negotiation situation involving several participants to allow cycles of responses to responses because the negotiation process would quickly become too confusing, WEBGUIDE does allow students (and teachers) to comment on all actions, including comments on comments. Even after a student has completed her voting on a proposal she can comment on other people's choices or change her vote. Figure 6 shows how a set of responses and comments might appear.

The procedure for modifying a proposal is a bit involved. Once a student has rejected all the existing versions of a proposal, then she can modify the proposal in her personal perspective and propose her modified version. Then the new version will be automatically integrated into the negotiation process of the original proposal. The label of the proposed item will be altered to read, `Proposed by name-of-first-proposer altered by name-of-second-proposer`. Students who have already voted can see this new label and can decide if they want to return to the negotiation perspective and reconsider their vote on this proposal. It might also make sense to have a more intrusive mechanism to alert people to newly proposed versions by event mechanisms. The design decision to restrict modifications this way in the negotiation process results in a simplification of the process. A student can only edit a proposed item under the conditions in which this item appears in the student's personal perspective. It is only possible to edit the original proposal, not proposals that already have the label `altered`. Otherwise, things would become too confusing. While a proposal can be rejected by its original proposer when she prefers an alternative version, it cannot be recalled because that would create an asymmetry between the proposer and other participants.

The negotiation process for a proposal cannot exceed a time limit, determined by the negotiation policy parameters. At the end of the time period, the system determines whether the proposal or a modified version is accepted or rejected. Again, installation parameters determine what kind of majority is required: 2/3 of those voting, majority of those eligible, simple majority, etc. If the results are indecisive, the proposal will be labeled `proposed for talk` and added to the discussion agenda. Then students will have to get together in the classroom and decide what to do about the proposal. When matters are decided in group meetings, someone with a special password can enter changes directly in the team perspective, bypassing the computer-supported negotiation process.

CONCLUSION

It is clear that the idea of intertwining perspectives and negotiation requires the information content to be divided into relatively small items that can be easily edited, rearranged, and negotiated. This is quite different from collaborative editing of lengthy documents. Moreover, there are a number of software ergonomic issues involved in a complex web application. It is hard to implement direct manipulation with HTML because new web pages have to be generated by a distant server when contents change.

We now have an interface design mock-up that shows several typical perspectives (see <http://www.cs.colorado.edu/~gerry/WebGuide/webguide.htm>). Although the menu functionality for the perspectives and negotiation is not implemented, the dialogs showing planned options can be seen in

Kay's personal perspective and in the Aztec negotiation perspective. One can see how illustrative contents show up under different perspectives and observe certain stages of negotiation.

WEBGUIDE will be further implemented in 1998. Initial evaluation of some of its concepts will be evaluated in a middle school classroom in Boulder, Colorado. Eventually we would like to see how different features are used in practice. For instance: To what extent do students group related proposals together? How much do students comment on their negotiation decisions or on those of others? Can students handle the process of modifying proposals? We would also like to explore different negotiation policy parameters: What happens to proposals that just one or two students slate for group discussion? What time limits, voting methods, negotiation participation rules are meaningful and effective?

Certainly, the full concept of negotiation has to be drastically simplified if middle school students are to succeed with it. The textual description of intertwining perspectives and negotiation in this paper is complex indeed, even for adult computer scientists to think about. We assume that the ideas can be made more intuitive in a CSCW system with a carefully designed interface. While we want the structure of perspectival views and negotiation options to be made explicit to students as they begin to develop critical thinking skills, we hope that much of the mechanism for implementing them can be hidden from view behind intuitive interface metaphors. We will have to introduce the functionality of WEBGUIDE into the classroom gradually, working closely with teachers and students and creating a curriculum and classroom process to complement it. This process will reveal opportunities for simplifying the system and making it more useful and usable. If we are successful in creating an effective WEBGUIDE system for twelve year olds, we will have learned much that is relevant to developing usable mechanisms for intertwining perspectives and negotiation in other domains of collaborative work and collaborative learning.

REFERENCES

- Ackerman, M. S. (1994) Augmenting the organizational memory: A field study of Answer Garden. In: *The Conference on Computer Supported Collaborative Work (CSCW'94)*, New York: ACM. Pp. 243-252.
- Bayer, E. (1995) *Flexibilität in E-Mail Systemen: Ein Modell für Aushandelbarkeit*. Diplomarbeit at the University of Bonn.
- Bieber, M., Vitali, F., Ashman, H., Balasubramanian, V., Oinas-Kukkonen, H. (1997) Fourth generation hypermedia: Some missing links for the World Wide Web. *International Journal of Human-Computer Studies*. Special issue on HCI & the Web.
- Bobrow, D. & Goldstein, I. (1980) An experimental description-based programming environment: Four reports. Technical Report CSL-81-3. Palo Alto, CA: Xerox Palo Alto Research Center.
- Boland, R. J. & Tenkasi, R. V. (1995) Perspective making and perspective taking in communities of knowing. *Organization Science*. Vol. 6, No. 4, pp. 350-372.
- Boland, R. J., Maheshwari, A. K., Te'eni, D., & Tenkasi, R. V. (1992) Sharing perspectives in distributed decision making. In: *The Conference on Computer Supported Collaborative Work (CSCW'92)*. New York: ACM. Pp. 306-313.
- Brown, A. & Campione, J. (1994) Guided discovery in a community of learners. In: McGilly, K. (Ed.) (1994) *Classroom Lessons: Integrating Cognitive Theory and Classroom Practice*. Cambridge, MA: MIT Press. Pp. 229-270.
- Ephrati, E., Zlotkin, G., Rosenschein, J. (1994) Meet your destiny: A non-manipulable meeting scheduler. In: *The Conference on Computer Supported Collaborative Work (CSCW'94)*. New York: ACM. pp. 359-371.

- Fitzgerald F., Rashid R. (1986) The integration of virtual memory management and interprocess communication in accent. *ACM Transactions on Computer Systems*, 4, (2), 147.
- Floyd, C. (1992) Software development and reality construction. In: Floyd, C., Züllinghoven, H., Budde, R., Keil-Slawik, R. (1992) *Software Development and Reality Construction*. Berlin, Heidelberg, New York: Springer-Verlag. Pp. 86-100.
- Geibel, R. (1993) *Computergestützte Gruppenarbeit. Die Förderung von Gruppenentscheidungen durch Group Decision Support System*. Stuttgart: M&P Verlag.
- Greeno, J. (1993) For research to reform education and cognitive science. In: Penner, Batsche, Knoff, Nelson (1993) *The Challenge in Mathematics and Science Education: Psychology's Response*. Washington, DC: APA. Pp. 153-193.
- Habermas, J. (1981) *Theorie des kommunikativen Handelns. Band 1. Handlungsrationalität und gesellschaftliche Rationalisierung*. Frankfurt: Suhrkamp.
- Herrmann, T. (1994) Grundsätze ergonomischer Gestaltung von Groupware. In: Hartmann, A., Herrmann, T., Rohde, M., Wulf, V. (Hrsg.) (1994) *Menschengerechte Groupware - Software-ergonomische Gestaltung und partizipative Umsetzung*. Stuttgart: Teubner Verlag. S. 65-107.
- Herrmann, T. (1995) Workflow management systems: Ensuring organizational flexibility by possibilities of adaptation and negotiation. In: Comstock, N. et al. (Ed.): *COOCS'95. Conference on Organizational Computing Systems*. August 13-16, 1995. Milpitas, California. New York: acm-press. New York: ACM. S. 83 - 95.
- Keller, R., Wolfe, S., Chen, J., Rabinowitz, J., Mathe, N. (1997) A bookmarking service for organizing and sharing URLs. In: *Computer Networks and ISDN Systems*. 29, Pp. 1103-1114.
- Koschmann, T. (ed.) (1996) *CSCL: Theory and Practice*. New Jersey: Lawrence Erlbaum Associates.
- Lindstaedt, S., Schneider, K. (1997) Bridging the gap between face-to-face communication and long-term collaboration. In: Hayne, S.; Pronz, W. (eds.) *Proceedings of the International ACM SigGroup Conference on Supporting Group Work. The Integration Challenge*. New York: ACM. S. 331-340.
- Martial, F. von (1996) A conversation model for resolving conflicts among distributed office activities. In: *SIGOIS 2,3*. Pp. 99 - 108.
- McCall, R., Bennett, P., d'Oronzio, P., Ostwald, J., Shipman, F., Wallace, N. (1990) Phidias: Integrating CAD graphics into dynamic hypertext. *Proceedings of ECHT'90*. Pp. 152-165.
- Nelson, T. H. (1981) *Literary Machines*. Sausalito, CA: Mindful Press.
- Pea, R. (1993) The collaborative visualization project. *Comm. ACM*. Vol. 36, No. 5, pp. 60-63.
- Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P. (1996) GroupLens: An open architecture for collaborative filtering of netnews. In: *The Conference on Computer Supported Collaborative Work (CSCW'94)*. NY: ACM Press. Pp. 175 -186.
- Scardemalia, M. & Bereiter, C. (1991) Higher levels of agency for children in knowledge building: A challenge for the design of new knowledge media. *Journal of the Learning Sciences*, 1, pp. 37-68.
- Sen, S., Haynes, T., Arora, N. (1997) Satisfying user preferences while negotiating meetings. In: *Int. J. Human-Computer Studies*. Pp. 407-427.
- Soloway, E., Guzdial, M., Hay, K. (1994) Learner-centered design: The next challenge for HCI. *ACM Interactions*. Vol. 1, No. 2, pp. 36-48.
- Stahl, G. (1996) Personalizing the web. Available at <http://www.cs.colorado.edu/~germy/HomePage/Publications/WWW6/PAPER82.html>.
- Stahl, G. (1993) *Interpretation in Design: The Problem of Tacit and Explicit Understanding in Computer Support of Cooperative Design*. Unpublished Ph.D. Dissertation. Department of Computer Science. University of Colorado at Boulder.

- Stefik, M., Foster, D., Bobrow, D.G., Kahn, K., Lanning, S., Suchman, L. (1988) Beyond the chalkboard: Computer support for collaboration and problem solving in meetings. In: Greif, I. (Ed.) (1988) *Computer-Supported Cooperative Work*. Pp. 335-366.
- Torrance, M. (1995) Active notebook: A personal and group productivity tool for managing information. In: *Proc. AAAI Fall Symposium on Artificial Intelligence in Knowledge Navigation and Retrieval*. Cambridge, MA. Pp. 131-135. Available at <http://www.ai.mit.edu/people/torrance/papers/aaai-fall-95.ps>.
- Wulf, V. (1997) *Konfliktmanagement bei Groupware*. Braunschweig, Wiesbaden: Vieweg.

ACKNOWLEDGEMENTS

The WEBGUIDE prototype and this paper resulted from the intertwining of two rather different perspectives on software development: a German theoretical inclination toward abstract modeling and an American pragmatic preference for mock-up. Through delicate negotiation, the authors arrived at a mutual understanding of the complex of subtle design and cultural issues that confronted our collaborative research.

The research reported here was undertaken during a six month visit by Herrmann to Boulder. Stahl's work was supported in part by grants from ARPA (N66001-94-C-6038), NSF (IRI-9711951) and the McDonnell Foundation.