Issues in the Design of Computer Support for Co-authoring and Commenting

Christine M. Neuwirth
David S. Kaufer
Ravinder Chandhok
James H. Morris

Carnegie Mellon University
Pittsburgh, PA 15213

ABSTRACT

This paper reports on a project to develop a "work in preparation" editor, or PREP editor, to study co-authoring and commenting relationships. As part of the project, we have identified three issues in designing computer support for co-authoring and commenting: (1) support for social interaction among co-authors and commenters; (2) support for cognitive aspects of co-authoring and external commenting; and (3) support for practicality in both types of interaction. For each of these issues, the paper describes the approach the PREP editor takes to address them.

GOALS OF THE RESEARCH

The goal of this project is to develop a "work in preparation" (PREP) editor, a multi-user environment to support a variety of collaborative and, in particular, co-authoring and commenting relationships for scholarly communication. In our research, we do not focus on collaborations in which co-authors or commenters interact at the same time, though systems that support research into the issues such collaborations raise are clearly valuable [Stef87]. Our focus is on enhancing the effectiveness of loosely-coupled collaboration. We focus on co-authoring because it represents an interesting challenge for collaborative work over networks: co-authors, after all, must share a planning environment that often relies on, but is nonetheless richer than a working draft. We focus on commenting because it poses a challenge for communication within authoring groups as well as between external readers and such groups. We focus on scholarly communication because scholarly communities as they exist today are already collaborative work groups. They are not explicitly organized around single, concrete goals, but members of groups share the common goal of advancing the state of knowledge. Such work groups are organized in local settings, but they also interact intensively at a distance, as members of a common "invisible college" [Cran72].

ISSUES IN SUPPORTING CO-AUTHORING AND COMMENTING

The PREP editor we are developing addresses three issues: (1) support for social interaction among co-authors and commenters; (2) support for cognitive aspects of co-authoring and external commenting; and (3) support for practicality in both types of interaction.

Issue 1: Support for the social aspects of collaboration

Generally speaking, we know very little about the social aspects of long-term collaborative writing relationships. Several researchers have conducted observational studies and developed initial frameworks to help define a set of requirements for supporting collaborative writing groups [Gere87; Krau87; Luns90]. Despite limited knowledge about social aspects of collaboration generally, one of the social aspects of collaborative work that we do understand to some extent involves problems of coordination. Coordination issues arise in answer to the question "What activities do writers need to perform by virtue of the fact that they are working together rather than alone?" [Malo88].

Support for the definition of social roles

One response to the coordination problem is to support the definition of social roles: Defining roles reduces the coordination problem by specifying "proper functions" (e.g., responsibilities and patterns of interaction) of the various collaborators. The Quilt system exploits this strategy [Fish88; Lela88]. In Quilt, three social roles are defined (co-author, commenter and reader) as well as six objects (base document, suggested revision, public comment, directed message, private comment, history) and a set of actions (create, modify, delete, attach a suggested revision, attach a public comment, attach a directed message, attach private message, and read). It is then possible in Quilt to specify which roles can perform which actions on the various objects. Quilt further defines three collaboration types: *exclusive*, in which only the author of a section in the base document can modify it; *shared*, in which any author can modify any section; and *editor*, in which a designated editor can modify any section and other authors may only make suggested revisions. In addition, Quilt supports user-defined types of collaboration.

Providing user-defined types is a crucial feature of Quilt. Without it, defining permissible actions on objects in terms of abstractions such as "author" and "commenter" would be problematic. This is because the systemic features of these roles vary in different contexts [Kauf*ms*]. For example, the APA [APA83] defines authors to be "not only those who do the actual writing but also those who have made substantial scientific contributions to a study." Thus, it is possible to imagine a collaboration in which the abstract roles and collaboration types predefined by Quilt would be at odds with the wider social meanings of those abstractions in the particular context. For example, a group of three authors may want to define permissions so that two of them--the ones actually doing the writing--can change the base document. The third, let us say, will not change the base, but will simply comment on the piece for accuracy. No predefined combination of role/collaboration type would suffice.

Our experience supporting user-annotations in Comments, a hypertext tool developed in order to study computer support for response to writing [Neuw88], also affords some insight into other crucial features involved in the definition of social roles, especially the interaction of social roles with writing processes.[1] For example, we have found that, regardless of their social role (either as co-author or commenter), some commenters want

---

[1] Users were students at Carnegie Mellon University enrolled from Fall, 1987 to Spring ,1989 in computer-based tools sections of a freshmen writing course and their teachers. We conducted user tests and interviewed users about their use of the program. We also conducted more formal evaluations [Hart*ms*]. The Comments program also has a menu option that allowed users to report problems to us via a compus-wide network and we maintained a campus-wide bulletin board where users could discuss reactions, problems, and so forth. Some users were paid for their participation in our studies. Users' previous computer experience ranged from experienced to hardly any experience.

the ability to rewrite the written text and not simply attach annotations.[2] This phenomenon may be due to the fact that many significant problems in texts (e.g., voice, persuasiveness, organization), though easy for an experienced writer to detect, cannot be easily described. For such problems, rewriting is often a more efficient strategy than trying to diagnose the problem, and writers often choose this strategy when revising others' texts [Haye87]. In any case, writers in the role of commenters often copied a region of the base document into a commenting box and proceeded to rewrite the copy. Writers who worked in this fashion, however, reported difficulties in revising because their revisions were physically separated from the larger body of text. More specifically, they reported needing a "sense of the whole text" even when commenting on a part. One exasperated commenter went so far as to copy an *entire* document into a comment box and to revise it from there. Whether a commenter is able to modify the base document or not should certainly depend on his or her rightful relationship (co-author, commenter) to the text. An optimal design, however, would not collapse cognitive needs and social roles, but give commenters the ability to rewrite his or her *view* of the text and deal with the effects of this revision on the original base document independently.

There is a potential problem in systems which support the definition of social roles: "premature" definitions of these roles could lead to undesirable consequences. For example, it is not always clear at the outset of a project who is going to make a "significant contribution" and therefore who should get authorship. But if authorship is defined at the outset, then it may reduce the motivation of someone who has been defined as a "non-author" and the person may not contribute as much. Just as we need more research into the social aspects of writing, we need more research with writers actually using systems that define social roles.

Despite potential problems, role specification is likely to be a useful strategy for managing some coordination problems; however, roles such as "co-author" and "commenter" substantially underspecify the activities involved in coordinating complex tasks such as collaborative writing.[3] Writers also need support for coordination activities that fall outside role boundaries. Discussing the full range of these activities is beyond the scope of this paper, but we will briefly sketch two that are especially acute in writing tasks: support for communication about plans and support for communication about comments.

Support for communication about plans

Talk about dividing the labor of writing is likely to include plans for the paper [Krau87]. Writers, however, to not "execute" plans in the same sense that programs do. Instead, writers use a plan as a resource in deciding what to do while they are writing [cf. Agre89]. Often, the partially completed product plays an important role in this process: The partially completed product becomes part of the task environment and constrains the subsequent course of the design [Flow81; Kauf86]. In addition, writers set new goals for themselves as they discover what it is they want to say [Haye80]. When writers work alone, they may not need to articulate the constraints that they have imposed and the new goals they have set. Not surprisingly, co-authors often need to communicate about the constraints in order to refine their views of the goals that co-authors have generated and increase the likelihood that they will generate compatible products. In addition, communication about evolving plans and constraints may improve performance by saving co-authors and commenters from having to infer the other's plans. For example, here is

---

[2] Unlike Quilt, the Comments program does not define social roles explicitly, but like Quilt, collaborators can be granted or denied permission to modify the base document.

[3] Of course, it is possible to define more specific roles. For example, we have been experimenting with Devil's Advocacy, among others [Neuw89a].

an excerpt from a co-author we recently observed communicating about constraints in a paper entitled, *Structure Editors: Evolving Towards Appropriate Use*:

> *In the title I want to stress that we have spent time discovering, sometimes the hard way, when structure editors are useful.*

If a co-author understands the goal, he or she is more likely to be able to produce revisions to the title that are compatible with the other author's goal.

Our approach accepts that plans do not control writing, indeed, that plans will not be made completely in advance of writing and concentrates instead on supporting communication about plans.

Support for the communication about comments

The problems with comments, that is, critical notes on texts, are well-known and legion: writers don't understand comments, they think the comments reflect confused readings rather than problems in their texts, they are frustrated by perceived lack of consistency in comments and contradictory comments [Neuw88]. The problems in author/commenter relationships become even more pressing if authors solicit comments from multiple readers. The interesting empirical question for the author/commenter relationship is how best to help the authoring group manage and make good use of the comments coming in?

Our approach to comments acknowledges that confusions and difficulties abound in the communication among commenters. By providing a system that facilitates communication about comments, we hope to provide one promoting more helpful interactions.

Issue 2: Support for the cognitive aspects of collaborative writing

Supporting the cognitive aspects of collaborative writing involves two things: task specific support for cognitive activities of writing *per se*, and support for the cognitive activities involved in collaborative writing.

Task specific support for cognitive activities of writing

There have been some attempts to understand the task-specific activities (e.g., jotting, drawing, writing, gesturing) that occur in collaborative tasks in order to inform the design of specialized tools to support those tasks [Cook87; Stef87; Tang88]. But because there is a tendency to equate the substantive work of writing with a written draft, most text annotators support only communication about the working draft or outlines of a draft. Experienced writers, however, typically produce intermediate external representations that have no direct relation to the text product [Flow89]. When working with environments that do not support the the creation of arrows, boxes, or other diagrams for displaying conceptual relationships among ideas and the suppression of detail, writers report frustration [Brid87] and important planning activity is curtailed [Haas89].

Our experience with the Comments program also indicates that the written draft is an essential though in many respects incomplete representation for supporting communication about writing. For example, many of the writers we have observed using our tools resort to paper to produce intermediate representations, such as plans for drafts, two-dimensional grids depicting similarities and differences across sources, and trees depicting structural characteristics of an outline or draft. These writers typically like to create (and discard) such intermediate representations quickly, often relying on

them only as temporary "sketches" "doodles" or "scribblings" for getting their bearings or adjusting their bearings with those of their co-authors. They further report that drawing editors are too slow to serve these "coordinative" functions and consequently they do not use them for these purposes. So they turn to hardcopy and pass hardcopy drafts to one another for review. The result is that much of their significant planning as co-authors is not done on-line.

Thus, research in writing processes as well as our own observations of writers working with annotation tools suggest that cognitive issues such as supporting the jotting, drawing and note-taking that writers engage in as they write are especially important in writing and that cognitive aspects must be taken into account when designing computer support for co-authoring and commenting.

External commenters differ from co-authors in that they are normally not privy to the informal planning leading up to the working draft. Nor can they claim ownership of the working draft or any future draft. Nonetheless, tools to support more of the writing process than just drafts allow us to define a host of potential new relationships for the external commenter in relation to an authoring group. An important goal of the current project is build a set of tools that will help us explore the interesting roles that external commenters can play vis-a-vis co-authors. Among the leading candidates for interesting roles are the following:

Allowing commenter access to planning objects

Suppose that co-authors use computer tools not only to write their working drafts, but to plan them as well. Then much of their planning environment could be recoverable and open to the scrutiny of external readers. Will external readers have any incentive to explore these planning environments as a context for commenting on a working draft? If so, then the boundaries between original co-author and external reader may grow less distinct. For us, it is an interesting empirical question to consider how much information about the co-authoring interaction the external commenter will seek when computer tools make larger amounts of that interaction available for inspection. The tools we build will allow studies that examine commenting situations of this type and observe what happens to the author/commenter relationship.

Allowing commenters to perform "authoring tests"

External commenters are not likely to involve themselves in the planning environment of the co-authors beyond that which is revealed in the working draft. External commenters, however, have been known to interrogate a text in some of the ways that co-authors expect one another to. Some journal editors, for example, advise their editorial boards to try to "outline" a text submitted for publication if it seems unclear. Doubtless, external commentators keep in mind some of these interrogation tests as they read. Whether they actually perform them and the extent to which they perform them is another matter. But if computer tools can make it easier for co-authors to perform these tests on their drafts, will they also make it cost-effective enough for the external reader to be willing to perform them as well? If so, then the external reader will be assuming some of the conventional duties of the author, much like the external reader described in the previous section who suggests actual revisions.

Requesting or requiring commenters to perform "authoring tests"

Commenters in this situation are requested to make a set of responses about the working draft (or required by a set of defined responses) and these responses function for the authoring group as a test of how well the draft is functioning. For example, the commenter may be asked to identify the "point" of the draft, gist it, or index the key words. This author/commenter relationship has potential efficiencies not shared by the others since commenters are given a specific task to perform on the working draft.

Indeed, experienced writers using the Comments program often asked their commenters to perform a specific type of reading.

Support for the cognitive activities involved in collaborative writing

The cognitive activities involved in collaborative writing are too numerous to detail here. We focus, therefore, on one: accessing comments. Most text annotation systems are based on a hypermedia model and the primary method for accessing information in hypermedia systems is following link icons from node to node. Typically the user brings a node (e.g., a text node) onto the screen, reads its contents and notes any links, then chooses to traverse some of the links. Such localized link following is adequate for browsing tasks but has been problematic for others [Hala87]. For example, we have found that co-authors and commenters want to visually scan a set of comments quickly and resent the time required by the "search and click" interface to call up each comment, inspect it and put it away. Some researchers have worked to tailor the navigational linking system of hypermedia systems to meet user's writing needs [Catl89; Neuw87], but the access problem remains to be addressed. Our approach calls for a tailoring the program to match user's cognitive activities [Norma86].

Issue 3: Support for the practical aspects of collaboration

Most annotation systems for writers assume that all collaborators can basically look at the same text. This assumption is impractical when members of a working group are working at remote sites. Since many academic co-authoring groups are members of the same "invisible college" [Cran72] who work at a distance, we need to worry about the practical obstacles of remote collaboration. The lack of document standards makes feedback over a network impractical even within highly-motivated co-authoring groups. There are two important problems involved in making collaborative authoring practical: compatibility and permeability.

Compatibility

In a study based on interviews with one partner from each of fifty pairs of collaborative researchers in social psychology, management science, and computer science, Kraut, Galegher, & Egido [Krau87] note that "[M]ost collaborators had difficulties with the incompatibilities among programs and computing environments.... this incompatibility was one reason why a single partner in the collaboration typically controlled the manuscript and incorporated the other's handwritten annotations and changes into an electronic version of the text."

Although many existing systems for computer support for collaborative work have addressed the issue of multi-user access to manuscripts [Deli86a, b; Edwa86; Fish88; Grei87; Lela88], none has adequately addressed the issue of access by those with incompatible systems. Potential collaborators' resources (time; money; support environment, for example, computer consultants, printers), however, often preclude them from adopting such systems [Cara88; Ehrl87]. Instead, real collaborators work with some more cost-effective combination of electronic manuscripts and hardcopy, and the collaborative system remains a laboratory rather than a field system. But as Grudin [Grud88] observes [cf. Malo85], "it is difficult or impossible to create a group in the lab that will reflect the social, motivational, economic, and political factors that are central to group performance." As a result, we fail to gain the accumulated experience in using collaborative systems in real-world tasks that is crucial to the evaluation and development of support for collaborative work [Hala87]. A "systems rationalist" perspective [Klin80] may discount the important of the failure of systems to address these issues, but researchers do so at the risk of developing systems that will never be used.

Permeability

Current computer tools for collaborative work are relatively *impermeable*--that is, the boundary between electronic text and hardcopy is difficult to cross. As a result, there is a mismatch between the requirements for successful use of the system and the work habits and requirements of users. It is not unusual, for example, for researchers to annotate hardcopy manuscripts on plane trips. It is unlikely, however, that a busy researcher would be willing to do the extra work required to enter those annotations in an electronic form upon his or her return. Indeed, an examination of applications in several areas (automatic meeting scheduling, project management, group decision support) indicates that when a system requires participants to engage in additional work, it is unlikely to be used [Grud88]. Since it is unlikely that participants' needs for hardcopy input and output will diminish significantly in the near future, failure to address this issue will result in the same negative consequences as failure to address the compatibility issue. Thus, an issue for the next generation of computer tools to support collaborative work will be to make the systems more permeable--that is, to reduce to a minimum the effort required to move from the electronic medium to hard copy and from hard copy to electronic.

Serious bottlenecks in scholarly communication arise, we believe, because pre-publication protocols for scholarly activity have yet to be seriously worked out or standardized. Authors often lack the means to share formatted text over a network, and readers often lack the resources to make hard-copy. Thus, a crucial requirement of the PREP editor is that it be prepared to accept input and produce output in a variety of standard forms including voice, paper, print file, and editable file. At base, we believe that paper is a durable commodity for scholarly exchange and that systems which cannot handle paper (either on the input or output stream) will remain isolated from real users. The best case, of course, in one in which both a user and a correspondent are on an electronic network and use the same document editor. Then it is relatively easy to communicate and the main contribution of the PREP editor will be to help organize the discussion. However, even in the hardest case -- in which the correspondent forwards a handwritten draft and can accept only paper in return -- we still want the PREP editor to be useful, to let the receiving correspondent at least scan the paper in, annotate it, print it, and send it back.

New potentials for interaction

Addressing the practical issues of compatibility and permeability enables us to consider new potentials for co-authoring and commenting interaction, enabling authoring groups to explore increasing the number of readers over national networks.

THE PREP EDITOR

The PREP editor[4] approaches these issues of collaboration and co-authoring by emphasizing communication, planning, and organized annotation. Central to the PREP editor is a focus on providing a usable, visual representation of the information that will allow new kinds of communication in addition to supporting existing styles.

---

[4] This section describes the PREP editor as it is currently implemented. Since we are using a prototyping, formative design methodology to develop the PREP editor [Goul88], the system will continue to evolve in response to behavioral observations. The system is implemented on MAC IIs running MacApp. We have also currently restricted ourselves to monochrome displays, since we expect students to use the system on low-end machines.

Basic constructs

At the structural level, the PREP editor shares basic features with many of the hypermedia systems reported in the literature, for example, Intermedia [Meyr86], Neptune [Deli86] and NoteCards [Hala87]. The system defines chunks, which roughly correspond to ideas. Chunks can contain text, grids, trees or arbitrary images. The system specifically targets several chunk types (synthesis grids, synthesis trees) that we have argued are useful external representations for writers building arguments [Neuw89]. Although the workspace includes drawing tools to help users make visual connections as they are formulating their early ideas and arguments, we intend to explore to what extent these targeted chunk types alleviate the need for full-blown drawing tools. The system also defines links between chunks, so that networks of concepts can be built. During the process of planning, when concepts are being formulated and relationships among them defined, authors can choose to work with the chunks as free floating objects in a workspace, which roughly corresponds to the "network mode" in Smith *et al.*'s Writing Environment [Smit87].

Chunks are stored in a database that is shared among the collaborators. But merely having shared access to a network of ideas does not make for a collaboration. Perusing a collaborator's entire scratch space typically makes as much sense as having to sift through the clutter of books, file folders and papers on his or her desktop. To facilitate mutual intelligibility, the PREP editor provides conventions for communicating about parts of the workspace. Specifically, the system allows authors to define "drafts." A draft defines an area in the workspace that an author intends others to access and consists of a sparsely filled grid of chunks. Typically, each column in the grid is used to store different workspace content. The columns can be related to each other (or all to a main column). For example, one column might be the content of a paper and another column the plan that is guiding the construction of the content. Commenters might add columns to hold their own comments as they read. Figure 1 depicts a draft with three columns: a paper plan, the content of a paper, and a co-author's comments. The difference between a column in a draft and some chunks in the workspace is the ordering of workspace content that the columns require. At first, a PREP editor draft might not look much like the traditional linear, text-based draft, but as work progresses, it typically begins to look more and more like a traditional draft--the content typically becomes more important than the plans and comments.[5]

Using the constructs provided by a workspace and a draft built upon a sharable database, co-authors and commenters can create plans and communicate about plans and comments, in addition to simply annotating the content of the paper.

---

[5] Although it is possible to produce the final copy of a paper in the PREP editor, we have not concentrated on general page layout. We include a spelling checker and simple type manipulations (bold, italic, etc.) in the list of supported features, however.

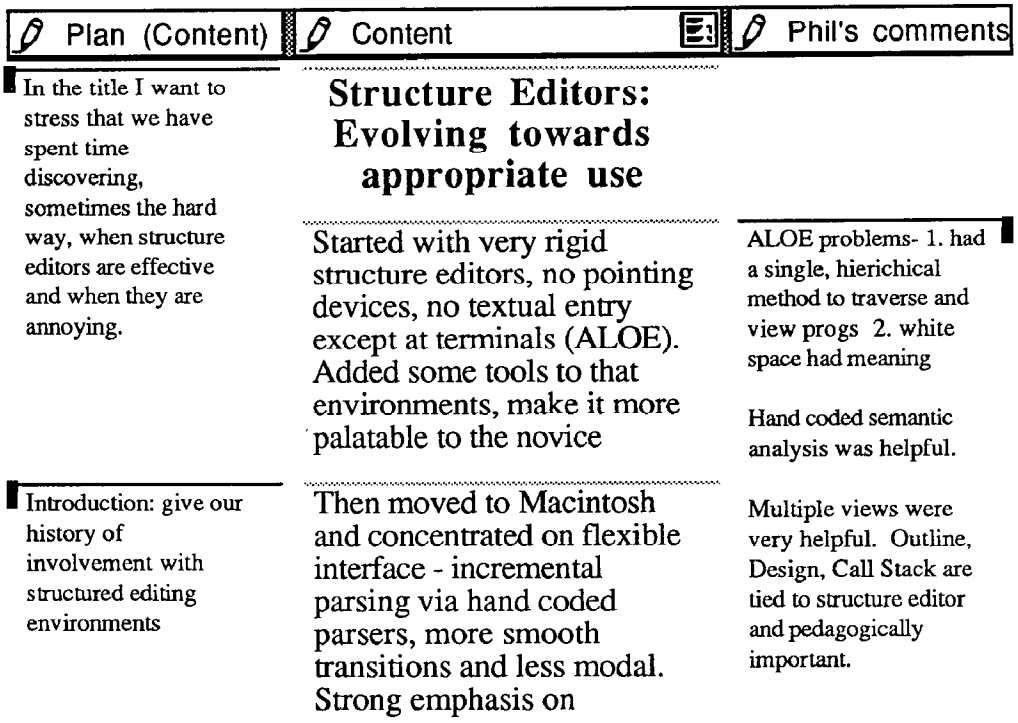

Figure 1. PREP editor with three columns in a "draft."

The interface

Much of our current work has focused on the interface, specifically on the visual representation of the draft and an optimized action grammar. For the visual representation, we are pursuing a path that could be called "dynamic glossing," since we support annotation in a style similar to old, glossed scholarly texts. While in some sense this means that we are mimicking the static annotation process, we are also taking advantage of the dynamic nature of the computer to use visual cues such as font size and spatial relationship to show the interconnections among chunks in the system. To create a visual system that will lend itself to providing and accessing comments easily, the visual grammar must be capable of supporting writers' needs. We have found, for example, that visual alignment of comments is a useful feature for allowing collaborators to see comments "at a glance" (see Figure 1), but in a flexible system, the general case requires a constraint-based layout algorithm that can handle arbitrary shapes and complex interconnections among dynamically selected items [Smol87]. This is an area we are pursuing vigorously. We have also worked on the action grammar, optimizing actions that are used frequently. For example, to create a comment, a writer need only click and drag the mouse.

Versioning

One of the most common events in a co-authoring relationship is the "edit-review-incorporate" cycle where an author gives the draft to another and the second reviews it, leaving the first author to incorporate the new material. Some systems aid this process by supporting "change bars" or other history mechanisms to indicate the points where the text has changed [Iris89]. The PREP editor will go further than this, allowing revisions to exist as distinct versions of the draft. In addition, by virtue of the inherent planning space, the reasoning behind the revisions will be communicated. This revision by versioning will allow expert writers to use operate in the space of the draft without the worry of losing old material.

Relation of the PREP editor to existing systems

The PREP editor does *not* provide a better general linking (i.e. hypertext) system. Instead, it supports linked chunks via somewhat new structural and visual constructs. These constructs would be possible to build on top of some other hypertext system. We are concentrating on improving the usability of hypertext--not its functionality.

CONCLUSION

Designing a computer tool to support co-authoring and commenting requires more than providing users with a hypermedia tool with a sharable database. Our approach has been to draw on the social and cognitive research literature in writing and upon our experience with prototype tools to identify social, cognitive and practical issues that we are attempting to address with a formative-evaluation-based prototype.

ACKNOWLEDGMENTS

The work reported here has been supported by NSF under grant number IRI-8902891. We thank Dale Miller for work on programming the PREP editor prototype and Todd Cavalier for work on graphic design for the PREP editor interface.

REFERENCES

[Agre89] Agre, P. E., & Chapman, D. *What are plans for?* MIT AI MEMO 1050a, MIT, Oct., 1989.

[APA83] American Psychological Association. *Publication manual of the American Psychological Association.* APA, Washington, DC, 1983.

[Brid87] Bridwell-Bowles, L. S., Johnson, P., & Brehe, S. Computers and composing: Case studies of experienced writers. In A. Matsuhashi (Ed.), *Writing in real time: Modeling production processes.*, pp. 81-107. Norwood, NJ: Ablex, 1987.

[Cara88] Carasik, R. P. & Grantham, C. E. A case study of computer-supported cooperative work in a dispersed organization. In M. Mantei and P. Orbeton (Eds.), *Proceedings CHI '88 Human Factors in Computing Systems*, pp. 61-66. ACM SIGCHI, Washington, D.C. May 15-19,1988.

[Catl89] Catlin, T., Bush, P., & Yankelovich, N. InterNote: Extending a hypermedia framework to support annotative collaboration. In *Hypertext'89 Proceedings*, pp. 365-378. ACM, Pittsburgh, PA, Nov. 5-8,1989.

[Cook87] Cook, P., Ellis, C., Graf, M., Rein, G., & Smith, T. Project Nick: Meetings augmentation and analysis. *ACM Transactions on Office Information Systems 5* (2):132-146, April, 1987.

[Cran72] Crane, D. *The invisible college.* University of Chicago Press, Chicago, IL, 1972.

[Deli86a] Delisle, N. M., & Schwartz, M. D. Neptune: A hypertext system for CAD applications. In *Proceedings of the ACM SIGMOD '86 International Conference on Management of Data*, pp. 132-143. ACM SIGMOD, Washington, D.C., May 28-30, 1986.

[Deli87b] Delisle, N. M., & Schwartz, M. D. Contexts--a partitioning concept for hypertext. In *Proceedings of the Conference on Computer-Supported Cooperative Work*, pp. 147-152. ACM SIGCHI SIGOIS, Austin, TX, December 3-5, 1986.

[Edwa86] Edwards, M. R., Levine, J. A., & Kurland, D. M. *ForComment.* Broderbund, 1986.

[Ehrl87] Ehrlich, S. F. Strategies for encouraging successful adoption of office communication systems. *ACM Transactions on Office Information Systems* 5:340-357, 1987.

[Fish88] Fish, R. S., Kraut, R. E., Leland, M. D. P., & Cohen, M. Quilt: A collaborative tool for cooperative writing. In *Proceedings of COIS '88 Conference on Office Information Systems*, pp. 30-37. ACM SIGOIS, 1988.

[Flow81] Flower, L. and Hayes, J. R. The pregnant pause: An inquiry into the nature of planning. *Research in the Teaching of English 15*:229-243, October, 1981.

[Flow89] Flower, L., Schriver, K. A., Carey, L., Haas, C., & Hayes, J. R. *Planning in writing: The cognition of a constructive process.* Technical Report 34, Center for the Study of Writing, Carnegie Mellon University, July, 1989.

[Gere87] Gere, A. R. *Writing groups: History, theory and implications.* Southern Illinois University Press, Carbondale, IL, 1987.

[Goul88] Gould, J. D. How to design usable systems. In M. Helander (Ed.), *Handbook of human-computer interaction.* Elsevier Science Publishers B. V., North-Holland, Amsterdam, 1988.

[Grei87] Greif, I., & Sarin, S. Data sharing in group work. *ACM Transactions on Office Information Systems* 5(2):187-211, April, 1987.

[Grud88] Grudin, J. Why computer-supported cooperative work applications fail: Problems in the design and evaluation of organizational interfaces. In *Proceedings CSCW '88 Conference on Computer-Supported Cooperative Work*, pp. 85-93. ACM SIGCHI & SIGOIS, Portland, OR, September 26-29, 1988.

[Haas89] Haas, C. How the writing medium shapes the writing process: Effects of word processing on planning. *Research in the Teaching of English, 23* (2):181-207, May, 1989.

[Hala87] Halasz, F. G. Reflections on NoteCards: Seven issues for the next generation of hypermedia systems. In *Hypertext'87 Proceedings*, pp. 345-365. ACM, Chapel Hill, NC, November 13-15, 1987.

[Hala87a] Halasz, F. G., Moran, T. P., & Trigg, R. H. NoteCards in a Nutshell. In *Proceedings of the 1987 ACM Conference on Human Factors in Computer Systems* (CHI+GI '87), pp. 45-52. Toronto, Ontario, Apr 5-9, 1987.

[Hartms] Hartman, K., Neuwirth, C. M., Kiesler, S., Sproull, L., Cochran, C., Palmquist, M., & Zubrow, D. Patterns of social interaction and learning to write: Some effects of network technologies. Manuscript under review.

[Haye80] Hayes, J. R., & Flower, L. Identifying the organization of writing processes. In L. Gregg & E. Steinberg (Eds.), *Cognitive processes in writing: An interdisciplinary approach.* Lawrence Erlbaum, Hillsdale, N.J., 1980.

[Haye87] Hayes, J. R., Flower, L., Schriver, K. A., Stratman, J., & Carey, L. Cognitive processes in revision. In S. Rosenberg (Ed.), *Advances in applied psycholinguistics, Volume II: Reading, writing, and language processing.* Cambridge University Press, Cambridge, England, 1987.

[Iris89] Irish, P. M., & Trigg, R. H. Supporting collaboration in hypermedia: Issues and experiences. In E. Barrett (Ed.), *The Society of text: Hypertext, hypermedia, and the social construction of information*, pp. 90-106. MIT, Boston, MA, 1989.

[Kauf86] Kaufer, D. S., Hayes, J. R. & Flower, L. Composing written sentences. *Research in the Teaching of English 20*(2):121-140, May, 1986.

[Kaufms] Kaufer, D. K., & Carley, C. *Interaction at a distance.* ms.

[Klin80] Kling, R. Social analyses of computing: Theoretical perspectives in empirical research. *Computing Surveys, 12*'(1): 61-110, 1980.

[Krau87] Kraut, R. E., Galegher, J., & Egido, C. Relationships and tasks in scientific research collaboration. *Human-Computer Interaction 3*:31-58, 1987.

[Lela88] Leland, M. D. P., Fish, R. S., & Kraut, R. E. Collaborative document production using Quilt. In *Proceedings of CSCW '88 Conference on Computer-supported Cooperative Work*, pp. 206-215. ACM SIGCHI & SIGOIS, Portland, OR, September 26-28, 1988.

[Luns90] Lunsford, A., & Ede, L. *Singular texts/plural authors: Perspectives on collaborative writing.* Southern Illinois University Press, Carbondale, IL, 1990.

[Malo85] Malone, T. W. Designing organizational interfaces. In *Proceedings CHI '85 Human Factors in Computing Systems*, pp. 66-71. ACM, San Francisco, April 14-18, 1985.

[Malo88] Malone, T. W. What is coordination theory? In *Coordination Theory Workshop.* National Science Foundation, Feb., 1988.

[Meyr86] Meyrowitz, N. Intermedia: The architecture and construction of an object-oriented hypermedia system and applications framework. In *Proceedings of the Conference on Object-oriented Programming Systems, Languages, and Applications* (OOPSLA '86), pp. 186-201. Portland OR, Sep. 29-Oct. 2, 1986.

[Neuw87] Neuwirth, C. M., Kaufer, D. S., Chimera, R., & Gillespie, T. The Notes program: A hypertext application for writing from source texts. In *Hypertext'87 Proceedings*, pp. 345-365. ACM, Chapel Hill, NC, November 13-15, 1987.

[Neuw88] Neuwirth, C. M., Kaufer, D. S., Keim, G., & Gillespie, T. *The Comments program: Computer support for response to writing.* CECE-TR-3, Center for Educational Computing in English, English Department, Carnegie Mellon University, January, 1988.

[Neuw89] Neuwirth, C. M., & Kaufer, D. S. The role of external representations in the writing process: Implications for the design of hypertext-based writing tools. In *Hypertext '89 Proceedings* , pp. 319-342. ACM, Pittsburgh, PA, November 5-8, 1989.

[Neuw89a] Neuwirth, C. M., Palmquist, M., & Gillespie, T. Role playing in peer review: The Devil's Advocate exercise. In D. Beil (Ed.), *Teacher's guide to using computer networks for written instruction*, pp. 157-164. Realtime Learning Systems, Washington, DC, 1989.

[Norm86] Norman, D. A. Cognitive engineering. In D. A. Norman & S. W. Draper (Eds.), *User-centered system design*, pp. 31-61. Lawrence Erlbaum Associates, Hillsdale, NJ, 1986.

[Smit87] Smith, J. B., Weiss, S. F., & Ferguson, G. J. A hypertext writing environment and its cognitive basis. In *Hypertext'87 Proceedings*, pp. 345-365. ACM, Baltimore, MD, Chapel Hill, November 13-15, 1987.

[Smol87] Smolensky, P., Bell, B., Fox, B., King, R., & Lewis, C. Constraint-based hypertext for argumentation. In *Hypertext'87 Proceedings*, pp. 215-246. ACM, Chapel Hill, November 13-15, 1987.

[Stef87] Stefik, M., Foster, G., Bobrow, D. G., Kahn, K., Lanning, S., & Suchman, L. Beyond the chalkboard: Computer support for collaboration and problem solving in meetings. *Communications of the ACM 30*(1):32-47, January, 1987.

[Tang88] Tang, J. C., & Leifer, L. J. A framework for understanding the workspace activity of design teams. In *Proceedings CSCW '88 Conference on Computer-Supported Cooperative Work*, pp. 244-249. ACM SIGCHI & SIGOIS, Portland, September 26-29, 1988.