UNC Collaboratory Project: Overview

TR90-042

November, 1990

John B. Smith
F. Donelson Smith
Peter Calingaert
John R.Hayes
Dorothy Holland
Kevin Jeffay
Marcy Lansman

The University of North Carolina at Chapel Hill
Department of Computer Science
CB#3175, Sitterson Hall
Chapel Hill, NC 27599-3175

A TextLab Report

This research was supported by NSF contract #IRI-9015443, IBM SUR-Agreement #866 and Office of Naval Research contract# N00014-86-K-0680.

UNC is an Equal Opportunity/Affirmative Action Institution.

Contents

Summary

This report describes a new project in computer-supported collaboration for scientific and engineering professionals, made possible by support from NSF, IBM, and ONR. For over twenty years the Department of Computer Science at the University of North Carolina-Chapel Hill has successfully conducted research on systems for intelligence amplification. Our longest continuing project of this kind has been the application of interactive computer graphics to assist a biochemist elucidating the structure of a complex protein molecule. A more recent project has developed a hypertext-based system designed to support authors of scientific, technical, and other expository documents.

Crucial to our success has been the selection of driving problems whose solutions have been of significance not merely to us as tool builders, but also to professionals in other disciplines. We have used the systems we built to study their human users engaged in complex intellectual tasks. As computers and communication have become more intertwined, with great strides in the development of distributed systems, and with the growing necessity for "team science", we believe the time is right to select a new driving problem -- support for multiple professionals working in collaboration.

We have assembled a multi-disciplinary team of researchers (Anthropology, Psychology, and Computer Science) that will carry out this research program. This team has ready access to a rich multi-media communications infrastructure that begins within our building and extends throughout the state of North Carolina.

The central focus of our research is observing groups doing real collaborative work using systems and communications media explicitly designed to aid in their tasks. We expect results that will expand our understanding of how people collaborate and how to design systems that augment collaborative activities. To this end, our project has five interdependent components:

- a theoretical foundation for observing and understanding the social and cognitive aspects of group collaborations
- tools for rapid prototyping and reconfiguration of application environments for use by working groups, and multi-media communications to support multi-person interactions
- protocol analysis tools to record and study how individuals and groups interact through the networked computer environment
- application testbed systems (generic and domain-specific) that can be used by groups engaged in real work
- group studies and experiments to test system, social, and cognitive hypotheses.

The purpose of this report is to describe our current view of the project. However, we expect our understanding of this work and our agenda for future research to evolve with the project. Consequently, we anticipate issuing updated reports from time to time to record our changing views of the project.

Introduction

Concept

Why do we collaborate?

The goal of this project is to amplify the capabilities of scientific and technical professionals engaged in complex tasks and, thereby, increase their productivity and the pace of scientific and engineering achievements. For many scientific and engineering tasks, most of the intellectual energy is expended on creating conceptual, rather than physical, constructs. Brooks has identified the building of complex conceptual structures as the *essential* difficulty inherent in large software engineering projects [Brooks 1987]. Creating such conceptual structures is common to many other tasks, such as designing scientific experiments or computer systems, developing strategic and tactical plans in business or the military, and creating government policies. Large-team scientific research programs (e.g., Global Change, Human Genome) require project management plans which can also be viewed as complex conceptual structures for coordinating, scheduling and controlling activities.

In our desire to tackle major problems, we face a curious dilemma. Had we mind enough and time, we would probably get maximum coherence and integrity in an intellectual endeavor if the project were done by a single exceptional individual. However, real-world constraints often foreclose this possibility. No individual has all of the expertise, the information, the diversity of point of view, or the time to carry out such projects. Instead, we must be synthesize these resources in a group. Frequently, however, the right people are not all in the same location. Physical separations, even relatively small ones, can seriously inhibit interactions among people, e.g., among departments scattered across a large university campus or across a corporation.

How does collaboration work?

We know very little about the complex processes involved when people collaborate. While we know something about how individuals interact with one another within organizations and groups, we know almost nothing about the cognitive processes that common sense tells us must be involved in reaching shared "knowledge" or "understanding" and jointly building a complex structure of ideas. A fundamental question, then, is how do groups work in collaboration to build large coherent conceptual structures. (We view "collaboration" as a much stronger notion than "cooperation" or "coordination" because of the shared intellectual activity that is implied.)

The central task collaborators confront is to meld the individuals' expertise in order to, first, build a coherent conceptual structure and, then, to express it fully in words, drawings, programs, etc. Team members -- particularly when they come from different disciplines and areas of expertise -- must first build a common understanding of key concepts and terms, often translating unfamiliar jargon from other disciplines into terms and concepts they understand. Through (perhaps lengthy) dialogs, the group must reach consensus on a common framework. Individuals construct their contributions within this larger framework for which they are likely to have only a partial understanding. The whole ensemble of shared and individual conceptual structures evolves as the collaboration proceeds. The key to studying this process, we postulate, is to map the evolving relation between a common structure of shared ideas ("group information") and the distinct substructures of ideas and knowledge held by individual team members ("private knowledge").

A key dimension of collaboration is the duration of time over which collaborations take place. Work on complex problems may occupy groups for months or years. One of the authors (F. D. Smith) has participated in creating and maintaining a complex conceptual structure (the formal specification of IBM's Systems Network Architecture) that has evolved over 18 years and is the product of collaboration by several hundred people. Obviously, the cast of participants changes greatly over the years in long-term collaborations such as this. One critical issue, then, is how to provide effective access to shared, evolving ideas to an ever-changing group of participants, most of whom may work semi-independently and may be separated geographically. We refer to this notion as "asynchronous" collaboration.

Whereas collaborators may spend a great deal of time working independently, many activities require periods of direct interaction -- exploring, questioning, proposing, reviewing, negotiating, agreeing. We refer to such episodes of direct interactions among members of a group as "synchronous" collaboration. Synchronous collaborations provide the fine-grained transfer of information within the context of a long-term asynchronous collaboration.

What tools and methods can help collaborators?

Why might computers be good tools for augmenting collaboration? Our view is that people working together usually share some concrete, often complex, conceptual artifact that is expressed in words, drawings, images, etc. These artifacts take many forms: a book manuscript, a patient's medical records, a musical score, a manual of procurement procedures, or a computer system specification. Much of collaboration is concerned with creating the conceptual artifact, reaching a shared understanding of it, and agreeing on changes to it as it evolves over time. Since computers are often the tools of choice for representing, storing, and manipulating conceptual artifacts, the notion of extending the same tools for augmenting collaborations is appealing. Consequently, we believe that a shared hypermedia database is the most promising tool for maintaining the artifacts required in a collaboration -- it is the foundation of our support for asynchronous collaboration.

We believe that episodes of synchronous collaboration will be most productive if all participants simultaneously have full access to their shared computer-stored materials. Furthermore, it should be convenient to begin and sustain these interactions without elaborate scheduling or, better yet, without leaving one's office. Consequently, we believe that shared visual workspaces implemented on graphics-based workstations and augmented by multi-media communications (audio and video) are the most promising technologies for supporting synchronous collaboration.

We know little about characterizing and augmenting effective, efficient collaborative strategies. We have few tools or methods for studying collaborative work in depth. Consequently, we believe fundamental research must be done in methodology before collaborative studies can advance appreciably.

Our computer systems are awkward and unimaginative in their support of this application. Consider just the task of designing software -- many tools (e.g., CASE tools) are available to aid in managing the myriad of details inherent in the realization of a design, but none explicitly support the conceptual aspects of this tasks. Recent research applying hypertext technology to the software process is promising, but these systems are experimental; they are also points within a very large design space [see, for example, Scacchi, 1988; Nagl, 1986; Delisle & Schwartz, 1986; Taylor, et. al, 1987; Bigelow & Riley, 1987; Biggerstaff, et. al, 1987]. Consequently, we believe research should be directed toward developing comprehensive systems to support multiple people working in collaboration on large, complex problems, especially within distributed computing environments.

Research Strategy

Our strategy for the project we are beginning includes the following elements:

- a theory of collaboration
- tools and communications infrastructure for a distributed environment
- an application system to serve as a testbed
- tools for studying user interactions with the system
- experimental studies of working groups.

In our strategy, all of these elements reinforce one another: theory guides system design, the system supports experimental studies, results test and inform theory, etc. -- not as a linear progression but through continuous interaction.

We have focused our research on the following questions:

- *How do collaborators go about building complex conceptual artifacts?*
- *What computer tools can we provide that will help them?*
- *How can we tell for sure that we are really helping them?*

We must limit our research to a part of this very large field. First, we will focus on groups of scientific and technical professionals. Second, we will initially study collaboration in a limited set of application domains, with particular emphasis on the design of software systems. We believe, however, that our results will apply to scientific and engineering collaborations for many other tasks. Third, we will limit our efforts to groups working within a distributed computing environment of professional workstations linked by a multi-media communications network. Although the groups may occasionally meet in the same room, primarily they work on collaborative projects from their respective offices, interacting with one another through shared visual workspaces and live audio/video. We are not addressing groups working in the same room using some form of electronic whiteboard.

Our research strategy anticipates advances in networks (notably gigabit speeds and capabilities for all-digital audio and compressed video) that will make multiple media types widely available. We do NOT propose to do research in communications or network technologies, but rather we will investigate working environments for collaboration that can make effective use of this technical resource. While we don't expect network communication to replace face-to-face encounters, better support systems should help make distributed collaboration an effective and efficient alternative for many activities.

We are not attempting initially to characterize large groups, but we are being careful that our results and systems will scale well, at least to groups of 100 or more. One focus is to characterize the essential requirements (e.g., performance, media types) that must be satisfied in any scaling-up.

This project builds on a previous program of research we have carried out on hypertext environments for writers, research in systems for synchronous collaboration (including shared visual workspaces and operating system support for groups), and new research in hypermedia systems for software design. Thus, each component rests on a substantial basis of work accomplished, and each is a natural extension of an existing resource. That work is briefly reviewed, below.

Expected Results

Our project strategy is implemented in five inter-related activities that will produce the following results.

Theoretical foundations. To understand how groups work together, we are developing a theory of collective cognition and collaborative work that focuses on the conceptual artifacts that groups build over the duration of a collaboration. These artifacts may be tangible, in the form of design notes, diagrams, minutes of meetings, drafts of documents, etc. But they may also be intangible, in the form of a shared body of knowledge or a common understanding of the goals for a project. Focussing on these products provides a basis from which to observe the actions of a group, the processes operating within the group, and the goals and constraints that guide their behavior. From a preliminary version of this theory, we will define a set of guidelines to be used by social scientists to observe working groups engaged in complex conceptual tasks. Their insights, in turn, will be used to refine and validate the theory.

System-building tools and communication infrastructure. The foundations for tools to create application testbeds already exist in our project -- earlier work on MoDE (a user-interface management system (UIMS) for creating graphical user interfaces) [Shan 1989, 1990], shared visual workspaces and operating system support for groups [Abdel-Wahab et al 1988, Guan 1989], and hypertext environments [Smith et al 1987, 1989]. New research is required to extend these tools to multi-user environments tailored specifically for group collaborative activities. They will also be useful models for constructing future testbed systems in many generic and other application-specific domains. We will conduct our experimental investigations using the multimedia communications facilities available in Sitterson Hall (the modern facility housing the UNC Computer Science Department) and on the state-wide network operated by the Microelectronics Center of North Carolina (MCNC). This will result in new understanding about the utility of various media (graphics, voice, video) in collaborations.

Application Testbed System. We are developing a collaboration support system that augments groups attempting to fashion computer-based artifacts representing complex conceptual structures. For our purposes, studying groups collaborating in software design is a natural first step to prove that our theories, tools, and methods work. We note that nurturing collaboration in the software design process has been proposed as part of a research agenda in software engineering prepared for the Computer Science and Technology Board [CSTB, 1990]. It should be pointed out that we are NOT proposing research in software engineering processes, methodology, or tools (e.g. CASE). We focus only on supporting a group as it conceives and expresses a design; we will only provide access to standard existing tools (compilers, debuggers, make, etc.) needed for realizing and testing programs. While our system will have a strong emphasis on software design, most components for creating, browsing, and linking information will not be application specific and can be used by groups (including our own) working in a number of domains. This system will represent, at the end of the project, a model design for a collaborative environment. To meet this objective, the system must be robust and offer reasonable performance. It should not be viewed, however, as a product for general use, but as a reliable, efficient prototype.

Group Protocol Analysis Tools. To help answer the third question -- how do we know our theory and systems are helpful -- we will study users' behavior while they work with our systems. To aid this study, we will develop new protocol analysis tools and methods for analyzing group activities when using computer tools. Like the application tools, these can be built by extending our prior work on automatically recording transcripts, or protocols, that describe a user's interactions with an application system. We use these transcripts to replay a session -- in real time, sped up, or manually stepped through. We have also built a cognitive grammar in the form of an expert system to analyze protocols, and we have developed several display and analysis tools. We will extend these tools to record, analyze, and display protocols for groups of individuals working together within a networked computer environment. This will require new techniques for integrating protocols from multiple sources, a facility for replaying group sessions, grammars that can parse both individual and group protocols, and additional methods for analyzing and displaying composite results. These tools may be a valuable resource for other researchers, as well.

Studies of Collaboration. At the heart of our project is an attempt to understand collaboration on complex tasks. How do people collaborate with and without computer support? What aspects of collaboration might be augmented by new computer and communication technologies? How can we determine the contribution of computer augmentation to collaborative intellectual achievement? To begin answering these questions, we are conducting a series of interviews with individuals who have worked in group collaboration projects. We will soon begin observing groups, first, working "in the natural" without our computer system, then, artificially constituted groups carrying out assigned tasks, and, later, groups using our system to do actual work. Some of these groups will be constituted so that not all members are in the same location. Consequently, we will also investigate whether additional media for communicating can enhance the quality of collaboration for geographically distributed groups. If, without leaving our offices, we could participate in a collaboration using shared visual workspaces and see (as well as hear) our colleagues, would the character of our interactions be enhanced? Does the addition of sight and sound make the process more appealing (and, hence, more frequent and effective)? Should voice and video be integrated with the workstation to allow more flexible control of the collaboration software? Our experimental studies of groups will be designed to understand social, cognitive, and technological aspects of collaboration.

We believe this is foundation work that will lead to better support systems, better tools and methods for studying the behavior of groups, and to a basic understanding of collaborative work. While the project addresses basic issues, it builds on work accomplished over the past four years that can be extended in natural and logical ways. We do not underestimate the difficulty of doing so. But understanding and supporting collaborative work is important. We are eager to help accelerate the pace of this research.

In the remainder of this report, we first review background work we have done that will support the project described here; after that, we discuss each of the five major components of the project in more detail.

Background

We describe here two NSF-supported projects that provide the foundation for the project we are beginning.

Cognitive Strategies for Writing Using Advanced Computer Tools

In this recently completed project (NSF Project # IRI-8519517), we had two major goals: to develop an advanced writing environment suitable for professionals who write as a part of their jobs and, second, to use that system to study writers' cognitive strategies, particularly the differences in strategies between experts and novices and those that produced more effective vs. less effective documents. We describe the theoretical basis for this work, the WE system and protocol tools, and then the studies conducted using them.

Our work in theory had two components. Since we were developing a system to help writers produce better documents, we began by synthesizing guidelines for effective documents from research in reading comprehension and cognitive psychology. These guidelines identified the characteristics of a text that help people read more efficiently and accurately. Thus, they provide a target for the writer to aim for in developing a document. The second theoretical component was a theory of cognitive modes and the strategies that guide writers in moving among these modes. This theory was synthesized from research in composition theory, cognitive psychology, and our own experiences as writers. Since this work is described in more detail in the theory section,below, we will not describe it further here.

We built a hypertext-based writing system, which we call the Writing Environment of WE, for short. It is multimodal: individual windows on the screen support either one or two specific cognitive modes used by writers. Each is thus specialized to help the writer accomplish a specific portion of the overall writing task. The system also permits objects, such as a cluster of related ideas, to be moved from one mode to another for further work. In Figure 1, the four system modes can be seen in the default screen layout (each can be expanded to cover the entire screen). The upper left mode is used for exploration, the lower left for organizing and global editing, the lower right for writing and sentence editing, and the upper right for coherence editing. WE is currently being used by several other groups in this country and in Europe for research purposes.

The third component of this project was development of a set of tools and methods for studying users' strategies by analyzing machine-recorded protocols. Since that work is being extended in a second NSF-supported

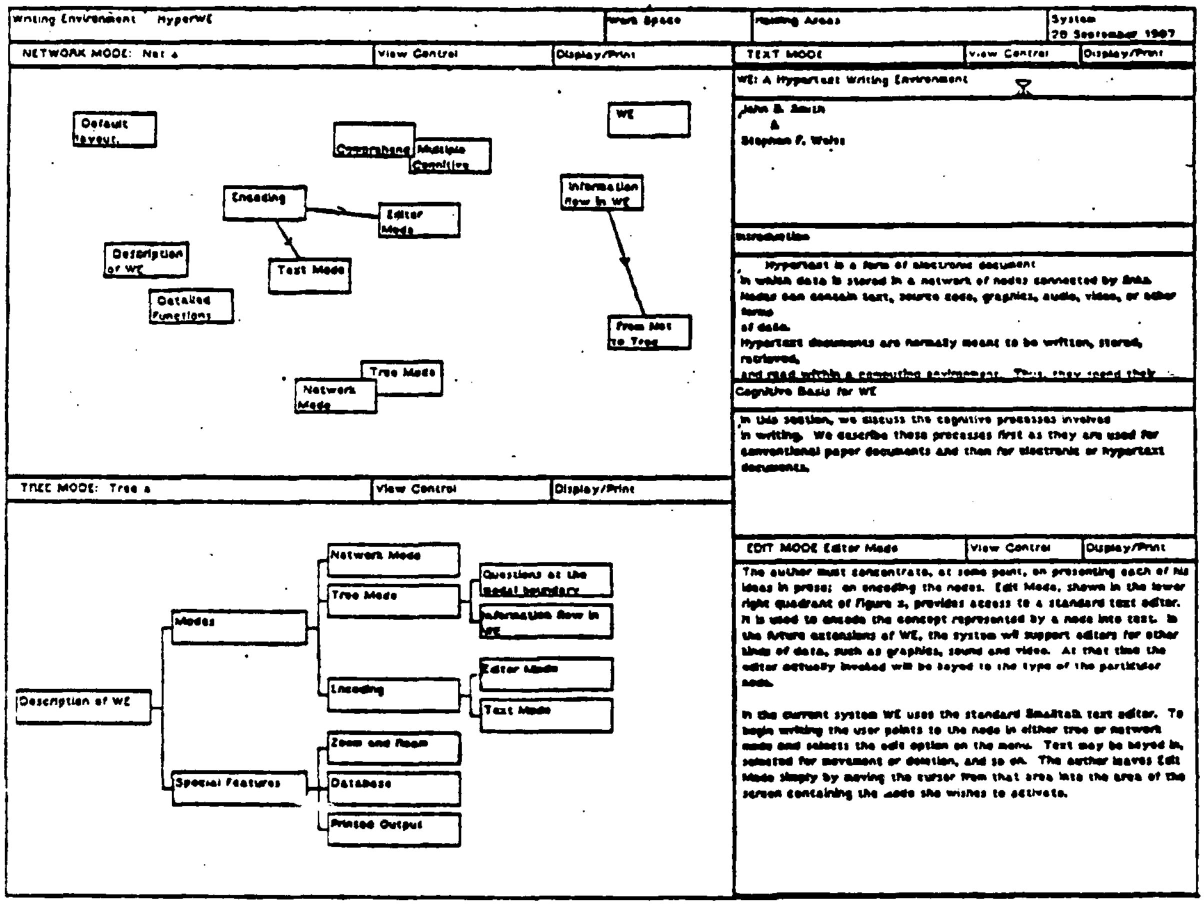

Figure 1: Writing Environment

Four System Modes

project and since we will build on it in this project, we describe it in more detail. This work in methodology had four components. First, we developed tools for automatically recording protocols of user's actions while they work with the WE computer system. The WE system includes sensors imbedded within it that record each users' action These records include each menu selection, the string typed in response to an interface prompt, the spatial location of an action, and similar data. Records are output by the system in accord with a simple protocol language syntax that denotes each such event, its time, and parameters of interest. Thus, the system automatically produces detailed -- but not too detailed -- records of subjects' actions in a formatted, machine-readable form.

A second tool, which replays a session using the recorded protocol as data, provides an overall sense of a user's strategy over a long session. It can replay a session in "real time," in time proportional to the original session, in "fast time" so that we may view a two-hour session in six or eight minutes. And it permits the researcher to manually step through segments of a session.

A third tool is a grammar, written as an expert system, that parses the protocols to produce parse trees that show users' cognitive strategies for particular sessions. The grammar for WE includes five levels of abstraction. Short sequences of *actions* -- such as pointing with the mouse, clicking, selecting the create node option, and typing a string identifying the node -- are, first, mapped onto *operations* -- create node. Each such system operation is then interpreted in terms of its effects on a set of *cognitive products* important for the application task -- such as a cluster of related concepts or an addition to the bottom of a tree. Next, the particular *cognitive process* used to generate the change in cognitive product is inferred. Finally, the *cognitive mode* in which the process takes place is identified.

The data produced by the grammar are analyzed in various ways, usually by extracting values and distributions and passing them to a statistical package. To help us interpret these results, we have developed two kinds of display tools: static and animated. Figures 2 and 3 are examples of static displays, each showing a horizontal slice of a parse tree -- in this case, the cognitive product level. In Figure 2, the user has created exploratory products first; then he constructs the top of his tree, writes blocks of text, then goes back and fills out the bottom of the tree, and, finally, finishes his writing. Thus, his strategy is almost a classic *stages* or *waterfall* model. In Figure 3, the user constantly moves from structure operations to writing operations -- a markedly different strategy that is apparent in the displays.

We have used WE to study writers' cognitive strategies, both in controlled experiments and in naturalistic actual-use situations. First, we compared different representations of hierarchical structures to determine effects on subjects' perception of structural relations and on their ability to draw relational inferences. In a second experiment, we compared the strategies of a group of expert writers (professional technical writers and editors) with those of a group of novice writers (nontechnical graduate

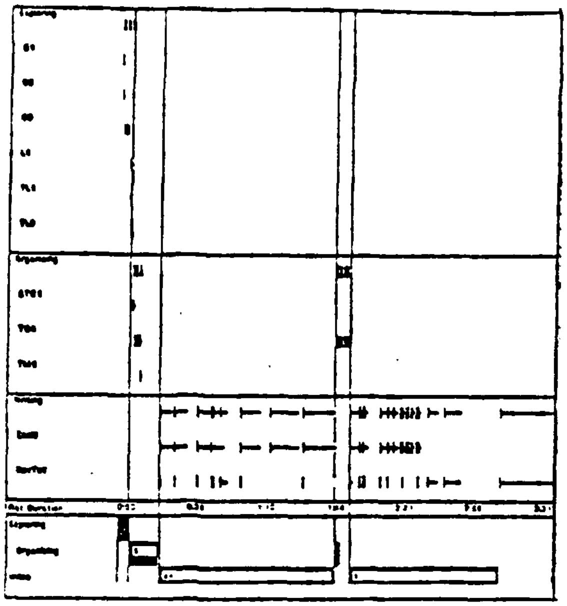

Figure 2: Protocol Display

User Strategy: Stages or Waterfall Model

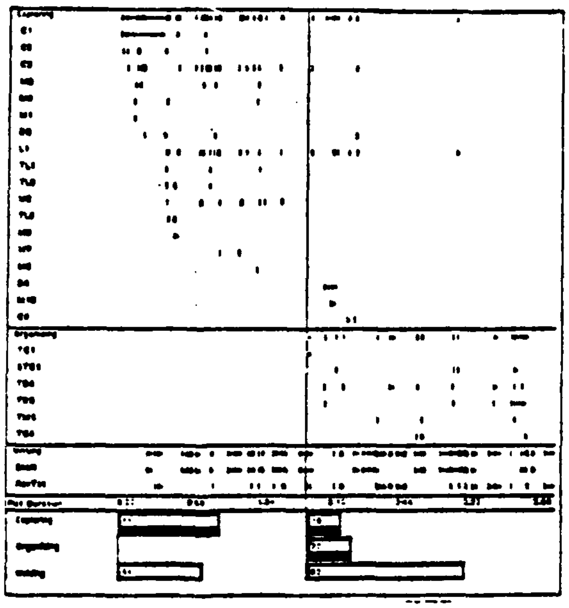

Figure 3: Protocol Display

User Strategy: Planning & Writing Interleaved

students). A third experiment compared strategies relative to the quality of the document produced. A fourth experiment is examining the effects of users' knowledge of the subject matter being written about with their strategies. A fifth experiment, which we will begin shortly, will evaluate differences in users' strategies produced by WE's structuring facilities vs. its text editor, alone; thus, it will measure the overall effectiveness of WE.

We have described this work in a number of journal articles, conference papers, and technical reports. These include the following: [Smith, et. al, 1987a; Smith, et. al, 1987b; Smith & Lansman 1987; Bush, et. al., 1988; Shan, et. al., 1988].

An Environment for Developing and Using Cognitive Grammars to Study Human-Computer Interaction

In this project (NSF Project # IRI-8817305), we are extending and generalizing the protocol analysis tools described above as well as developing new methods for studying the strategies of computer users engaged in complex conceptual tasks.

This work has four major components. First, we are developing a tool to assist the researcher with the management of protocols. It permits the user to sort and select protocols based upon associated attributes and values and to link those selected with different analytic functions, including our grammar and a standard statistical package. Second, we are developing additional display functions to assist researchers in analyzing and comparing protocols. One such tool is an animation mode that links a session being replayed on one screen with various graphical representations of analyses and other data shown on other screens. Thus, the researcher may watch the replay of a subject's session, the parse tree for that session, a transcription of the subject's think-aloud comments, and the researcher's own observations, all shown dynamically but coordinated with one-another. Third, we are conducting exploratory studies to test the usefulness of subjects' retrospective protocols cued by the session replay program. A fourth task is refining the rulebase for our cognitive grammar based on data from our experimental studies and, especially, the retrospective protocols.

These projects have produced results in their own right, but they will also contribute to the work described below. Each will be discussed further in the context to which it applies.

Project Components

Theory

Concept and Strategy

Groups differ widely in their efficiency and in the coherence and integrity of the conceptual products they produce. Some groups gel, functioning as a *collective cognition* in which individual minds work cooperatively to produce a single integrated artifact. Other groups remain federations and produce assemblies of separate or awkwardly connected components.

Why?

Our long-term goal is to answer this question by building a theory of collective cognition and a computer system to support this form of collaboration. The strategy we *wish* we could follow would be to, first, develop the theory, then use the theory to guide development of the system, and, finally, use the system to study specific instances of collaborative behavior. This would approximate the "waterfall" model of software development or the "stages" model of composition. We do not believe a linear strategy such as this is possible.

We will follow a less appealing but, we believe, more realistic strategy. Throughout the project we will observe collaborative groups to gain insights into the nature of collaboration and to test and refine the evolving theory. These observations will constitute a continuous "bass line" for our work. During the first year, we will build a preliminary version of the theory. We know its basic form, which we outline below; but we must develop it in further detail. From this early theory, we will extrapolate guidelines to help observers focus on the intellectual artifacts being constructed by the group and the processes and factors that affect their production. These observations, in turn, will test both theory and guidelines for accuracy, completeness, and usefulness, leading to their revision. Thus, building theory, extracting guidelines, and observing groups form a cycle that we will iterate throughout the project.

Into this cycle we will insert the collaborative support system. It will be based on our understanding of the application and theory at the time and on the views of other researchers described in the literature. As we continue our studies of groups -- working conventionally at first and then with our system -- we will modify the theory as needed to include computer-supported collaboration. As our understanding of collaboration grows, we will revise the system design accordingly, using the system-building tools described below. By the end of the project, we expect this iterative strategy to produce viable versions of both theory and system.

Related Theory and Research

To build a theory of collective cognition and collaboration, a multidisciplinary team will combine research from several fields, including cognitive psychology, anthropology, and computer science. In the remainder of this section, we first describe three broad areas of research that contain components for a theory. Then, we outline the basic form of the theory.

Cognitive Modes

Informal references to different *modes*, or kinds, of thinking are common. People have long realized that writing a technical report, giving directions, or solving a math problem draw on different mental skills and knowledge. Only recently have attempts been made to define the concept precisely. Ken Hammond [Hammond, 1988] and his colleagues have defined a single dimension of modes they use to characterize different kinds of thinking engaged by technical experts, such as medical diagnosticians and aviation weather analysts. Our group has described a multi-dimensional notion of cognitive mode, synthesized from research in composition theory and cognitive psychology, including comprehension [Smith & Lansman, 1988; Hayes & Flower, 1980, Hayes & Flower, 1986; Flower & Hayes, 1984; Bereiter & Scardamalia, 1987; Kintsch & van Dijk, 1978]. We examined in detail modes used by writers of technical and scientific documents. However, the concept is general and can be extended to other forms of intellectual work. We are currently doing this for software development and for data analysis.

A particular cognitive mode is an interdependent combination of *goals, products, processes*, and *constraints*. The *product* of a mode is the symbolization of a concept or relation among concepts. Different cognitive modes provide different options for representing concepts or structures, such as words, diagrams, notes, outlines, and other forms. Thus, different forms prevail in different modes. *Processes* act on products to define them or to transform one form into another. Thus, certain processes are favored in certain modes, while others are de-emphasized or suppressed. The *goal* of a mode represents the individual's intention for engaging that particular way of thinking. While goals are abstract, they are made concrete in the particular product the individual aims to produce. The *constraints* for a mode determine the choices available. Constraints are relaxed or tightened in accord with the individual's large-scale strategies for engaging different modes of thinking for different purposes.

To illustrate this concept, consider two modes used by writers: exploratory thinking and organizing. During exploration, the goal is to externalize ideas, consider different combinations, and to gain a general sense of the information available or missing. Thus, constraints are minimal to encourage creativity and multiple perspectives. The processes that are emphasized are memory recall, associating, relating, and building small component structures. The products generated are, thus, notes, jottings, diagrams, perhaps loose networks of ideas. During organization, the goal is to plan the actual document to be written; thus, constraints are tightened to produce a logical, coherent organizational plan. That plan is normally a hierarchy or other regular form. And the processes are analyzing, synthesizing, sustained conceptual building, and refinement based on noting consistent/inconsistent relations in the structure. *Exploration* and *organization* are, thus, distinctly different ways of thinking. And they differ still from other activities such as actual writing and several forms of editing.

Cognitive modes are used strategically. Individuals move from one mode to another in accord with a general procedure they know and use to accomplish a particular intellectual activity. But they also move back and forth among modes -- both recursively and iteratively -- to solve problems that arise or to take care of new developments, such as the appearance of new information not available earlier. Consequently, this theory of modes and strategies is *not* a stages or waterfall model but rather a dynamic system in which the history of an individual's movement among modes would normally form a network, rather than a linear sequence.

One can characterize intellectual behavior by identifying the particular modes people use for a given task, their overall strategies, and their responses to particular problems. We have followed this approach in extensive studies of writers' cognitive strategies and used the insights gained to develop and refine a mode-based Writing Environment [Smith, et. al, 1986; Shan, et. al., 1988] in which different system modes correspond with specific cognitive modes. We are now following a similar approach to develop mode-based systems to support software development and data analysis and to study these tasks.

Activity Theory

The concept of cognitive mode does not take into account the impact of social and cultural interactions that impinge upon the thinking of people working in the real world. These issues were addressed in the work of Vygotsky, Leont'ev, and the body of subsequent research known as *activity theory*. We will build on four major concepts:

- o mediating device
- o higher (conscious, voluntary) mental functions
- o zone of proximal development
- o activity.

[This summary is derived from Vygotsky (1962, 1978; 1986; 1987) with guidance from Kozulin (1986), Minick (1987), Wertsch (1985), Scribner (1984), Lee (1985), and Holland & Valsiner, 1987)]

For the "cultural historical school of psychology," culture is essential in the development of human cognition. A group's cultural tradition provides the means, in the form of symbols, to transform lower-level, biologically-based mental functions into *higher mental functions*. Symbols function in the mental world as tools do in the physical world. They become psychological devices for mediating between one's mental states and processes and one's environment. For example, remembering, as made possible by an individual's biologically given mental functions, is a "lower-level" mental function. However, when people learn to use *mediating devices* -- such as mnemonic associations derived from their language and culture -- as tools for remembering, their memory capacity is increased and they have more conscious control over the process. Similarly, computer systems that help people perform intellectual tasks are an important new form of mediating device. As we discuss below, taking into account the mediating influence of a computer system on an individual's or group's intellectual activities is essential.

Vygotsky argued that mediating devices are largely invisible under normal circumstances since once learned or developed, they become habitual and are thought to be "just the way we think" in performing a particular activity. Vygotsky called habitual behaviors of this sort *fossilized.* Since these mental tools are normally "invisible", mediated cognition can best be observed at the time when new mediating devices are being developed to solve a new problem and before the new form of cognitive behavior becomes routine and reflexive. Thus, focusing on *snags* and their resolution is important. For a computer system, several kinds of snags would be expected, but perhaps the most interesting form would occur after changes are made to the system and before users have become familiar with the new function or operation.

Whereas Vygotsky stressed semiotic mediation and the importance of cultural meaning systems in cognition, his student and colleague, A. N. Leont'ev and subsequent activity theorists, emphasized the idea that cognition is situated in *activity.* Individual cognition always takes place in, and is responsive to, socially created activities Individual thinkers always interpret the topic at hand in relation to an activity learned from their fellows. Activities are usually organized around a common motive and directed toward specific objects or products [Scribner 1984]. Consequently, they are inherently goal directed. Thus, while derived from different intellectual traditions, activity theory and the theory of modes are compatible and complementary perspectives in their emphasis on products, goals, and activities/processes.

Finally, Vygotsky argued that before we can carry out a task by ourselves, we must first learn the skill in the context of another person. This situation Vygotsky's called the *zone* or *proximal development* or *zoped.* New skills are learned through collaborative work that involves at least one (relative) neophyte and at least one (relative) expert. As the neophyte's ability develops to carry out the task alone, the expert curtails his or her participation. By analogy, as some groups learn to collaborate more effectively, their successive states may be considered forms of *higher cognition,* as described above, but understood developmentally from the perspective of *zoped.*

Thus, Vygotskian/Activity Theory provides both a set of useful concepts as well as a suggested path of development for constructing a plausible notion of collective cognition.

Group Interaction Theories

Research in group interactions is extensive. Most relevant for our purposes are studies that focus on factors that affect groups' technological environments [De Sanctis and Gallupe 1987; Kiesler et al 1984], their social environments [see especially research on work groups in business corporations--Alderfer and Smith 1982, Ancona 1987], their composition [Rousseau 1985, Kenny 1985], and their conflicts. Against this background, we are primarily concerned with the development of the groups over time [Brandstatter et al 1982, Guzzo 1982, Hill 1982, Laughlin and McGlynn 1986, Clark and Stephenson 1989; Wegner et al 1985].

For our purposes, the most useful work from social psychology is their study of group development. Studies of task-oriented groups -- both in the laboratory and in natural settings --show that they grow more productive as they develop common knowledge about the task, leaders, and strategy. But as groups grow older, they also tend to become more insulated. Especially pertinent are studies by [Gersick 1988, Insko et al 1980, 1982, 1983, Katz 1982]. Thus, if the focus of observation for groups is on the artifacts they produce -- as will be the case for our studies -- then one must be conscious of these less tangible forms as well as concrete products. A useful discussion of the diversity of these components is [Moreland, 1987].

The anthropological and sociological literature includes a number of studies of group structure; useful for our purposes are concepts of status systems, norms, and roles [especially informal roles such as newcomer or scapegoat]. Status systems are important because they affect the contributions of members. As Levine and Moreland [1990] point out "People with higher status speak more often than others, are more likely to criticize, command, or interrupt others, and are spoken to more often than others." From our preliminary observations of groups, we have noticed that some group modes, such as a presentation, permit low status, and normally silent, group members to make a verbal contribution to the whole group. This work raises a number of question that we will apply to collaborative groups working in a computer and communication environment. For example, do people behave differently when they work together through the multimedia communication system versus through face-to-face interaction in a meeting, relative to their status in the group or to other factors?

Because of the importance ascribed by Vygotsky to the resolution of snags in the development of new forms of mediation, we are also concerned with group conflict. The psychological literature contains helpful delineations of different types of conflict and conflict-related processes. For example, Levine and Moreland [1990] identify five major areas: social dilemmas, power, bargaining, coalition formation, and majority and minority influence. Especially useful are studies of the role conflict plays in the generation of subgroups --both functional subgroups and majority/minority coalitions -- and in their interactions with the group as a whole [see, for example, Insko et al 1985, Kerr et al 1987, Levine and Russo 1987, Nemeth 1986, Clark and Maass 1988].

The literature concerned especially with working groups is large. Much of it is concerned with group structure and group dynamics, particularly in the context of decision-making by groups. A recent review of this work is [Kraemer & King, 1988]. However, most of these studies have emphasized avoiding negative aspects of groups, such as competition, rather than concentrating on the positive aspects of how, in fact, participants collaborate. Thus, this body of research is less helpful than would be expected.

Finally, substantial research has been reported on the differential effects of using alternative communication channels. Most comparisons have been between face-to-face communication and audio teleconferencing (with or without video), but a few studies have included mediation by computer. Comprehensive reviews of this research have been offered by [Short, 1976], [Williams, 1977], [Williams, 1978], [Fowler, 1980], [Heimstra, 1982], and [Rice 1984]. One study, [Collins, 1988], has explored the relationship between a particular computer-based tool and work group structure and is, thus, applicable to our work as a model.

Modes of Activity

To observe the evolution of groups and to characterize differences among them, we will develop a theory of *modes of activity* and the *strategies* groups adopt that govern their movement among modes. During the first months of the project, we will build a preliminary version that might perhaps more appropriately be considered an architecture or framework in which to build a theory, rather than a fully-formed theory, per se. It will draw most heavily on activity theory and the theory of cognitive modes and strategies, but it will also include individual concepts from the various theories of group interaction. As we outline the theory and guidelines, we will point out the more important of these relationships.

A mode of activity will be defined as a particular configuration of goals, products, processes, and constraints as they are manifest in the situated activities of a group and its members. This notion of mode is, thus, a fusion of activity theory and the theory of cognitive modes and strategies, discussed above. The real issue will be to recognize these four constituents as they occur within the activities of a group and its members. The key to doing is to focus, first, on the products being generated by the group.

Product and process form a dialectic: processes operate on intellectual products to define them and to transform one type or structure into another. The set of product types included within modes of activity will be more extensive than those for cognitive modes. It will include tangible products: those that become part of the evolving conceptual artifact being constructed by the group but also those that are working products, such as minutes of meetings or ideas sketched on a whiteboard that are not part of the central artifact, per se. The set of product types will also include intangible products, such as a body of shared knowledge. To generate and transform this larger set, groups will employ a larger range of processes than individuals, as well.

Similarly, goals and constraints form an axis. Goals represent the intentions of the group; constraints, the limitations and other shaping or inhibiting factors within the situational matrix that affects the group as it attempts to realize its goals. Goals and constraints will play a particularly important role in our studies, since they are the main factors that express the "situatedness" of the group within its organizational and social contexts. A theory of modes of activity for groups must also take into account the individuals that comprise a group as well as the group as a whole. This portion of the theory, especially, will incorporate concepts and results from *group interaction theories*. For example, there may be instances when the members of a group are fairly advanced in their respective thinking about the structure of the central artifact; thus, their individual modes of thinking might be described as *organizational*. However, if the group has not reached consensus on the structure for the central artifact and the views of its individual members differ widely, then the group as a whole is still in an early *exploratory* mode of activity. Thus, the mode for the group may not be simply the sum of the modes of activity of its individual members.

Guidelines for Observing Groups

Whereas a theory of modes and strategies will describe the general mechanism of collaborative groups, the guidelines will identify specific modes -- and their constituent products, processes, goals, and constraints -- engaged by groups performing particular tasks and the specific strategies and problems that cause them to move from one mode to another. We will begin by making ethnographic-type observations of groups. We will focus, first, on the particular informational artifact they are constructing at any one moment; then we will characterize the processes and procedures they are using, their apparent goal in performing this activity, and the constraints that affect their work. From these observations, we will identify the set of mode-types that are used and reused by the group over the course of its work. We will also look for specific problems or factors that cause them to shift from one mode to another to infer a set of strategies. In subsequent studies of other groups, we will test the guidelines by seeing if it accounts for most group activities or misses important actions, provides useful insights, etc.

For example, groups and their members working collaboratively on a proposal or other document are likely to adopt all of the modes used by individual writers. They explore, organize, draft, revise, etc. But they also engage in other *group* modes: building a body of shared knowledge, agreeing on a common set of goals, negotiating both substantive and procedural issues, browsing and reviewing one-another's work, forming alliances, etc. Thus, while the modes of activity engaged by a group are both more numerous and more complex than the cognitive modes used by individuals working alone, a mode/strategy framework can be built for this kind of work. Our preliminary studies encourage us to believe that it will also be quite useful.

The collaboration support system we will build will include different system modes to support different modes of activity for the group. It will also include mode-mode communications capabilities so that products developed in one system mode can be worked on in another mode more appropriate for the activity of the group at that time. As we develop a better understanding of groups' modal behavior, we will use that knowledge to modify the system design. We will add new modes, collapse modes, add or take away specific functions, etc. to make the system more resonant with the groups using it. But as we observe groups and build theory, we must be aware of the symbiotic interaction between system and group, in the form of mediation.

Vygotsky was concerned with mediation, primarily through the language that melds a group of people into a culture. Computer systems, such as a collaboration system, are powerful mediating devices. As such, these tools can be expected to influence the behavior of groups using them. Since the system will provide direct representations of concepts and conceptual structures, groups can be expected to begin thinking within the terms and constructs provided by the system. That is, if the data model for a document is a graph of nodes that contain paragraphs and links that denote sequence and relation, then users writing a document collaboratively will soon begin to discuss their activities within these terms -- e.g., "I think this node belongs here rather than there." (We have seen this in user's of our WE system.) Once the behavior of a group becomes habitual, and hence fossilized in Vygotsky's terms, the members will be unaware of the system's mediating influence. However, as problems arise and are solved and as new versions of the system are provided to the group, these "snags" and developments will bring the mediating nature of the system, once again, into perspective. These will be important times for our studies since they will be periods of development in the group as an intellectual organism. That is, if the changes in the system produce the benefits they were intended to produce -- i.e., increased capability of the group to work together productively -- then the group will have advanced to a higher form of (collective) intellectual activity -- an approximation of Vygotsky's notion of *higher mental function* and our notion of *collective cognition*.

To step back and trace the ridgeline of this discussion, we have seen that developing a general theory of collective cognition and group collaboration and developing guidelines for observing specific groups go hand-in-hand. As we study actual groups by observing them within the framework of the guidelines, the insights brought into focus will lead to revisions in the theory. But when we introduce a powerful computer system into the process, we affect the nature of the collaborative process. Thus, theory must expand to include this new form of mediated activity. But this expanded theory will lead, we hope, to further improvements in the system, which in turn will lead to still other changes in group collaborative processes, etc. Thus, the entire construct of theory, guidelines, system, and studies form a cycle, but a cycle that does not repeat. Rather, it expands continuously.

Summary of Expected Results

o Theory of modes of activity and strategies for their use

o Guidelines for observing groups engaged in a particular task

System-Building Tools and Communications Infrastructure

Goals & Strategy

Our goal is to develop tools that will let us craft both general and application-specific systems for collaboration. For effective experimentation, we need capabilities for rapid prototyping and easy reconfiguration. Since we will study both synchronous and asynchronous collaborations, our tools must integrate functions needed for both types. All of the tools described here are now developed in at least prototype form, but more research is needed to unify and extend them for multi-user collaborations. We believe this research will result in new understanding of how to fashion tools for creating collaboration support systems.

We describe these tools from the perspective of the type of collaboration they most directly support -- asynchronous or synchronous. For asynchronous requirements, we describe a user interface management system that will permit us to easily reconfigure not just the user interface, but also the function and organization of the system. (It is this tool that will make practical the iterative strategy of system development described below.) We then describe the graph data service that provides the underlying data model and supports multiple concurrent users for hypermedia applications.

For synchronous interactions, we describe two specific components for collaborative work. The first is an extension of the user interface management system to provide shared visual workspaces. These will permit multiple users to work simultaneously on the same conceptual artifacts. The second is sight and sound, which we believe are essential for sustaining synchronous interactions among people. Our experimental groups will be provided with live audio and video communications in addition to the shared graphical interfaces.

Asynchronous Collaboration

User interface management system. The central element of our rapid prototyping capability is a user interface management system, called MoDE, especially designed for easy reconfiguration of applications [Shan 1989, 1990]. Using it, the developer can construct a user interface for a system by selecting components from a library of objects and adapting or connecting them through direct-manipulation. An interface created with MoDE is a hierarchy of "modes" -- each mode is an area on the display screen in which interactions with the user are different from those in any surrounding area. Thus, there is a natural relationship between system interface modes and the modes of activity used by the group.

Each mode is defined by its appearance, its semantics, and its interaction with the user. The connection model used in MoDE provides a clear separation of the user interface from the application without limiting the information flow between them. Separation is achieved by introducing a notion of "semantic objects." User interface modes are connected directly to semantic objects and, through them, indirectly to the application. Each semantic object reveals just a part of the potentially large and complex interface to an application. A complicated application interface can, thus, be divided into a number of small, manageable semantic objects that are individually maintained. Thus, either interface or application can be rapidly reconfigured with fewer and simpler changes.

Another important aspect of MoDE is the event-driven implementation of input control for the user interface. Not only does this provide better performance and eliminate missed events, it also is the foundation for the protocol-gathering tools (described in a later section). We observe that MoDE could have broader implications as a model for general user-interface tools. A working version of MoDE exists today, and further extensions should be completed in the next few months.

Graph Data Service. The most natural data abstraction for hypermedia applications is a graph, in particular, an attributed directed graph (i.e., a graph that can have an arbitrary number of user-defined attributes on nodes and links, and links that have direction -- they connect a source node to a target node). Hypermedia applications are constructed more easily if the underlying database provides the graph abstraction at its interface. We are developing a graph data service (the implementation is a distributed client/server model) that will support a multi-user collaborative system. For the multi-user environment, notions of ownership, protection, and concurrency control are provided. Very few experimental graph data services have been tried, some examples are HAM [Campbell & Goodman 1987] and GRAS [Brandes & Lewerentz 1985] (we have found none so far that support both asynchronous and synchronous modes of collaboration). A first prototype of our system is currently being implemented and tested.

The graph service actually provides multiple levels of graph abstractions. At the lowest interface used by a client application, the objects provided are nodes, links, attributes, and subgraphs. At one layer higher are typed subgraphs (e.g., list, tree, connected graph,...) and appropriate operations which enforce constraints and preserve essential properties of the type. Subgraphs may share nodes and links. At the highest, most abstract, level, the notion of a composite subgraph is supported. Composite subgraphs are groups of subgraph instances constrained to relate in certain ways such as embedding (one graph exists within another's node set) and bridging (a collection of links from one graph to another).

The lowest level of the server structure is the data base management system (DBMS) that provides persistent storage and concurrency control.

We are implementing a rudimentary system ourselves, but we hope to find a commercial system that will serve our purposes. So far, we have not found a suitable object-oriented database system that provides acceptable performance, and we believe the relational model is inappropriate for our purposes..

Synchronous Collaboration

Shared Visual Workspaces. These are abstractions that denote a collection of objects (e.g. documents, images, programs) and the tools used to view and change them. Each participant in a collaboration can logically share the same view of these objects (and the same facilities for operating on the objects). A shared visual workspace facility permits multiple users to work together on the same object(s) at the same time. It supports a form of distributed electronic meeting in which team members work at their individual workstations, linked by the network. In this context, they all receive the same visual information. While one is performing an operation, the others see those interactions. Cooperation is managed through various conventions, such as "passing the chalk," to keep participants from interfering with one another. The design issues for shared visual workspaces and the results from our prior research on these systems have been discussed elsewhere [Abdel-Wahab et al 1988, Guan 1989, Calingaert et al 1990].

Our prior research has addressed several other important conceptual issues: (1) provision for dynamic addition or removal of users participating in a shared visual workspace, (2) ability of users to participate in multiple shared workspaces concurrently, and (3) shared workspaces that include views from multiple tools. We have also developed a number of mechanisms to enhance systems support for synchronous collaboration. Among these are protocols for the formation, modification, and management of dynamically changing groups of users with shared workspace views, and a concept of conditionally jointly-owned objects with an associated protection mechanism [Guan et al 1990]. Portions of these will be adapted for this environment.

To provide shared visual workspaces, we will extend the MoDE system in two dimensions: (1) building user interfaces that work with the X Window System [Scheifler & Gettys 1986], and (2) providing for dynamic configuration of workspace windows so they can be displayed on, and manipulated from, multiple users' workstations. By extending the MoDE system to support multiuser interfaces, we retain the capability for rapid prototyping and for event-driven recording of user protocols. While not a complete solution for multi-user interfaces, the X Window System offers many desirable properties. The most appealing aspect is the ability to interconnect distributed applications and user interfaces over networks. It is also widely available, its components operate in many heterogeneous environments, and it can be readily used to support multi-user collaborations.

Since we cannot possibly create all of the software needed for realistic support of any application, we will make existing (and familiar) programs readily available to group members. In most cases, these tools have been written for a single user -- they assume a single input source and present a single view (most programs useful in collaborative environments will be interactive). Since it would be impractical to modify even the most-used tools, it is necessary to provide "adapters" that allow single-user tools to be used unchanged in multi-user environments. Such adaptation can be accomplished by interposing, between the tool and its users, "agent" processes that present a single input stream to the tool and replicate its output to multiple viewers. In earlier work, we have developed agent processes that convert a traditional single-user text-based tool without graphics or mouse input, such as "vi", into a multi-user tools for a group of remote users. With the networking capabilities of the X Window System, we will be able to provide a similar mechanism to adapt X11-client programs so they can be shared and used by a group of remote collaborators.

Shared visual workspaces based on the X Windows System can be widely distributed to workstations in our building, local campus, or state-wide via a network of Ethernets located at most of the state's universities (this network consists of bridged Ethernet segments interconnected by the microwave facilities provided by the Microelectronics Center of North Carolina (MCNC)). MCNC also provides connections to the National Science Foundation national network. This affords us the capability for experiments using our systems for groups where the members are not collocated on one campus.

Sight and Sound. The shared visual workspaces will be supplemented by other media for communication so that participants may also discuss their thinking and their actions. We will be particularly concerned with the kinds of problems/issues that are handled by various forms of conferencing. We would like to determine the situations in which voice and graphics alone are inadequate, as well as those in which video is inadequate and face-to-face communication is preferred. Offices, laboratories, and conference rooms in our building (Sitterson Hall) have either two or four 75 ohm cable appearances. Any of the 75 ohm outlets can be attached selectively to the building's CATV system which is independent from, but connected to, the campus system. The Sitterson Hall CATV system is a mid-split two-way system which can support 16 two-way motion video paths. With a modest additional expenditure, we can equip 16 workstation locations in offices or laboratories with a color TV camera, a microphone, and a color TV monitor. Using the CATV system, these facilities can be dynamically configured to support concurrent conferences among multiple, independent groups of participants. Within each conference, voice-activated camera switching can be used to make the current speaker seen and heard by all participants. We can also use quad-split images to make up to four participants visible concurrently. Another straightforward extension is use of commercially-available adapters to display motion video in windows on the workstation display. When used in conjunction with the X11-based shared visual workspaces, this facility provides a reasonable (and low risk) approximation to integrated multimedia workstations and can be used to evaluate experimentally the impact of media on group collaborations.

The Sitterson CATV system is also coupled with the campus broadband system and the statewide microwave system thereby providing two-way video conferencing capability to MCNC, many of the state's universities, and all four of the state's medical schools. This extends our capability for experiments using multi-media conferencing and shared visual workspaces to many locations throughout the state. MCNC will collaborate in these studies.

For experiments in augmenting collaborations with voice alone, we can equip workstation locations with speaker-phones interconnected via the Wang InteCom S-10 Integrated Business exchange digital switching system which serves as the in-house "PBX" for Sitterson Hall. The InteCom software supports an interface (OAI) which allows control of some switching functions (e.g. conference call setup) to be performed from workstations. We will implement, as part of the user interface, a convenient mechanism (e.g., menu or button selections) for participants to use for establishing voice conferences, alone or supplementing the shared visual workspaces.

Summary of Expected Results

This research will result in new understanding of how to build tools for creating collaboration support systems, specifically, user interface management systems, hypermedia graph-data services, and shared visual workspaces. It should also produce new evaluations of the relative effectiveness of various media for group communications.

Relation to Other Parts of the Project

The tools and communication infrastructure provide the framework for implementing the application testbed environment and for supporting group activities during the experimental observations.

Application Testbed System

Goals & Strategy

We will build a collaboration support system to serve as a testbed for studying the activities of groups working together to build and express a complex conceptual structure. It will be realized as a highly reconfigurable platform for testing and refining functions to support groups working in a distributed computing environment. The key task we will support is the creation and refinement of conceptual structures expressed by computer-manipulated materials. As noted above, this task is central to a number of scientific and engineering activities. Our system will provide support for three kinds of information: human language text, programming language text, and 2-D graphics (e.g., line diagrams). Users will be able to create, browse, and link (in the hypermedia sense) elements of all three types of information. Given these, the system can be used in a number of collaborative applications, including designing software systems, writing technical documents, formulating project plans, etc. Since our research group will be engaged in all these activities, we will insist on using our system ourselves (long before we would ask anyone else to do so). Many of the groups we plan to observe will be engaged in software design and our system will have a strong emphasis on supporting this application. No attempt will be made, however, to create new tools for software engineering -- the system will only provide interfaces so that existing compilers, debuggers, make, etc., can be used.

Background

There is a great deal of research on computer systems that support cooperative work. Many are surveyed in [Kraemer & King 1988], or reported in the proceedings of the two conferences on Computer Supported Collaborative Work (CSCW) [ACM 1986, 1988] and the book by Greif [Greif 1988]. Most have to do with group decision making, group dynamics, or communications hardware and software.

NICK is a system developed at MCC for facilitating decentralized meetings, providing aids for preparing for a meeting, for conducting the meeting (e.g. an electronic chalkboard), and for summarizing the meeting afterward [Begeman et al 1986].

Rapport, a system developed at AT&T, facilitates decentralized meetings. It allows standard UNIX software to be run by one participant with the displayed results shared among all participants [Ensor et al 1988].

Lantz (Stanford) has reported an experimental system for computer conferencing based on provision of shared user interfaces to existing single-user applications [Lantz 1986].

Cognoter supports collaborative work in the CoLab at Xerox PARC. Unlike the previous systems, it is designed to support face-to-face meetings and directly supports prewriting tasks of brainstorming, organization, and evaluation. Each participant sees the same screen and may interact with the system (add nodes, move nodes, create links, etc.) according to rules appropriate to the particular activity [Foster 1986].

Information Lens and *Object Lens* are two generations of systems for information sharing and coordination created by Malone et al at MIT. These systems allow users to build cooperative applications using semistructured messages and other objects. These can be manipulated by rule-based agents to automatically process information in ways specific to the need of a particular group or individual [Malone et al 1987, Lai et al 1988].

gIBIS is a hypertext system developed at MCC to support early design deliberations in a group with particular emphasis on capturing the rationale for decisions. It implements a specific method called "issue based information systems" which models the design process as a rhetoric or argumentation among holders of diverse viewpoints [Conklin & Begeman 1988].

Over the past four years, we developed a hypertext-based Writing Environment (*WE*) that includes many of the ideas that are generalized and extended in this project. During the past year, we have extended our work to a second application: a software-oriented environment aimed toward maintaining a mature software design and supporting its continued evolution. Unlike other software tools, this system places strong emphasis on browsing and on tools to assist users attempting to understand both the structure and technical details in order to respond to questions and to make incremental changes. These two earlier systems have provided us with a body of ideas, prototype components, and experience that we will draw on to develop the more complex and extensive collaborative system, described below.

Testbed System

The research system we will build differs from the systems described above in several respects: its integration of support for both synchronous and asynchronous collaboration, its support for protocol collection and session replay, and its use of an integrated but shared hypergraph data model. We are also emphasizing media for communication among people not meeting face-to-face. Perhaps the most fundamental difference is that while other systems are ends in themselves, the system we will build is primarily an environment for further research into collaboration. Figure 4 depicts the system we intend to build; the discussion that follows describes components that can be seen in that representation.

The system will be a based on a hypermedia storage model and will support both synchronous and asynchronous collaboration in which small groups (O(10) members) develop large, complex conceptual structures, e.g., software systems and associated documents and architectural diagrams. It must be robust enough to be used as a system of choice by groups working in a research environment. It should be designed to be scalable for supporting larger groups (O(100) members). And it must be extensible and reconfigurable so that we may test new features and different combinations of features. Key architectural elements that provide a stable framework in which components will vary are the following:

- o based on concept of a central conceptual artifact that the group develops over the course of a project
- o supports conventional, but also ad hoc, subgraph structures for links within the central artifact
- o provides flexible but consistent interface architecture based on a concept of modes of activity
- o supports both synchronous and asynchronous collaboration

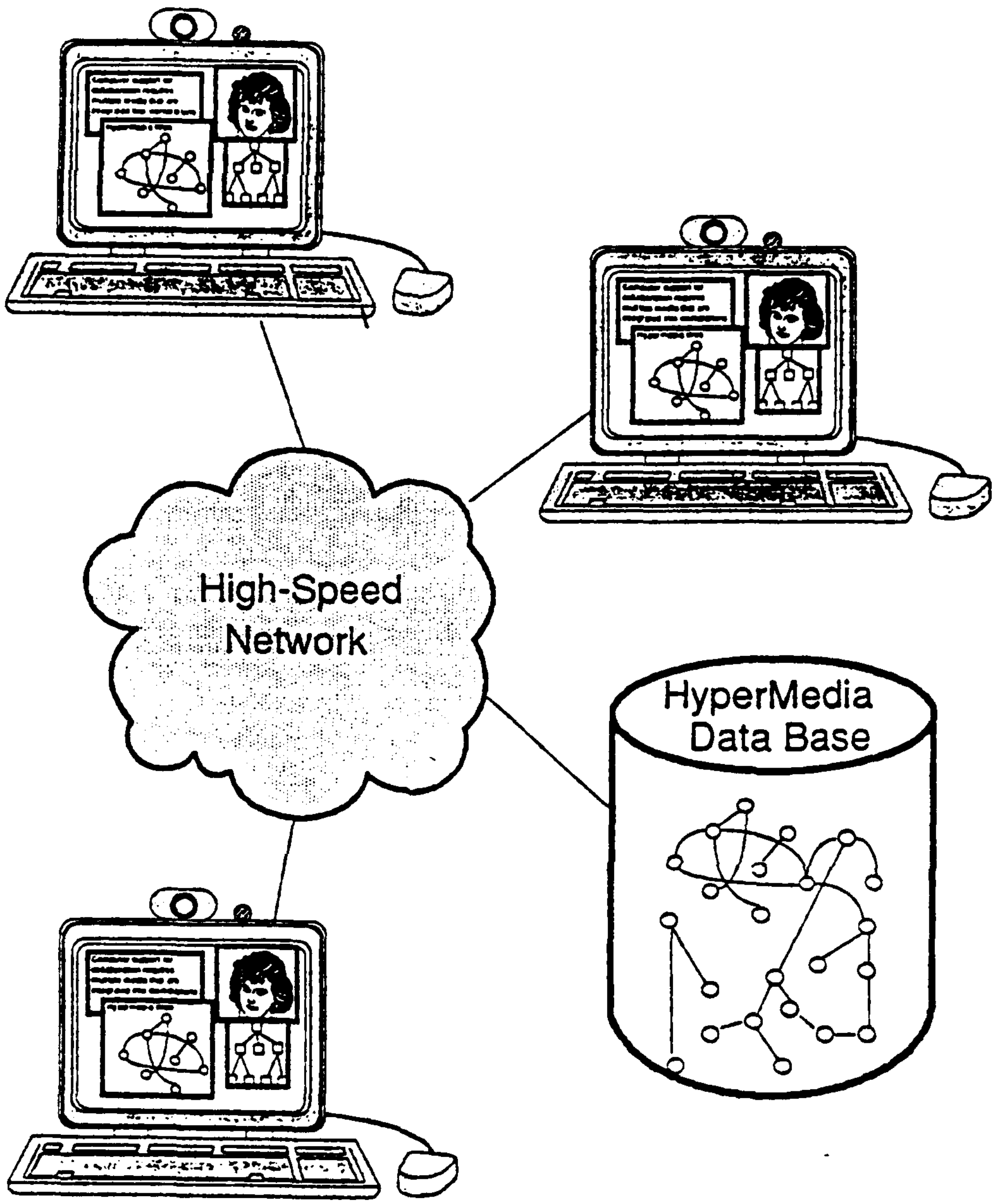

Figure 4:
Overview of Collaboration System

- incorporates multimedia communication, including shared graphical workspaces, voice, and video
- records protocols of users' interactions with the system and with one-another
- can easily be reconfigured for specific studies and to evaluate specific user functions or combinations of functions

The system will encourage users to think of themselves as working on different parts of a single large artifact -- an artifact that provides different but linked views that correspond with familiar components. The single large artifact can be represented by a directed graph structure that includes attributes and values associated with both links and nodes. The nodes identify blocks of content -- text, source code, or diagrams -- stored separate from the graph structure. The links identify relations among these content nodes, such as the hierarchical structure of a document or the calling sequence or message network of programs. Attributes on links, among other functions, identify elements of conventional project components -- such as a requirements document for a software system, a functional decomposition diagram of the system architecture, source code, or internal documentation -- or ad hoc components defined by members of the group -- such as work in progress or a collection of elements located in several conventional components. While these individual components or subgraphs can be viewed separately, they are also connected by links that associate a node in one component subgraph with one or more nodes in other subgraphs. Thus, for example, a source code node might be linked to a node in a requirements document that describes a particular system function, to a node in an architectural decomposition diagram, and to a node containing it's "internal documentation."

The system will also store "external" information that supports the group but is not part of the conceptual product, per se. A significant source of "extra" information is electronic mail and our system will include a mail interface that allows mail to/from any source to be linked easily to other information. Other informations such as minutes of meetings, private notes and work, schedules, etc., can be included in the graph as well.

Our work on the underlying graph abstractions supporting these applications addresses several important research issues. Current graph servers are based on a data model that is primarily hierarchical. This is an important limitation for many applications; for example, large software systems present complex graph structures that are not even planer, much less hierarchical [Brooks 1987]. The concept of subgraph composition extends the richness of the graph model. Another issue addressed by the subgraph composition model is user navigation and comprehension. The problem of getting "lost in hyperspace" is well known; erecting separate, but linked, structures over the general graph model is a promising approach for providing coherence as well as flexibility. Finally, the model raises interesting possibilities for defining concepts of completeness and correctness for the central artifact. Problems of maintaining consistency among documents or among documents and programs are notorious. Defining expected correspondences between subgraphs -- such as between source code and internal documentation -- and noting when one has been updated and not the other could indicate specific (potential) inconsistencies. Graph correctness -- such as a required hierarchical structure for a particular document subgraph -- is a stronger standard that could be rigorously enforced or, at least, noted.

The user interface will be based on a uniform architectural construct -- a *mode*. Each mode corresponds to a window; but in addition to maintaining separate areas of display, each maintains a particular type of data structure and supports only functions that do not violate the integrity of that particular data object. For example, modes that work with hierarchies provide a set of node definition and linking functions different from those provided by modes that work with graphs. The system will support communication between modes in several forms such as cut and paste of data objects, appropriately filtered or transformed We will also wrap standard tools, such as editors, in a mode framework to provide consistent appearance and interaction.

The system will support both synchronous and asynchronous collaborations. To support synchronous interaction, the system will provide two important mechanisms -- shared visual workspaces and voice/video communications. Mode windows may be shared visual workspaces (described in earlier sections). Thus, two or more group members may work on the same portion of the same subgraph at the same time. We will explore different mechanisms to coordinate their interaction and to maintain the integrity of the data object. They may share the same view of the subgraph, but we will also explore support for different perspectives on the data object. We will also include voice and video communication. Thus, members may work on the same object while also seeing and/or talking with one-another. Video and multi-media conferencing have generated considerable interest in recent years; our project differs from others in its focus on evaluating the effectiveness and added benefit of these rich, but expensive, channels. We can do this by configuring systems that support shared workspaces only, shared workspaces plus voice, and shared workspaces plus voice and video, and then observing differences in behavior for groups using the different systems.

While real-time collaboration is important and must be supported, asynchronous collaboration may turn out to be the more dominant activity. We suspect that the vast majority of actual work in a collaborative project is done by individuals working alone. We must communicate -- face-to-face or through communication channels -- to develop goals, share knowledge, plan, negotiate, help one-another with problems, etc. But we work alone to produce most of the text, code, or diagrams that comprise the material results of the group's collaborative efforts. Consequently, we will look closely at system functions that can help a group member understand the context for the portion of the project that individual is working on, and that support other similar activities that utilize the central artifact but are carried out by one person alone.

Finally, the system will assume different configurations to support specific studies, to test new features, and to evaluate the effectiveness of individual components. Above, we describe the MoDE interface tool we will use to define new modes, different combinations of functions within a mode, or different combinations of modes. Here, we note that change within a consistent architectural framework should be considered a basic characteristic of the system.

Summary of Expected Results

The principal result for this portion of our work will be a distributed collaboration system that supports building large conceptual structures, including applications such as the design of large software systems and accompanying documentation and diagrams. The system will serve as a testbed both for studies of groups working under different conditions and for developing and evaluating new functions for the application. But it must also be robust enough to be a system of choice for groups working in a research environment.

Relation to Other Parts of Project

The system is central to all parts of the project. It will evolve in its design in step with our evolving understanding of groups; thus, it is closely related to our efforts to develop a theory of collective cognition and group collaborative behavior. The system will be built using the tools described above; conversely, we could not achieve the degree of flexibility needed without adopting a system-building tools strategy. It will include protocol gathering functions that will produce most of the data used in our studies and in our efforts to build theory.

Protocol Tools and Methods

Concept and Strategy

Studying the cognitive activities of a single individual is a difficult, frequently subjective, and time-consuming task; studying the collective cognition of groups is more so. Consequently, developing new tools with which to observe and characterize groups is extremely important for our research and for others studying collaboration and cooperative work.

Methodological Issues

The most common form of data used to study complex cognitive processes as well as human-computer interaction have been concurrent protocols: either think-aloud protocols or keystroke records of user sessions.

A critical problem for studying cognitive behavior of any kind is gaining access to valid and sufficient data. Researchers frequently ask subjects engaged in a task to narrate their mental processes or "think-aloud" while they work [Newell & Simon, 1973]. These data have provided rich materials for studying a number of mental skills. However, they pose significant theoretical and practical problems for individuals, and this approach is probably unworkable for groups. For studying individual subjects, methodological issues center on the validity and completeness of the data and possible distortions in subjects' thinking introduced by the think-aloud procedure [Nisbett & Wilson, 1977; Ericsson & Simon, 1983]. Since having the individual members of a group all think-aloud while they work cooperatively is probably impractical, if not impossible, we won't review theoretical issues further, except to note that for cognitive activities in which spatial and/or abstract thinking play an important role -- such as would be expected for users of graphics-based computer systems -- both sides of the debate agree that thinking aloud should be expected to distort task performance and provide incomplete data. Thus, we conclude that thinking aloud is an inappropriate method of gathering data for groups.

An attractive alternative for studying subjects working with a computer system is to have the system record the users' actions while they work. This is most frequently done by having the system record each keystroke performed by the user. The problem with this approach is, first, that it may not record important spatial information for graphics-based systems, and, second, it produces a flood of very detailed data that is hard to analyze. That is, the analytic program must have the full interpretive capability of the user interface control program in order to parse the resulting sequence of keystrokes.

A third approach is to use video and/or audio recordings of subjects. While this method produces a very rich record of behavior, it is time-intensive and requires special training and controls for the human judges who code the protocols to produce reliable, consistent data. While these costs may be sustainable for selected meetings of a group, they are too high for this to be the primary method of collecting data for a group over a long period of time and for groups in which cooperative work takes place asynchronously as well as synchronously.

Tools for Studying Individuals

In previous research, described above, our group developed a number of new tools and methods for studying individual users. These tools support collecting machine-recorded protocol, managing large numbers of protocols, analyzing protocols with a cognitive grammar and other functions, and displaying results in both static and animated forms. They will form the basis for a new set of tools we will develop to study groups.

Tools for Studying Groups

One of the most important parts of our project will be the studies we will do of groups using our system for actual collaborative work. To carry out these studies, we must develop tools that can provide a comprehensive as well as detailed record of the groups' activities. New analytic and display functions will also be needed to support the studies and theory-building portions of the project.

The first issue is defining and gathering data that can be used to develop a theory of collective cognition, to evaluate specific computer and communications support functions, and to examine individual research questions. These data will be of two forms: machine-recorded and human coded. Since much of the collaborative work of the group will be done on-line using the system, we can build into that system a tracking function similar to that for the WE system, described above. Since the system will be networked, we will include in its design a central clock that will be updated regularly in each workstation. Consequently, actions performed by group members can be recorded in separate time-stamped protocol streams that can later be integrated to form a comprehensive, continuing group protocol. A second form of data will include coded representations of activities not conducted within the computer environment: minutes of meetings, notes of trained observers, etc. These data will be coded and individual actions associated with a particular time or duration; thus, they, too, can be integrated with the machine-recorded protocols to form a comprehensive, multi-strand group protocol We may also be able to coordinate video and/or audio recordings, but the methodology does not depend on this. Thus, we will develop tools for recording, integrating, and maintaining a rich body of data that can support a variety of analyses.

The second tool will be a replay function to recreate the group's activities, using the protocol as data. It will be similar to the animated display tool described above and will use similar techniques to integrate multiple protocol streams. The "display" will consist of multiple adjacent workstations with multiple windows in which the actions of different individuals will be displayed. This tool will permit us to replay sessions for an individual group member, combinations of members, or the entire group. We can also include off-line data in the form of coordinated text displays (or other appropriate forms), as also described for the animation tool. Observing the group's interactions for synchronous collaboration or their independent activities with respect to different parts of the central information artifact for asynchronous work will provide a fascinating, bird's eye view of the group.

The third set of tools will be analytic programs: a grammar to parse the integrated protocol for the group and statistical programs to analyze extracted data and distributions. The group grammar will be more complex than the one we developed for parsing individual protocols. It will include a more extensive set of mode, process, and product symbols, and it must infer the overall activities of the group as well as those of its individual members. Developing this grammar is a significant challenge. If we are successful, it will provide a powerful tool for extensive, detailed studies of groups during the grant period and beyond. If we do not complete the grammar during the project, we will base our studies on individual measures extracted directly from the integrated protocol data. Other analytic tools will consist primarily of standard statistical utilities. We currently use SPSS. We will develop support functions to extract statistical measures from both the underlying protocol data and from the parses produced by the grammar(s).

The fourth set of tools will be display tools. The protocol recording and integration tools will produce very rich data that can be analyzed by the grammar and statistical tools. But we --human researchers -- must *understand* these results if they are to be useful and if they are to lead to meaningful insights into collaborative behavior. The critical problem is to provide a perspective that is both general and richly detailed, but does not swamp the viewer. To get a sense of what such tools might show us, consider the following, (over)simplified portrait. For ease of description, we assume that a project produces a "final" product. Of course, many systems/products never achieve stability; the portrait applies to them if one takes as "stable" any snapshot of the product during its development.

At the end of a project, the group will have produced a large, integrated product -- the hypergraph artifact. It has discernable components (virtual documents, such as requirements or specifications; code; and diagrams). Each of these components has a form or shape -- some are deep hierarchies, others broad, still others are graph structures. And these components and their elements have various links that connect one-another. The *shape* of this artifact incorporates important information about the abstraction it realizes -- some shapes indicate more independence of components than others -- and, hence, is important from the standpoints of software methodology, training, system maintenance, etc.

Now, envision not this static form but its evolution over the history of the project up to this point. It frequently begins with an amorphous concept -- or a previous system, a problem, a work-order, etc. -- that evolves over time into the stable artifact. During this development, it might take the form of a "river" of multiple strands that converge and diverge with the mainstream. At some time during this history, we might see the entire group working on one part of the data object, as they work out the basic design or agreed upon a set of goals during a meeting. Or we might see individuals working on different parts during asynchronous collaboration. We might see strands diverge, grow individually, and then recombine with the central artifact as private work is incorporated into the principal design.

We can see this evolving shape from the abstract perspective of the mind's eye; we must now build display tools that let us see its specific form with the physical eye. Our goal, then, for this portion of our work is to develop display tools that can provide both general, comprehensive views of this evolution as well as closeup views for portions. They will include both static and animated forms. They will be integrated and coordinated to provide multiple, linked perspectives. While many will be based on tools we have built for displaying individual protocols, new ones will be needed, as well.

In the history of the artifact, in its evolving shape are the stuff of comprehension. If we can see these things, then we will begin to see collaboration and we will begin to understand it.

Summary of Expected Results

- o protocol tracker internal to each workstation, but synchronized for the system
- o protocol management tools
- o replay tools for recreating individual sessions as well as group and subgroup interactions
- o grammar and other functions to parse/analyze protocols and data extracted from them
- o display functions

Relation to Other Parts of Project

This portion of the project is integral to the entire project. The data gathering, analysis, and display tools will be used in our studies of specific issues, in the development and testing of a theory of collaborative work, and in evaluating specific system features and/or combinations of features. Since the tools, themselves, will be integrated into the system and communication system, they are an integral part of our system development effort. Consequently, our system-building tools, described above, will include in their requirements the capability to support protocol-related functions.

Studies

Goals and Strategy

Over the next three years, we will conduct four kinds of studies. First, we will observe actual working groups throughout the project. We believe that immersing ourselves in close, detailed ethnographic studies is essential. Second, we will address specific questions concerning collaborative strategies and patterns of behavior for groups. Third, we will conduct experiments to evaluate specific system features and configurations of features. Finally, we will conduct an extended study under naturalistic conditions to evaluate the overall effectiveness of the testbed system. The knowledge we gain from these studies will contribute to our on-going effort to build a theory of collective cognition and collaboration.

These studies will use several different kinds of users as subjects. First, we will use ourselves as subjects for pilot studies. We are a multidisciplinary collaborative team of approximately the size and kind our studies and systems are aimed at, building several different conceptual structures -- e.g., system, theory, interpretations of large bodies of data. Self-study will also give us first-hand knowledge of what we are asking other groups to experience. Second, we will observe and use as subjects other groups within the department. These include students enrolled in a software engineering class as well as other research groups. The former is particularly attractive since we can establish semi-controlled conditions in which groups are receiving the same instruction in methods, are working on projects of comparable scope, for the same period of time. Third, we will observe other scientific and technical working groups outside the department. One such group will be a multisite team working under the auspices of MCNC and supported in its work by the MCNC video network.

Specific Studies

Year 1

Our studies for the first year will adopt two primary goals: to develop, test, and refine a methodology for observing and characterizing groups; and, second, to conduct a set of pilot studies of groups using conventional computer/communication tools to serve as a basis of comparison with later groups using the testbed system. These studies will also inform the design of specific interface features and user functions.

In the section on theory, we outlined a concept of modes of activity and strategies for their use. That construct can provide the basic framework in which to build a theory of collaborative work and collective cognition. A first step toward doing this will be to develop specific guidelines that can focus ethnographic observations of groups. We will watch actual groups, focusing on the artifacts they are developing at any one time -- both tangible artifacts, such as specifications or notes of a meeting, but also intangible artifacts, such as a body of shared knowledge or a common understanding of goals. From these initial studies, we will develop tentative descriptions of specific modes of activity; we will then use those descriptions to guide subsequent observations. We will be particularly conscious of factors that a trained observer can see to be operating in the group but which are not represented in the guidelines. Cycles of observation, inference, specification, testing, and refinement will be repeated throughout the project. We will also look closely at the snags or problems that cause shifts in group modes and/or new collaborative techniques developed by the group. These latter observations will help us focus on strategies and on new forms of mediation developed by the group. At the end of the project, this line of research will produce tested and refined guidelines that can be widely applied to technical groups working in a variety of contexts.

We will observe three groups during the first year: ourselves, another research group within the department, and an MCNC group. In observing ourselves, we stress that our theory and studies team will include two cognitive psychologists, an anthropologist, a computer scientist, and two graduate students -- one from each social science area. The students will be trained as observers and will observe the system and tools building teams as well as the overall group. Each study will last for approximately three months and will result in a detailed set of notes, summary reports, and other publications. These studies will also produce a set of recommendations for system functions that would have facilitated the work of the groups they observed.

Year 2

During the second year, the focus of our studies will shift from groups working with conventional tools to those working with the testbed system. We will first do a pilot study on ourselves as we use the system (we can't ask others to use a system we do not use ourselves.) This study will test our protocol recording tools; it will further test our revised guidelines for observing groups; and it will test our techniques for integrating external data with the machine-recorded protocols. We will also make a first attempt to characterize the pattern of activities of a group over an extended period of time, based on these integrated data.

After revising our methodological tools and techniques, as needed, we will conduct a second series of exploratory studies. These studies will be of 2-3 groups within the department who volunteer to use the system on a small software development project. Goals for these studies, in addition to further testing of methods, will be to identify distinguishing patterns of behavior for the groups. Again, these studies will be written up to provide detailed notes and descriptions as well as summary and interpretive papers.

During this second year, we will also begin a series of small, focused studies and experiments to evaluate specific system features, such as screen layout, combinations of user functions, etc. These studies will continue for the remainder of the project.

Year 3

During the third year, we will conduct two rather large studies. In the first, we will evaluate the relative effectiveness of adding video and/or voice communication along with shared visual workspaces We will configure the testbed system so that we may observe groups using only shared visual workspaces to support synchronous collaboration, shared workspaces plus voice, and shared workspaces plus voice plus video. Groups will be drawn from MCNC and from the department. We will observe groups working both from individual offices that provide individual video transmission as well as from teleconferencing rooms that include workstations. Among the questions we will address are the following:

- if video is available, will it be used?
- what are the incremental costs and the incremental benefits of voice and voice plus video?
- how do these additional channels affect users' strategies and behavior?
- what are the differences in providing video communication from workstations in individual offices vs. from a teleconferencing room equipped with workstations?

- o what tasks do groups use video? for what tasks do groups use face-to-face meetings?
- o what tasks are done asynchronously?

In a second study, we will evaluate the overall effectiveness of the system. We will create a semi-controlled experiment using volunteer teams from a software engineering class: half of the volunteer groups will use our system, half will use conventional tools. We will observe meetings for all groups and record protocols of system use. Groups using our system will be able to discontinue using it and switch to conventional tools if they feel it interferes with work on their project. A group of judges will evaluate systems and materials developed by the teams. We will measure effectiveness by comparing the evaluation each team receives; this measure will not be definitive, but it will give us a useful assessment of our efforts. The data we collect will be quite rich and can be analyzed in a number of different ways. For example, we will look to see if groups that adopt a more integrative vs. distributed strategy produce better systems but at greater effort, as reported by [Bendifallah & Scacchi, 1989].

These studies, we believe, will just scratch the surface of the capabilities these tools, methods, and systems can provide and of the issues that can be addressed. If time permits, we will consider other questions that are sure to come up during our work. But regardless, we will continue to use these resources beyond the duration of the project in an on-going program of research to which the department is committed.

Summary of Expected Results

- o Guidelines for observing groups, tested and refined
- o Detailed descriptions, analyses, and characterizations of some 10-12 working groups
- o Four specific studies

Relation to Other Parts of Project

This portion of the project is closely related to all other parts. It will use the guidelines described earlier, and its results will be synthesized and refined to build a theory of collective cognition and collaboration. As we evaluate individual user functions and behaviors, we will use the system building tools to adjust the user interface and other segments of the system. The system, itself, will serve as the testbed for most of the studies for years 2 & 3. As the results of these studies are generalized and incorporated into theory, the cycle closes and repeats.

Summary of Expected Results

We are in the early stages of this project. We cannot be sure which of our planned acctivities will truly be useful and which will have to be modified. However, we summarize here what we anticipate will be our most important results.

- o Guidelines for observing groups engaged in collaborative work
- o Detailed descriptions, analyses, and characterizations of 10-12 working groups
- o Tools for building and supporting collaborative applications:
  - -- General graph and subgraph service
  - -- Interface builder for collaborative applications
  - -- Integral voice and video
  - -- Communications media and infrastructure evaluation
- o An advanced distributed hypermedia system that supports both synchronous and asynchronous collaboration for task that involve building large conceptual structures, with emphasis on software design
- o Tools for recording, managing, analyzing, and displaying group protocols
- o Studies of specific collaborative issues
- o Theory of collective cognition and collaboration for groups

REFERENCES

Abdel-Wahab, H. M., Guan, S.-U., & Nievergelt, J., (1988). Shared Workspaces for Group Collaboration: An Experiment using Internet and UNIX Interprocess Communications. *IEEE Communications,* 26(11): 10-16.

ACM, (1988). *Proceedings of the Second Conference on Computer-Supported Cooperative Work,* Portland, Oregon, Sept. 26-28, 1988.

Alderfer, C.P., Smith, K.K. (1982). Studying intergroup relations embedded in organizations. *Adm. Sco. Q.* 27:35-65.

Ancona, D.G. (1987). Groups in organizations: Extending laboratory models. See Hendrick 1987, 9:207-30.

Begeman, M., Cook, P., Ellis, C., Graf, M., Rein, G. and Smith, T. (1986). Project NICK: Meetings augmentation and analysis. *Proceedings of the Conference on Computer Supported Cooperative Work.*

Bendifallah, S., & Scacchi, W. (1989). Work structures and shifts: An empirical analysis of software specification teamwork. *Proceedings of the 11th International Conference on Software Engineering,* Pittsburgh, May, 1989.

Bereiter, C., & Scardamalia, M. (1987). *The psychology of written communication.* Hillsdale, NJ: Erlbaum.

Bigelow, J., & Riley, V. (1987). Manipulating source code in Dynamic Design. *Proceedings of Hypertext '87,* pp. 397-408.

Biggerstaff, T., Ellis, C., Halasz, F., Kellog, C., Richter, C., & Webster, D. (1987). *Information Management Challenges in the Software Design Process.* MCC Technical Report # STP-039-87.

Boehm, B.W., (1981). *Software engineering economics.* Engelwood Cliffs, NJ: Prentice-Hall, Inc.

Brandes, T., & Lewerentz, C., (1985). GRAS: A Non-standard Data Base System within a Software Development Environment, *Proceedings GTE Workshop on Software Engineering Environments for Programming-in-the-Large,* pp. 113-121.

Brandstatter, H., Davis, J.H., Stocker-Kreichgauer, G., eds. (1982). *Group Decision Making.* London: Academic Press.

Brooks, F. P., Jr., (1987). No Silver Bullet -- Essence and Accidents of Software Engineering. *Computer* 20(4): 10-19.

Brooks, F. P., Jr., (1975). *The mythical man-month.* Reading, MA: Addison-Wesley.

Calingaert, P., Abdel-Wahab, H. M., Bollella, G., Guan, S.-U., Smith, F. D., & Smith, J. B., (1990). Synchronous and Asynchronous Collaboration: A Program of Research, submitted to *The Third Conference on Computer-Supported Cooperative Work,* Oct. 1990.

Campbell, B. & Goodman, J. M., (1987). HAM: A General-purpose Hypertext Abstract Machine, *Proceedings Hypertext '87,* pp. 21-32.

Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human-computer interaction.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Claparede, E. (1933). La genese de l'hypothese. *Archives de Psychlogie*, 24, 1-155.

Clark, N.K. Stephenson, G.M. (1989). Group remembering. See Paulus 1989, pp. 357-91.

Clark, R.D. III, Maass, A. (1988). Social categorization in minority influence: The case of homosexuality. *Eur. J. Soc. Psychol.* 18:347-64.

Collins, P. D., & King, D. C. (1988). Implications of computer-aided design for work and performance. *Journal of Applied Behavioral Sciences* 24/2,173-190.

Computer Science and Technology Board (1990). Scaling Up: A Research Agenda for Software Engineering, *CACM,* 33:3 (March) 281-293.

Conklin, J., & Begeman, M. L., (1988). gIBIS: A Hypertext Tool for Exploratory Policy Discussion, *ACM Transactions on Office Information Systems*, Vol. 6, No. 4, 303-331.

Cummings, L.L., Staw, B.M., eds. (1985). *Research in Organizational Behavior*, Vol. 7, Greenwich: JAI Press.

Curtis, B., Krasner, H., & Iscoe, N. (1988). A field study of the software design process for large systems. *CACM* 31:11 (November), 1268-1287.

Daiute, C. (1985). Issues in using the computer to socialize the writing process. *Educational Communication and Technology* 33/1 (Spring), 41-50.

Delisle, N., & Schwartz, M. (1986). Neptune: A hypertext system for CAD applications. *ACM SIGMOD*, 15:2, 132-143.

DeSanctis, G., Gallupe, B. (1987). A foundation for the study of group decision support systems. *Mange. Sci.* 33:589-609.

Englebart, D. (1984). Collaboration Support Provisions in Augment, *Proceedings AFIPS Office Automation Conference*, pp. 51-58.

Enson, J.R., Ahuja, S.R., Horn, D.N.; & Lucco, S.E. (1988). The rapport multimedia conferencing system - a software overview. *Proceeding of the 2nd IEEE Conference on Computer Workstations*, Santa Clara, CA, March 7-10, 1988.

Ericsson, K. A. & Simon, A. S. (1980). Verbal reports as data. *Psychological Review*, 87, 215-251.

Flower, L. S., & Hayes, J. R. (1984). Images, plans, and prose: The representation of meaning in writing. *Written Communication,* 1, 120-160.

Foley, J. D. & Wallace, V. L. (1974). The art of natural graphic man-machine conversation. *Proceedings of the IEEE,* Vol. 62, pp. 462-471.

Foster, G. (1986). *Collaborative systems and multi-user interfaces.* Ph.D. Dissertation, Division of Computer Science, University of California, Berkeley.

Foster, G. & Stefik, M. (1986). Cognoter, theory and practice of a Colaborative tool. *Proceedings of the Conference on Computer Supported Cooperative Work,* I. Greif, Ed. Austin, TX, Dec 3-5, 1986.

Fowler, G. D. & Wackerbarth, M. E. (1980). Audio teleconferencing versus face-to-face conferencing: A synthesis of the literature. *The Western Journal of Speech Communication* 44, 236-252.

Gersick, C.J.G. (1988). Time and transition in work teams: Toward a new model of group development. *Acad. Manage.* J. 31:9-41.

Greif, I., (1988). *Computer-Supported Cooperative Work: A Book of Readings,* Morgan Kaufmann Publishing Co., Palo Alto, CA, 1988.

Greif, I., Ed. (1986). *Proceedings of the conference on computer supported cooperative work.* Austin, TX, Dec 3-5, 1986.

Guan, S.-U., (1989). *A Model, Architecture, and Operating-System Support for Shared Workspace Cooperation.* Ph.D. dissertation, University of North Carolina at Chapel Hill.

Guan, S.-U., Abdel-Wahab, H. M., & Calingaert, P., (1990). Operating System Support and Protection Model for Jointly-Owned Objects. Submitted to *The Journal of Systems and Software.*

Guzzo, R.A., ed. (1982). *Improving Group Decision Making in Organizations.* New York: Academic Press.

Hammond, K., (1988). Judgement and decision making in dynamic tasks. *Information and Decision Technologies* 14:3-14.

Hayes, J. R. & Flower, L. S. (1980). Identifying the organization of the writing process. In L. W. Gregg & E. R. Steinberg (Eds.), *Cognitive Processes in Writing.* Hillsdale, NJ: Lawrence Erlbaum Associates, pp. 3-30.

Hayes, J. R., & Flower, L. S. (1986). Writing research and the writer. *American Psychologist,* 41, 1106-1113.

Heimstra, G. (1982). Teleconferencing, concern for face, and organizational culture. In Burgoon, M. (Ed.), *Communication Yearbook 6.* Beverly Hills CA: Sage, pp. 874-904.

Hendrick, C., ed. (1987). *Review of Personality and Social Psychology,* Vols. 8, 9. Newbury Park: Sage.

Henry, L. K. (1934). The role of insight in the analytic thinking of adolescents. *Studies in Education,* 9, 65-102.

Hill, G.W. (1982). Groups versus individual performance: Are N + 1 heads better than one? *Psychol. Bull.* 91:517-39.

Holland, D. & Valsiner, J. (1987). Symbols, Cognition and Vygotsky's Developmental Psychology. Ethos 16(3).

Insko, C.A., Gilmore, R., Drenana, S., Lipsitz, A., Moehle, D., et al. (1983). Trade versus expropriation in open groups: A comparison of two types of social power. *J. Pers. Soc. Psychol.* 44:977-99.

Insko, C.A., Gilmore, R., Moehle, D., Lipsitz, A., Drenan, S., et al. (1982). Seniority in the generational transition of laboratory groups: The effects of social familiarity and task experience. *J. Exp. Soc Psychol.* 18:557-80.

Insko, C.A., Smith, R. H., Alicke, M.D., Wade, J., Taylor, S. (1985). Conformity and group size: The concern with being right and the concern with being liked. *Pers. Soc. Psychol. Bull.* 11:41-50.

Insko, C.A., Thibaut, J.W., Moehle, D., Wilson, M., Diamond, W.D., et al. (1980). Social evolution and the emergence of leadership. *J. Pers. Soc. Psychol.* 39: 431-48.

Jarrell, N. & Barrett, B. (1986). Network-Based systems for asynchronous group communications. Austin, Texas: *Proceedings of the Conference on Computer-Supported Cooperative Work*, pp. 184-191.

Katz, R. (1982). The effects of group longevity on project communications and performance. *Adm. Sci. Q.* 27:81-104.

Kenny, D.A. (1985). The generalized group effect model. In *Individual Development and Social Change: Exploratory Analysis*, ed. J. Nesselroode, A. von Eye, pp. 343-57. Orlando: Academic Press.

Kerr, N.L.. MacCoun, R.J., Hansen, C.H., Hymes, J.A. (1987). Gaining and losing social support: Momentum in decision-making groups. *J. Exp. Soc. Psychol.* 23:119-45.

Kieras, D. E. (1980). Initial mention as a signal to thematic content in technical passages. *Memory and Cognition*, 8, 345-353.

Kieras, D. & Polson, P. G. (1985). An approach to the formal analysis of user complexity. *International Journal of Man-Machine Studies,* 22, 365-394.

Kiesler, S., Siegel, J., McGuire, T.W. (1984). Social psychological aspects of computer-mediated communication. *Am. Psychol.* 39:1123-34.

Kintsch, W., & van Dijk, T. A. (1978). Toward a model of text comprehension and production. *Psychological Review,* 85, 363-394.

Kozulin, A. (1986). Vygotsky in Context. In Thought and Language, by Lev Vygotsky, translated by Alex Kozulin. Cambridge, MA: *The MIT Press.* pp. xi-lvi.

Kraemer, K.L., & King, L.K. (1988). Computer-Based systems for cooperative work and group decision making. *Computing Surveys*, 20/2, 115-146.

Lai, K.-Y., Malone, T. W, Yu, K.-C., (1988) . Object Lens: A "Spreadsheet" for Cooperative Work, *ACM Transactions on Office Information Systems*, Vol. 6, No. 4 (October), 332-353.

Lantz, K. A., (1986). An Experiment in Integrated Multimedia. *Proceedings of Conference on Computer-Supported Cooperative Work*: pp. 267-275.

Lantz, K. A. (1986). An experiment in integrated multimedia conferencing Austin, Texas: *Proceedings of the Conference on Computer-Supported Cooperative Work*, pp. 267-275.

Laughlin, P.R., McGlynn, R.P. (1986). Collective induction: Mutual group and individual influence by exchange of hypotheses and evidence. *J. Exp. Soc. Psychol.* 22:567-89.

Lederberg, J. & Uncapher, K., (eds.), (1989). Towards a National Collaboratory: Report of an Invitational Workshop at the Rockefeller University, March 17-18, 1989. *Distributed by the National Science Foundation.*

Lee, B. (1985). Intellectual origins of Vygotsky's semiotic analysis. In *Culture, Communication, and Cognition*: Vygotskian perspectives, James V. Wertsch, ed. New York: Cambridge University Press. pp. 66-93.

Levine, J.M, Russo, E.M. (1987). Majority and minority influence. See Hendrick 1987, 8:13-54.

Malone, T. W, Grant, K. R., Lai, K. Y., Rao, R., & Rosenblitt, D., (1987). Intelligent Information Sharing Systems, *CACM*, Vol. 30, No. 5, pp. 390-402, May 1987.

Meyer, G. J. F., Brandt, D. M., & Bluth, G. J. (1980). Use of top-level structure in text: key for reading comprehension of ninth grade students. *Reading Research Quarterly*, 1, 72-103.

Minick, N. (1987). The Development of Vygotsky's Thought: An Introduction. In *The Collected Works of L. S. Vygotsky*. Volume 1: Problems of General Psychology, Norris Minick, trans. New York: Plenum Press, pp. 17-36.

Moreland, R.L. (1987). The formation of small groups. See Hendrick 1987, 8:80-109.

Nagl, M. (1986). A software development environment based on graph technology. *Proceedings of 3rd International Workshop on Graph-Grammars and their Application to Computer Science*, pp. 458-478.

Nemeth, C. (1986). Differential contributions of majority and minority influence. *Psychol. Rev.* 93:23-32.

Nemeth, C., & Brilmayer, A.G. (1987). Negotiation versus influence. *Eur. J. Soc. Psychol.* 17:45-56.

Newell, A., & Simon, H.A. (1973). *Human Problem Solving.* Englewood Cliffs, NJ: Prentice-Hall.

Nisbett, R. E. & Wilson, T. D. (1977). Telling more than we can know: Verbal reports on mental processes. *Psychological Review,* 84, 231-259.

Reisner, P. (1981). Formal grammar and human factors design of an interactive graphics system. *IEEE Transactions on Software Engineering,* 7(2), 229-240.

Rice, R. E. (1984). *The New Media: Communication, Research and Technology.* Beverly Hills CA: Sage.

Rousseau, D.M. (1985). Issues of level in organizational research: Multi-level and cross-level perspectives. See Cummings & Staw 1985, pp. 1-37.

Sarin, S. K. (1984). *Interactive On-Line Conference,* Ph.D. Thesis, MIT, Also available as Laboratory for Computer Science Technical Report MIT/LCS/TR-330.

Sarin, S. K., & Greif, I. (1985). Computer-Based Real-Time Conferences. *IEEE Computer* 18/10 (October), 33-45.

Scacchi, W. (1988). The USC System Factory Project. *Proceedings of Software Symposium '88,* pp. 9-41.

Scheifler, R. W., & Gettys, J., (1986). The X Window System, *ACM Transactions on Graphics,* Vol. 5, No. 2, pp. 79-109.

Schwartz, M. N. K., & Flammer, A. (1981). Text structure and title-effects on comprehension and recall. *Journal of Verbal Learning and Verbal Behavior,* 20, 61-66.

Scribner, S. (1984). Cognitive Studies of Work. Special Issue of *Quarterly Newsletter* of the Laboratory of Comparative Human Cognition. 6(1-2).

Selfe, C. L., & Wahlstrom, B. J. (1985). *An emerging Rhetoric of Collaboration: Computers, Collaboration, and the Composing Process.* Available through the ERIC Clearinghouse.

Seliger, R., (1985) *The Design and Implementation of a Distributed Program for Collaborative Editing.* Master's Thesis, MIT, Also available as Laboratory for Computer Science Technical Report MIT/LCS/TR-350.

Shan, Y.-P., Thorn, J. & Rooks, M. C. (1988). *WE 1.0 User's Manual.* UNC Department of Computer Science Technical Report # TR88-036.

Shan, Y.-P., (1990). MoDE (Mode Development Environment). To appear in *IEEE Software* 7(3), May 1990.

Shan, Y.-P., (1989). An Event-Driven Model-View-Controller Framework for Smalltalk. *Proceedings of ACM Fourth Annual Conference on Object-Oriented Programming, Systems, and Applications* (OOPSLA), pp. 347-352.

Short, J., Williams, E., Christie, B. (1976) *The social psychology of telecommunications*. New York: John Wiley.

Smith, J. B. & Lansman, M. (1989). A cognitive basis for a computer writing environment. In Britton, B. K. & Glynn, S. M. (Eds.) *Computer writing aids: theory, research, and practice,* Also published as UNC Department of Computer Science Technical Report # TR87-032.

Smith, J. B. & Smith, C. F. (1987a). *A strategic method for writing.* Chapel Hill, NC: UNC Department of Computer Science Technical Report # TR87-024.

Smith, J. B., Weiss, S. F., Ferguson, G. J., Bolter, J. D., Lansman, M., & Beard, D. V. (1987b). WE: A Writing Environment for Professionals. *Proceedings, National Computer Conference '87*: 725-736.

Smith, J. B., Weiss, S. F., Ferguson, G. J., Bolter, J. D., Lansman, M., Beard, D. V. (1986). *We: A Writing Environment for Professionals.* Chapel Hill, NC: Department of Computer Science Technical Report # TR86-025.

Smith. J. B., Weiss, S. F., & Ferguson, G. J., (1987). A Hypertext writing environment and its cognitive basis, *Hypertext '87 Proceedings.* New York: ACM Press, pp 195-214. Also published as UNC Department of Computer Science Technical Report # TR87-033.

Stefik, M. (1986). Cognoter, theory and practice of a collaborative tool. In *Proceedings of CSCW* '86, pp. 7-15.

Stefik, M., Bobrow, D. G., Lanning, S., Tatar, S. & Foster, G. (1986). WYSIWIS revised: Early Experience with Multi-user Interfaces. Austin, Texas: *Proceedings of the Conference on Computer-Supported Cooperative Work*, pp. 276-290.

Stefik, M., Foster, G., Bobrow, D. G., Kahn, K., Lanning, S., Suchman, L. (1987). Beyond the Chalkboard: Computer support for collaboration and problem solving in meetings. *Communications of the ACM* , 30/1 (January) 32-47.

Swarts, H. Flower, L. S., & Hayes, J.R. (1984). Designing protocol studies of the writing process: An introduction. In R. Beach & Bridwell (Eds.), *New Directions in Composition Research*. New York: Guilford Press, pp. 53-71.

Taylor, R. N., Baker, D. A., Belz, F. C., Boehm, B. W., Clarke, L. A., Fisher, D. A., Osterweil, L., Selby, R. W., Wileden, J. C., Wolf, A. L., & Young, M. (1987). Next generation software environments: Principles, Problems, and Research Directions. *COINS Technical Report #87-63.*

Vygotsky, L. S. (1962). *Thought and language.* Cambridge, MA: MIT Press.

Vygotsky, L. S. (1978). *Mind in society.* Cambridge, MA: MIT Press.

Vygotsky, L. S. (1987). *The collected works of L. S. Vygotsky. Volume 1: Problems of General Psychology.* New York: Plenum.

Wegner, D.M., Guiliano, T., Hertel, P. (1985). Cognitive interdependence in close relationships. In *Compatible and incompatible relationships,* ed. W.J. Ickes, pp. 253-76. New York: Springer-Verlag.

Wertsch, J. V. (Ed.) (1985) *Culture, Communication, and Cognition: Vygotskian perspectives.* New York: Cambridge University Press.

Williams, E. (1978). Social and psychological factors *Journal of Communication* 28, 125-131.

Williams, J. P., Taylor, M. B., & Ganger, S. (1981). Text variations at the level of the individual sentence and the comprehension of simple expository paragraphs. *Journal of Educational Psychology,* 73, 851-865.