Guided Tours and Tabletops: Tools for Communicating in a Hypertext Environment

RANDALL H. TRIGG
Xerox Palo Alto Research Center

The author of a complex hypertext document is often faced with the problem of conveying the document's meaning to future readers through a shared computer environment. Two tools implemented in the NoteCards hypertext environment, guided tours and tabletops, allow authors to employ annotation, graphic layout, and ordered presentation when communicating to readers. This paper describes these tools and gives examples of their use. Issues of remote pointing arising from an application in legal argumentation are discussed as well as early work on the use of these tools to support sharing of hypertext strategies among NoteCards users.

Categories and Subject Descriptors: H.1.2 [**Models and Principles**]: User/Machine Systems—*human factors*; H.4.2 [**Information Systems Applications**]: Types of Systems; I.7.m [**Text Processing**]: Miscellaneous

General Terms: Design, Experimentation, Human Factors

Additional Key Words and Phrases: Collaborative work, hypermedia, hypertext

1. INTRODUCTION

Hypertext is a technology whose potential in a variety of domains is increasingly apparent despite publicity that at times stretches the limits of credibility. This potential was in evidence during Hypertext '87, the first conference devoted solely to the topic of hypertext and hypermedia systems and applications [8]. While encouraging the enthusiasm and high hopes of the participants, the conference organizers also devoted a significant portion of the discussions and presentations to the outstanding problems. Perhaps primary among these was the problem of navigating through complex information networks or being "lost in hyperspace." In particular, it was noted that more resources need to be devoted to easing the burden placed on the reader of hypertext documents [5]. To read such a document one not only has to make sense of the text at each node but also must navigate through the document without becoming lost or confused. Solving this problem requires a deliberate act of communication between author and reader. In this sense, our concerns overlap with research in what is called the "rhetoric of hypertext" [9]. However, that work primarily addresses the

Author's address: Xerox Palo Alto Research Center, 3333 Coyote Hill Road, Palo Alto, CA 94304.
Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.
© 1989 ACM 0734-2047/88/1000-0398 $01.50

semantics of individual links, whereas the work described here concerns communication through multilink, multinode structures.

In addition, this research is driven by a desire to provide the means within computer systems to support the formation and sustenance of user communities. This accords with an increasing interest among computer system designers in system evolution. These designers are expanding their view of the design task from creating and installing "black-box" technologies to a more encompassing notion of system that includes the activities and environments into which the technology is introduced. This broadened view leads to an emphasis on involving users in the design process, designing for change, and supporting dispersed and evolving communities of users. Systems are designed to leave (or restore) control of the activity in the hands of the user rather than "deskilling" the user's work by locating control in the software. Such systems need to be malleable in the sense that the user can appropriate them to the task at hand in ways that arise naturally in the course of activity. The danger, however, is that as systems become more malleable, they also become more complex due to an increased range of options. This forces users to spend more time dealing with the system and less with the task at hand. In cases of well-established activities for which there is a wealth of collective experience, this danger can sometimes be avoided. For example, an electronic mail system can be designed to be a transparent tool, allowing users to think about the message being composed rather than the software. (For more on the tool perspective, see [1].)

Computational substrates such as hypertext, however, are characterized by their extreme generality. Such technologies are intended to be flexible enough to support a variety of applications, in effect acting more like a medium than a traditional tool. Users of such general-purpose systems tend to develop application- and style-specific strategies. A primary motivation for the work described in this paper was to design and implement a means for communication among a community of users such that the strategies developed by community members are available within the medium itself. In this case, the medium is NoteCards, a hypermedia system developed at Xerox PARC and in use in a variety of communities throughout North America and Europe [6]. The particular NoteCards tools described here are the *guided tour* and its building block, the *tabletop*. In what follows, I first provide background on the general notion of guided tour followed by an overview of NoteCards and the two tools. Next, I describe the experience of an early user, focusing on deictic reference, that is, on-screen cross-window pointing. Following a discussion of the on-line strategy manual as a guided tours application, I conclude by noting two specific directions for future work.

2. GUIDED TOURS: LINEAGE AND KINSHIP

The ancestry of guided tours dates back to Vannevar Bush's classic "As We May Think" article in which he describes the notion of a "trail." For Bush, a trail is a sequence of links through a "memex."

> . . . when numerous items have been thus joined together to form a trail, they can be reviewed in turn, rapidly or slowly . . . It is exactly as though the physical

> items had been gathered together from widely separated sources and bound together to form a new book. It is more than this, for any item can be joined into numerous trails [3].

In Bush's scenario, the trails being created covered portions of "the enormous mass of the common record." Much of the work of research was viewed as the creation of such trails: "The inheritance from the master becomes, not only his additions to the world's record, but for his disciples, the entire scaffolding by which they were erected" [3]. The idea of trails was implemented as *paths* in the TEXTNET system [14]. Following Bush, TEXTNET's paths are entities that can be named, saved, and communicated to fellow browsers of the network.

Hammond and Allinson [7] describe a similar tour-like capability based on a "travel holiday metaphor." Their context is one of developing navigational tools in a hypertext system for teaching in nonformal fields. They distinguish between two forms of navigation, learner controlled and system controlled, and use the term *guided tour* to refer to the latter. Users can get on and off these tutorial tours at will and monitor their progress using maps or overviews of the database.

A somewhat different approach to tour-like facilities is Zellweger's notion of "active path" [18]. Her paths or "scripts" connect objects in and among multimedia documents. Each stop along the path can cause the system to perform any of a range of actions including "play back a previously-recorded voice annotation, send text to a text-to-speech synthesizer, open a new window, animate a picture, or query a database" [18]. Weyer and Borning's electronic encyclopedia [17] implements a similar notion of active tour and includes support for interactive simulations.

The guided tours facility in NoteCards is meant to support communication between the author of a "notefile" and future readers and, as such, overlaps to some degree with the goals of these and other systems. However, the facility differs from existing systems in at least three important ways. (1) The guided tour is accessed via a graph-based interactive interface allowing both tour authors and readers to work from the same concise overview of the guided tour's structure. (2) The stops on a NoteCards guided tour are *sets* of cards in the hypertext network arranged on the screen according to a particular layout. This allows the spatial juxtaposition of multiple sources of information in addition to the sequential organization inherent in the tour. (3) The tour itself is a full-fledged node in the hypertext network.[1]

Before diving into the details of guided tours, a brief overview of the NoteCards system is in order.

3. OVERVIEW OF NOTECARDS

NoteCards is an extensible computer environment that supports the formulation, structuring, and management of information [6]. NoteCards provides the user with a "semantic network" of electronic notecards interconnected by typed links. The system provides tools to organize, manage, and display the structure of the

[1] Elli Mylonos (personal communication, February, 1988) has augmented Apple's Hypercard system to support such tours (i.e., tours which are themselves user-accessible hypertext nodes). In her system, a single card is used to store the sequence of cards in a linear tour. The code implementing the "Next" button on each card in the tour refers back to the tour card to compute the next card to bring up.

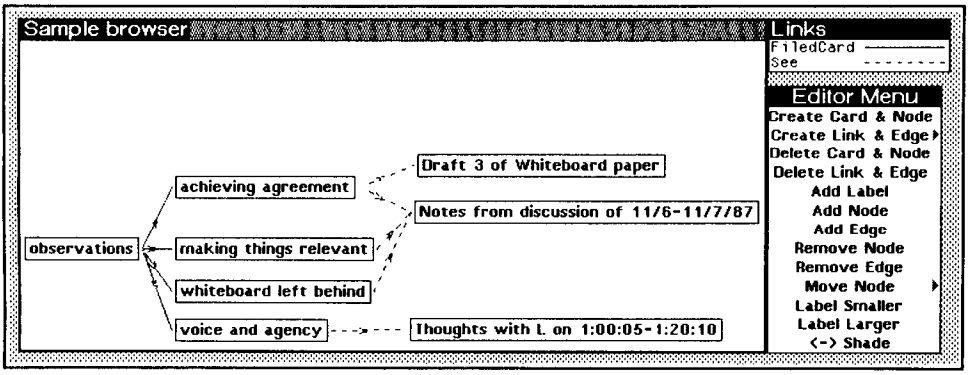


Fig. 1. Example of a NoteCards browser.

network, as well as a set of methods and protocols for creating programs to manipulate the information in the network. To view or edit a notecard, the system retrieves the card's "substance" from the database and displays it in a window appropriate to the type of card, for example, text, graphics, and animation. The links connecting cards are used both to organize and navigate through the network. Individual links are represented by link icons (usually boxed titles) in the card's substance. All of the information relevant to a network of cards and links is stored in a structured file called a *notefile*, managed transparently by the system.

Different kinds of notecards are defined in an extensible hierarchy of notecard types. These include *fileboxes* for representing hierarchical structures, *sketch* cards for rendering drawings as well as text, and *browser* cards containing node-link diagrams (i.e., maps) of arbitrary pieces of the NoteCards network. Because the guided tour card is a specialization of the NoteCards browser, it is worth taking a closer look at that particular card type. (See [15] for details on tailoring NoteCards by creating new card types.)

Browser cards contain graph-based displays of portions of the network of cards and links in a notefile. They are created by specifying a set of "root" cards, a set of link types, and a cutoff depth. The resulting graph consists of all cards reachable by starting from the root cards and following links of the given types to the given depth. The graph is layed out automatically with nodes represented as selectable link icons corresponding to the cards (buttoning a node brings up the corresponding card). Edges are represented as lines drawn in different dashing styles according to the type of the corresponding link (see Figure 1). Once the browser has been created, it can be modified by moving nodes (rubber banding any attached edges) and by adding textual annotations. In addition, it can be used to change the structure of the notefile itself. That is, cards and links can be created and deleted using the menu attached to the lower right corner of the browser in Figure 1. (It is this capability that allows authors to create guided tours from the same facility that readers use to follow them.)

Guided tours differ from standard browsers in that only links from a restricted set of types are allowed. Furthermore, the cards appearing in a guided tour

browser are almost always tabletop cards since these comprise the "stops" on the tour.

4. TABLETOP CARDS

Tabletop cards are a means in NoteCards for capturing the layout of a particular set of cards on the screen. A tabletop is a snapshot which records the list of cards, the shapes of their windows, their positions on the screen, the scrolled locations (vertically and possibly horizontally) of the windows' contents, and the order in which to open the windows so that the original (possibly) overlapping arrangement can be preserved.[2] When a tabletop card is created, the user is asked to select a set of cards forming the contents of the tabletop. Cards are selected from those open on the screen so that the current layout can be captured in the tabletop. Once a tabletop card has been created, various operations can be performed on it either from the title bar of the tabletop card's window, or more simply, from a link icon pointing at the tabletop card (Figure 2). Several of these operations allow modification of the tabletop card's contents, that is, which cards are to be included in the tabletop layout. Though the tabletop card has no directly editable substance, it can be opened, closed, (re)titled, and linked in the same way as cards of other types.

The operations available for tabletops are

Recompute Tabletop. Usually, the user has rearranged the cards in the tabletop, thereby changing some aspect of their layout. Choosing Recompute causes the new layout to be installed in the tabletop card.

Recompute Tabletop—New Cards. Accessed as a "pull-across" menu item from *Recompute Tabletop*, this operation lets the user specify a new set of cards. That is, the tabletop is recomputed from scratch.

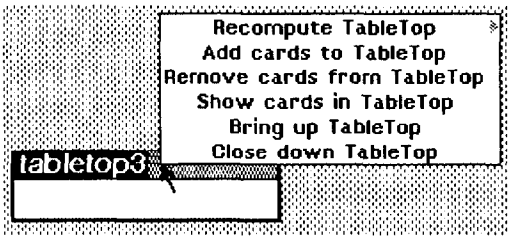
Add Cards to Tabletop. The user adds cards to the tabletop by selecting their open windows. (Alternatively, links to the cards being added can be selected.) Once the selection is made, the tabletop is recomputed so that the new cards are correctly ordered relative to the cards already in the tabletop.

Remove Cards from Tabletop. The user removes cards from the tabletop by selecting from a menu of their titles.

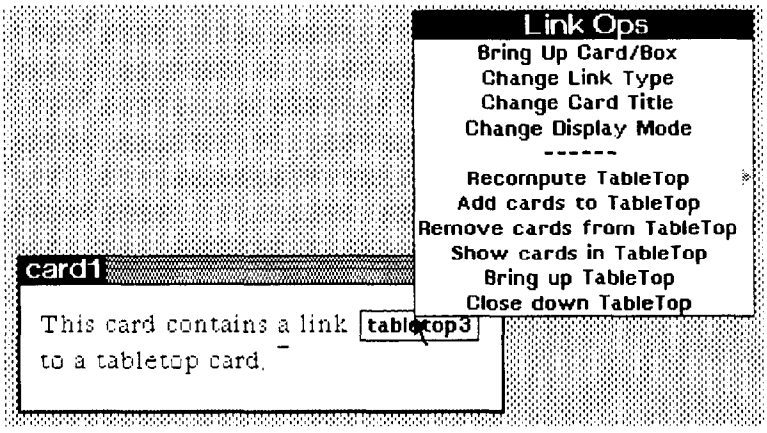
Show Cards in Tabletop. This brings up a menu containing the titles of all cards in the tabletop. The user can select from this menu to bring up a single card in the tabletop, which is at its assigned position/shape and scrolled appropriately.

Bring Up Tabletop. This opens all cards in the tabletop and positions, shapes, scrolls, and orders their windows according to the layout information stored in

[2] This notion of using one card to capture and direct the layout of a set of other cards arose in several places in the NoteCards user community. For example, in order to support on-line demonstrations of notefiles, a "script" card was invented independently by at least two user groups. Script cards are specialized text cards containing links to other cards arranged in groups, where the groups of links are bracketed using carriage returns. A "Next" button is provided somewhere on the screen which moves the cursor to the next line of the script card and brings up the linked cards appearing on that line. One group used Script cards to build on-line presentations of a notefile used to study works of ancient Greek literature [11]. A similar facility was developed in a system to support instructional design [15].



(a)



(b)

Fig. 2. (a) Accessing tabletop operations from a tabletop card. (b) Accessing tabletop operations from a link icon.

the tabletop. Any cards already open are moved into place and scrolled appropriately. (Note that bringing up a tabletop is not the same as opening a tabletop card. When a link is followed to a tabletop card, for example, the card is opened in the standard way. Bringing up a tabletop causes the cards *in* the tabletop to be opened. The tabletop card itself is only opened if it is an element of its own tabletop.)

Close Down Tabletop. This closes any open cards in the tabletop. Note that multiple tabletops can both share cards and be open simultaneously. Closing down one of them closes all of its open cards, including those shared by other open tabletops.

In addition to their use as building blocks for guided tours (described in detail in the next section), tabletops have had three main uses: as storage of card layout across notefile closings/openings, as aids for on-line demonstrations of notefiles, and to help in creating hard copy screen snapshots for off-line presentations. Figure 3 shows one tabletop from a sequence used to demonstrate collaborative aspects of a shared project notebook notefile. The filebox "Tabletops" on the lower right contains links to each of the tabletops in the sequence. (The displayed tabletop was brought up directly from the link icon labeled "history TT" (as in

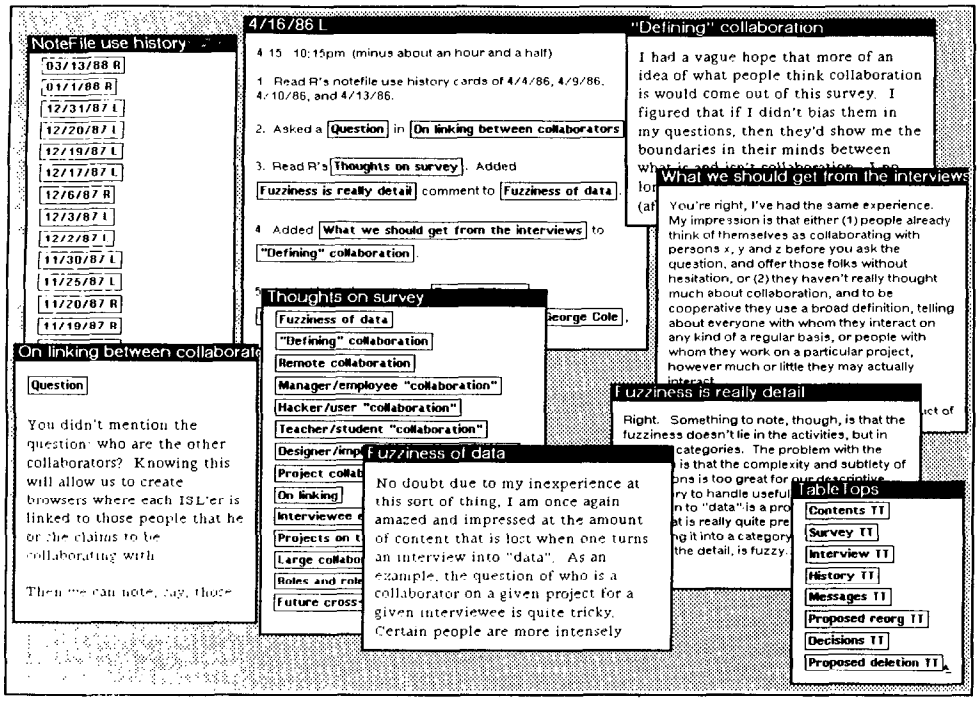


Fig. 3. Filebox of tabletops with "History" tabletop displayed.

Figure 2b), avoiding the need to first open the tabletop card.) These same tabletops were used to create transparencies for use in talks about collaborative NoteCards.

5. GUIDED TOURS

The guided tour facility consists of a graphic interface to a network of tabletop cards. Specifically, a guided tour is a graph whose nodes are tabletop cards and whose edges are *GuidedTour* links connecting the cards.[3] Because the tabletop cards may have other links connecting them to other parts of the network, the guided tour is a subgraph of the subnetwork containing those cards. As described earlier, the guided tour card is a specialization of the NoteCards browser card.

5.1 Creating a Guided Tour

Authors of new guided tours usually begin by creating an empty guided tour card. Using an attached menu, they create new tabletop cards and position the icons in the guided tour card's window. With a few mouse clicks they can link their tabletops together to form the paths (actually an arbitrary graph structure)

[3] The links joining tabletop cards in a guided tour are examples of *global* or node-to-node links as opposed to the more common *local* or point-to-node links. Global links are anchored to the card as a whole rather than to a point within the card's substance. Though not marked by icons in the card's substance, the global links of any card can be inspected and traversed via the *ShowLinks* option available from the card's title bar.

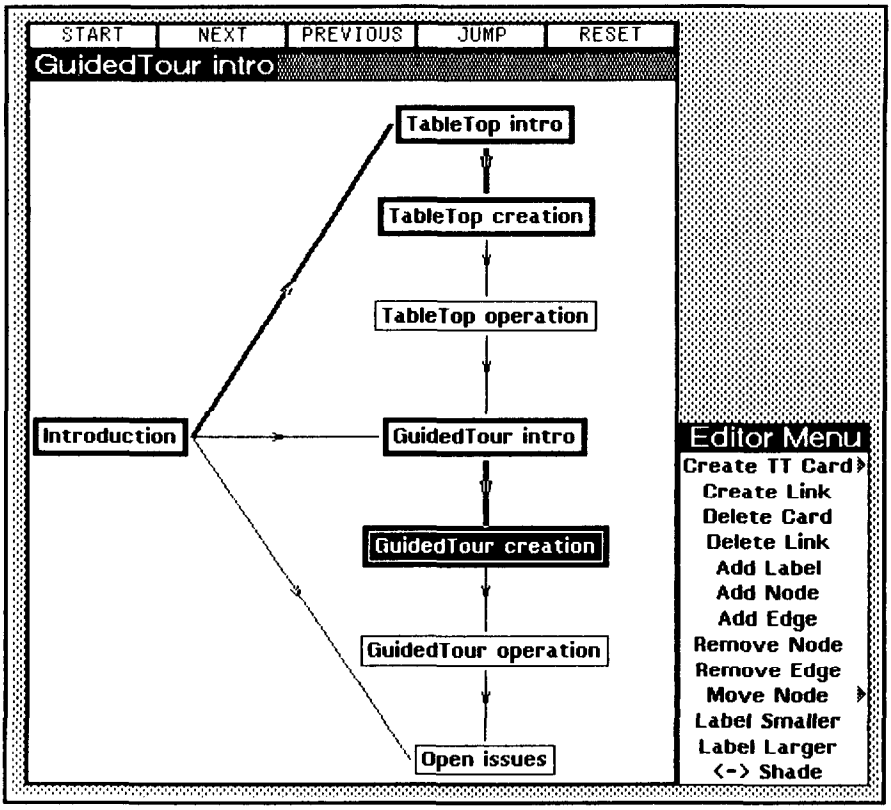


Fig. 4. A guided tour card.

making up a guided tour. Authors of guided tours can also incorporate existing tabletops into their tour either by pulling them into the graph one at a time or by using the automatic browser capability (described in Section 3) to bring in and layout a piece of some existing tour. The operations used to create and modify guided tours can be seen in the menu attached to the lower right corner of the card in Figure 4. Because these operations are essentially the same as those for editing browser cards, the training necessary for an experienced NoteCards user to build guided tours is minimal.

5.2 Operating a Guided Tour

To operate or "run" a guided tour, the reader makes use of the five buttons arrayed along the top of the guided tour card (Figure 4). The operations invoked by these buttons are described below. In each operation, the current tabletop is automatically closed down before bringing up the next. (This is made more efficient by not closing those cards shared by the current and next tabletops.)

START—Clicking on START causes the first tabletop in the tour to be brought up. For simple linear or singly rooted tree structures there is an obvious "first" tabletop. Some tours, however, might have multiple roots, that is, multiple nodes

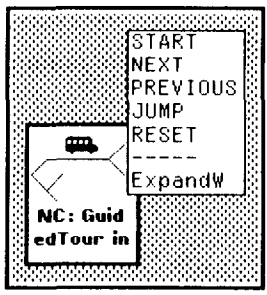


Fig. 5. Operating a guided tour from the card window's shrunken icon.

without incoming edges. In such cases, the user is asked to choose from a menu of possible starting tabletops.

NEXT—This is the standard way to move along a path. Clicking NEXT simply closes down the current tabletop and brings up the next one on the path. If the current node is a branch point having multiple outgoing edges, then the user is asked to choose from a menu of possible next tabletops. In any case, the new current node is highlighted as is the edge that was followed.

PREVIOUS—Clicking here provides the reader with a menu of tabletops already visited. These appear in the order in which they were visited. Clicking on an item in the menu effectively JUMP's to that point in the tour.

JUMP—This operation is used to leave the current path by jumping to an arbitrary tabletop selected by the user from the guided tour graph.

RESET—This operation closes the current tabletop and resets the "state" of the guided tour. In particular, any highlighting of nodes and links in the guided tour graph is turned off and the PREVIOUS list is cleared.

At any point in time, the graph structure displayed in the guided tour card provides various indicators of the reader's place in the tour. The node in the graph whose tabletop is currently open is highlighted in "reverse-video." The nodes for tabletops previously visited are drawn with heavier borders. Finally, the edges corresponding to links followed using the NEXT command are displayed in bold. The guided tour card appearing in Figure 4 shows a point in time when the reader has visited five cards. Starting at "Introduction," the top branch was taken, two tabletops were visited, and then a JUMP was made to "GuidedTour intro" whereupon NEXT was again used to move to the presently displayed tabletop "GuidedTour creation."

This standard mode of operating a guided tour (using the buttons along the top of the card) is extended in several ways. First, because screen space is at a premium and because the window containing the guided tour graph is often large, the user can run the tour from the guided tour window's shrunken icon as shown in Figure 5. (If the JUMP command is selected from the shrunken icon, the guided tour window is temporarily opened to allow the user to select a node to jump to. Following this "expansion," the window is automatically reshrunk to its icon form.) Second, it may be desirable to "circumvent" the tour by peeking at the cards in the next tabletop or by bringing up a tabletop without closing down the current one. This can be done by using the mouse to select a node in the guided tour graph and then invoking one of the standard tabletop operations.

Finally, it is important to emphasize that the user can stray from the tour at any time by following links emanating from cards in the tabletop. Such explorations deeper into the notefile do not affect the current state of the tour.

6. DEICTIC REFERENCE IN GUIDED TOURS

The purpose of guided tours is to help make the contents of a notefile intelligible to someone not already familiar with it. Often, in addition to the cards themselves, one needs to provide a description of and rationale for their contents, explication of their context of use, and even history of their construction. Only the simplest notefiles are self-explanatory in these ways. Naturally, these descriptions must make reference to the cards they describe. In an on-line demonstration of a notefile, pointing can be done by the demonstrator using a mouse. In an on-line guided tour, the author must communicate such an explanation *in absentia*. We use the term *remote deictic reference* to refer to the on-screen cross-window pointing involved in such communication.

In an early application of guided tours, Cathy Marshall, a researcher studying legal argumentation, confronted the problems of remote deictic reference head-on. Her notefile contains an analysis of the arguments in a Supreme Court case *People v. Carney* concerning the legal status of mobile homes with regard to issues of individual privacy and warrantless searches. She is concerned with the theoretical issues underlying logical structures of arguments as well as physical representation of those structures in a hypertext environment [10]. In an effort to communicate the contents of her notefile (and at the same time the essence of her work) to future readers, she created the guided tour shown in Figure 6. Her tour contains two separate paths meant for possibly different audiences. The path on the right is directed at those interested in an exploration of the case itself and the relevant legal issues. The longer path on the left (labeled "Overview of the contents of this notefile") is used to inform readers of the representation issues relevant to her analysis. The side paths branching off this "spine" provide background, rationale, and further detail.

Figures 7, 8, and 9 show several styles of pointing used by Marshall in her guided tour. In Figure 7, the large card at the bottom shows an argument about motor homes structured according to the Toulmin representation appearing in the middle card.[4] The card at the top provides a general description of the structure, and the two descriptive cards containing arrows pick out salient features of the structures. Note the use of overlapping cards (along with the arrows) to draw attention to the referents of the descriptions. The descriptive card titled "Looking at argument constituents" invites the reader to follow a link appearing in a different card.

In Figure 8, careful spatial layout is used so that the descriptive cards along the left align with the columns they describe in the argument network card. Note the card at the bottom of the tabletop pointing the interested reader toward similar argument browsers elsewhere in the notefile. Finally, in Figure 9, Marshall attempts to influence the order in which we as readers peruse the cards in the

[4] See [10] for an explanation of the NoteCards representation of Toulmin's "argument layout" [12] appearing in Figure 6.

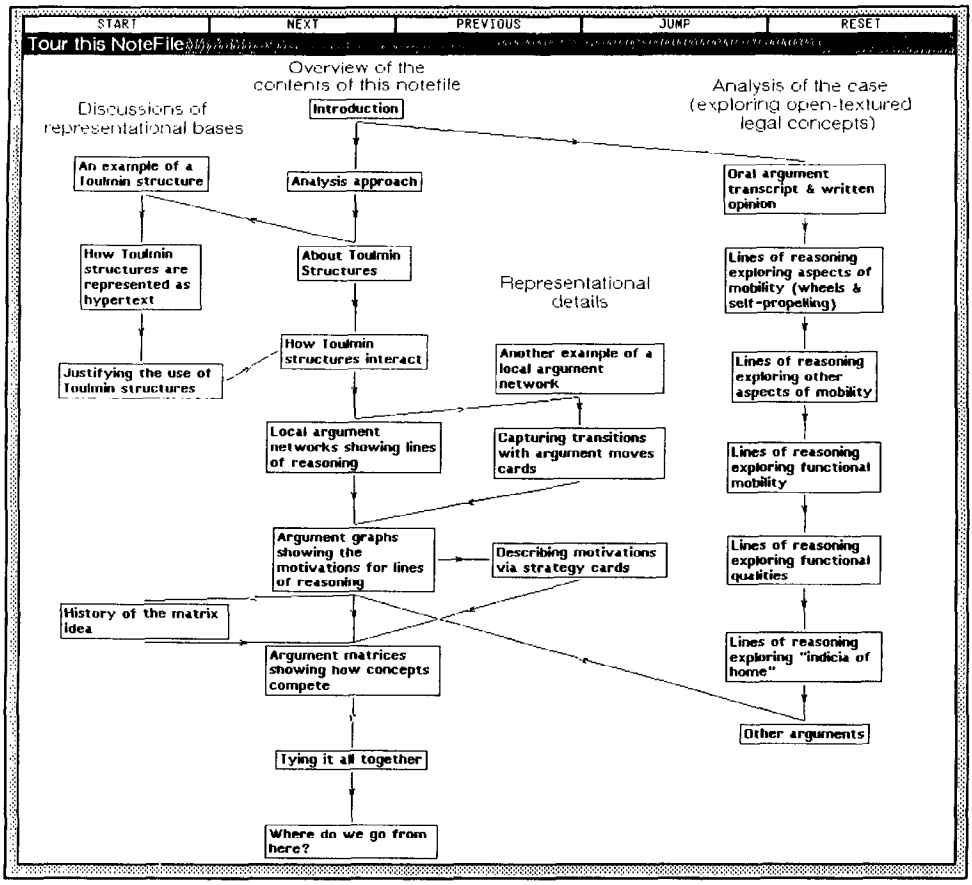


Fig. 6. Guided tour with multiple paths.

tabletop. The descriptive card directly below the "title" card at the upper right starts our gaze moving in a counterclockwise direction through the argument structure on the left, the description at the lower left, and the structure card on the right.

One might suppose that the problem of pointing between cards is solvable using standard NoteCards links. After all, this is the way pointing traditionally happens in hypertext systems. Today, however, links in NoteCards are inadequate to the task in two ways. First, though the source anchor of a link can be a point within a card, the destination must be an entire card. Point-to-point links would be useful for connecting descriptive cards to points or objects inside described cards. The second deeper problem is that these links need to be context dependent. Often, they only make sense in the context of the tabletop in which they appear and even in the context of a particular path within the enclosing guided tour. Though there are hypertext systems that support point-to-point links, I am not aware of any that have implemented context-dependent links.

In principle, any number of contextual features ranging from user-settable switches to the time of day could be used to constrain the set of links visible in

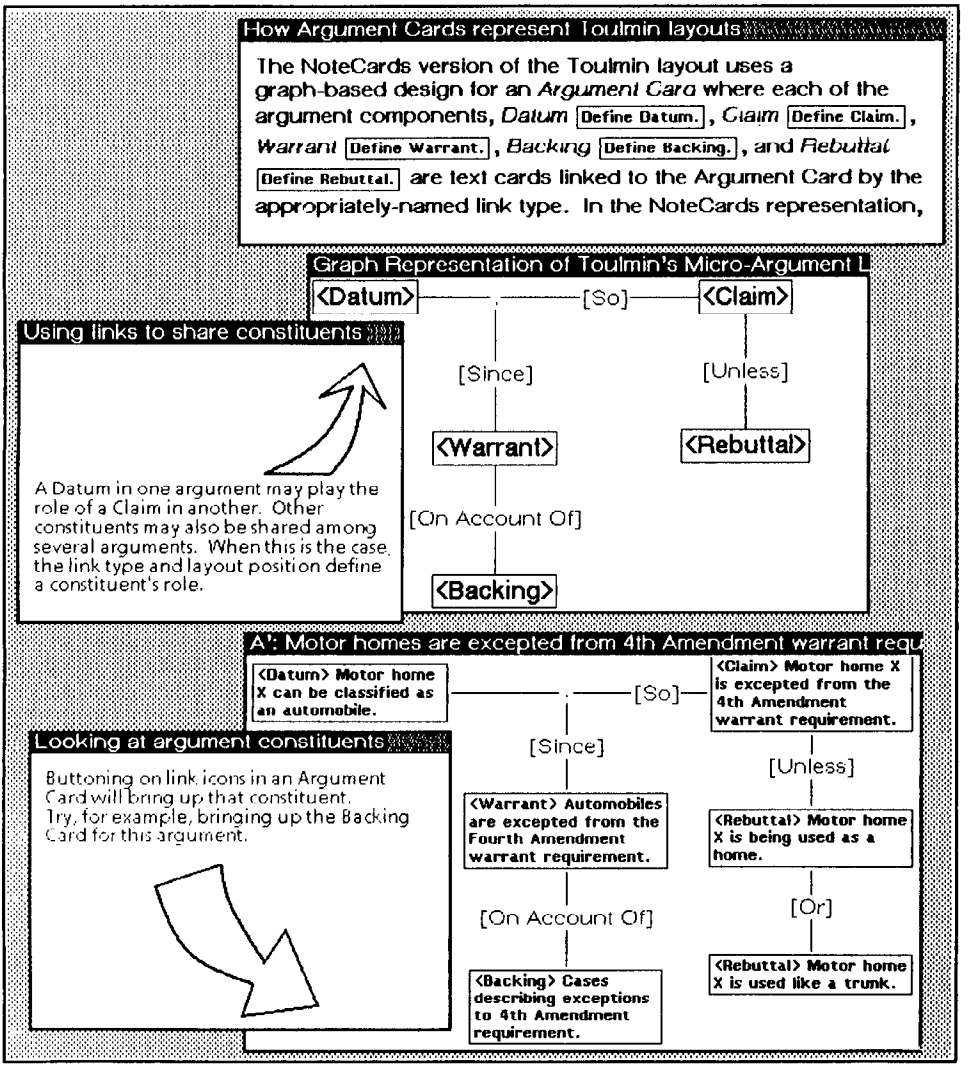


Fig. 7. Pointing between cards in a tabletop.

a given situation. With regard to guided tours and tabletops, however, it is useful to identify three sorts of card-oriented "temporal" contexts. Past context includes cards that the reader has already seen, perhaps earlier in a guided tour, perhaps by "off-path" wanderings. Present context includes the cards currently displayed on the screen and their physical layout. Future context might include where the reader is headed given that the reader is currently embarked on a given guided tour. Each of these could influence the set of links (or graphically represented "pointers") visible in a displayed card.

The issues uncovered in Marshall's guided tour, however, go beyond point-to-point connections between cards and the conditional visibility of links. In fact, remote deictic reference is but one of several problems to be addressed in the

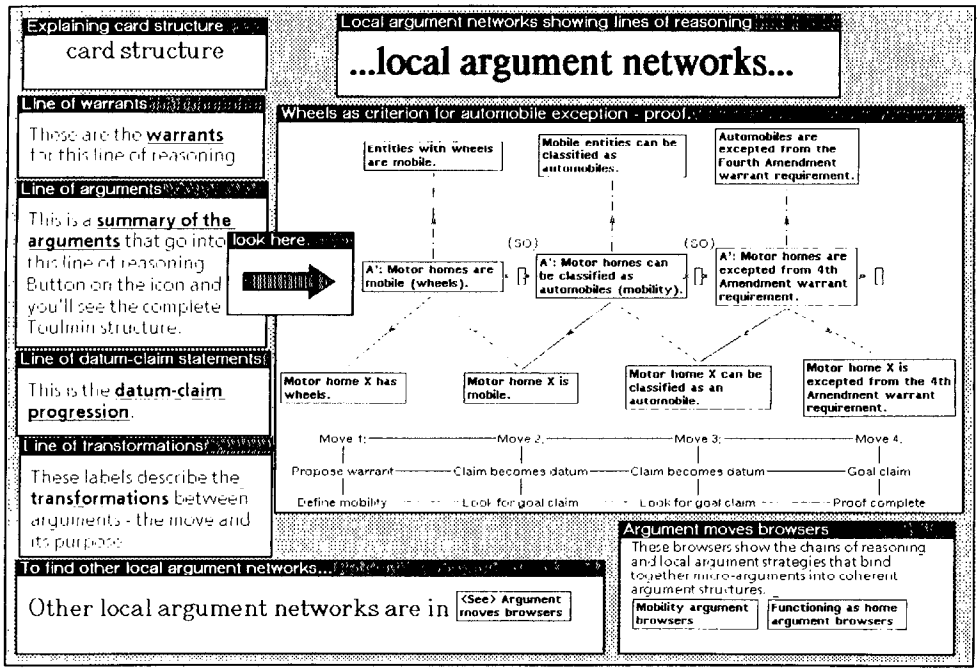


Fig. 8. Spatially aligned cards in a tabletop.

study of self-explaining hypertexts. Even within a single tabletop, the author may need to focus attention, group related cards, refer to sets of cards and regions within cards, and organize cards spatially to encourage specific patterns of perusal. I expect practical and aesthetic conventions (and eventually tools that support them) to arise partly from user experience over time and partly by drawing on work in such fields as graphic design. Starting from the perspective of a rhetoric of hypertext, Carlson [4] arrives at the same requirement for "composition guidelines":

> Traditional rhetoric considers the sentence and the paragraph as the basic units of composition. Answers to the issues of modularization and node size may help to isolate the basic units in the rhetoric of hypertext; however, further analysis and description of prose in hypertext systems must take into account the medium of display. In short, the "grammar" of hypertext intersects issues of screen design and user interface [4, p. 96].

7. THE NOTECARDS STRATEGY MANUAL

As described in the introduction, a primary motivation for this work is to support communication among members of the NoteCards user community. To test the tools described here, we have chosen to focus on the following question: How can dispersed users working in a tailorable system like NoteCards exchange methods, styles, and strategies for using the technology? This focus addresses a central need within the user community and at the same time provides the opportunity to study problems and successes in supporting mutual intelligibility using guided

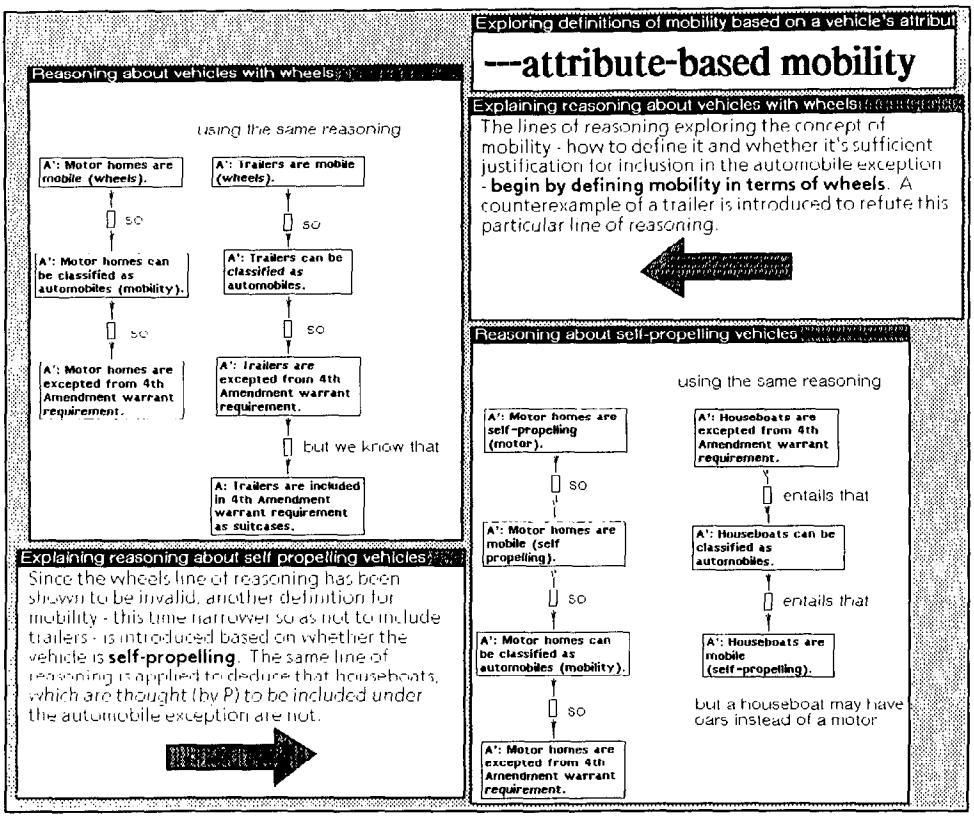


Fig. 9. Directing the reader's gaze in a tabletop.

tours. (For further discussion of the use of NoteCards to support mutual intelligibility, see [16].) For example, one user might find that the browser overview in NoteCards can be used as a brainstorming tool in the early stages of writing. Rather than (or in addition to) composing a verbal description of the tool, that user could copy a representative portion of her notefile into a "strategy manual" together with a guided tour that documents and escorts the reader through it. Another user might not employ the same technique in the same way but could be inspired to devise and apply a variant. That user would then augment the original guided tour with descriptions and instances of the variation.

As of this writing, the first version of a strategy manual notefile exists, organized in two ways: by part of the system and by user application. For example, a strategy for using browser overviews as tools for writing would appear in sections on using browsers as well as on styles of writing in NoteCards. Trigg and Irish [13] report on a study of the writing styles and strategies of NoteCards users. We are currently seeding the strategy manual by drawing on the notefiles of each of the users studied there. For example, Figure 10 shows at a strategy for "top-down notetaking" taken from Julian Orr's doctoral thesis notefile. After obtaining feedback from NoteCards users on this initial version of the strategy

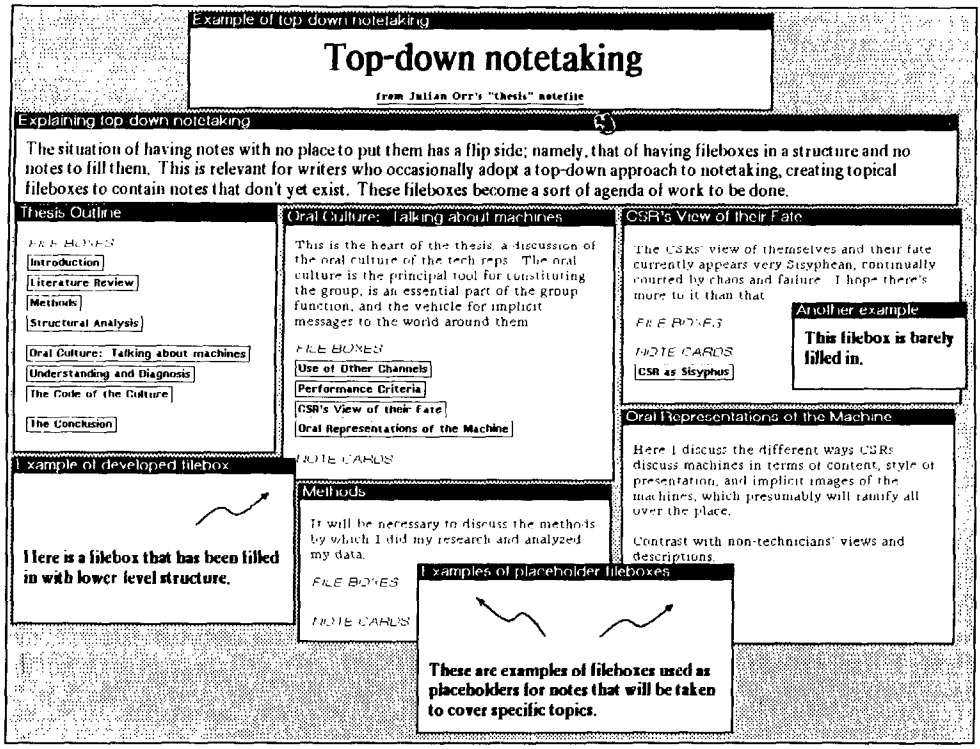


Fig. 10. A tabletop in the strategy manual notefile.

manual, we will investigate tools to support users adding new strategies and modifying existing ones.

8. FINAL REMARKS

An important topic for future work on guided tours involves the problem of matching a tour to an audience. Because readers have different interests, authors tend to create tours with branching paths and side trips. The issue then becomes how to determine which branch to take at the various choice points. The version of the system discussed here simply provides the reader with a menu of titles of possible next tabletops. However, the author may have more information to contribute to that determination, either locally or globally. At a local level, the author might know questions to ask of the user (say, regarding personal interests or time available) to help determine whether a side path is appropriate. At the global level, the author might have other questions for the reader that could determine a default walk through all the branch points. The design question is how to allow the author to plant such information in the tour for the benefit of future readers.

Another issue to be addressed in the future concerns the dynamics of tours. As described here, a guided tour captures only static layout information (tabletops) together with ordering constraints (links between tabletops). Guided tour authors

have already requested control over certain aspects of the dynamics of following guided tours. For example, the order in which a tabletop's cards are brought up is essentially unpredictable (except between cards whose windows overlap in the tabletop). Control over this bit of dynamics could help an author convey to a reader the order in which to read a tabletop's cards. In part because guided tours in NoteCards are not required to capture such dynamics, the author's interface can be completely graphical. The challenge is to preserve this visual interface while adding the ability to specify dynamic parameters. An important example of work in this area is Marc Brown's BALSA-II algorithm animation system [2]. BALSA-II provides a powerful scripting facility that captures dynamics through a "programming by example" author's interface. Though the scripts are stored as PASCAL programs, authors can effect small changes in behavior with little or no knowledge of programming. Augmenting the NoteCards guided tours tool with such a capability is an important future step.

In conclusion, many tools that support work of various kinds (text editors, spreadsheets, CAD systems) have very few capabilities for communicating *about* the work. At the same time, tools for communicating, for example, electronic mail, are difficult to integrate with these systems for doing work. Hypertext systems have the potential to represent both the work and the accompanying descriptions and metadiscussions. Tools like guided tours are needed, however, to ensure the intelligibility of hypertext documents especially on occasions when the authors are not present. My hope is that this work will inspire further research into these issues in settings where mutual intelligibility is attempted across time and space.

ACKNOWLEDGMENTS

Many thanks to Peggy Irish and Cathy Marshall for early work on the strategy manual notefile, invaluable discussions and feedback on guided tours, and comments on earlier drafts of this paper; to Lucy Suchman, Susan Irwin, and three TOOIS reviewers for helpful comments on later drafts; to Cathy Marshall and Julian Orr for permission to use figures from their notefiles; to Jeff Shrager for the first implementation of tabletops; and to NoteCards users George Cole, Dan Jordan, Tom Moran, Kenny Morrell, and Dan Russell, who participated in the design and implementation of various ancestors of today's guided tours and tabletops.

REFERENCES

1. BODKER, S., EHN, P., KAMMERSGAARD, J., KYNG, M., AND SUNDBLAD, Y. A Utopian experience: On design of powerful computer-based tools for skilled graphic workers. In *Computers and Democracy: A Scandinavian Challenge*, G. Bjerknes, P. Ehn, and M. Kyng, Eds., Gower, Brookfield, Vt., 1987, pp. 251–278.
2. BROWN, M. H. *Algorithm Animation*. MIT Press, Cambridge, Mass., 1988.
3. BUSH, V. As we may think. *The Atlantic Monthly 176*, 1 (June 1945), 101–108.
4. CARLSON, P. A. Hypertext: A way of incorporating user feedback into online documentation. In *Text, ConText, and HyperText: Writing With and For the Computer*, Edward Barrett, Ed. MIT Press, Cambridge, Mass., and London, 1988, pp. 93–110.

5. CHARNEY, D. Comprehending non-linear text: The role of discourse cues and reading strategies. In *Proceedings of the Hypertext '87 Workshop* (Chapel Hill, N.C., Nov. 13–15). ACM, New York, 1987.
6. HALASZ, F. G., MORAN, T. P., AND TRIGG, R. H. Notecards in a nutshell. In *Proceedings of the ACM CHI + GI Conference* (Toronto, Ontario, Apr. 5–10). ACM, New York, 1987, pp. 45–52.
7. HAMMOND, N., AND ALLINSON, L. Travels around a Learning Support Environment: Rambling, Orienteering or Touring? In *Proceedings of the ACM CHI '88 Conference* (Washington, D.C., May 15–19). ACM, New York, 1988, pp. 269–274.
8. HYPERTEXT 1987 PLANNING COMMITTEE. In *Proceedings of the Hypertext '87 Workshop* (Chapel Hill, N.C., Nov. 13–15). ACM, New York, 1987.
9. LANDOW, G. Relationally encoded links and the rhetoric of hypertext. In *Proceedings of the Hypertext '87 Workshop* (Chapel Hill, N.C., Nov. 13–15). ACM, New York, 1987.
10. MARSHALL, C. M. Exploring representation problems using hypertext. In *Proceedings of the Hypertext '87 Workshop* (Chapel Hill, N.C., Nov. 13–15). ACM, New York, 1987.
11. MORRELL, K., AND TRIGG, R. H. From books to workstations: Problems in developing a computer-based curriculum in the humanities. In *Proceedings of the International Conference on Data Bases in the Humanities and Social Sciences* (Montgomery, Ala., July). Paradigm Press, Osprey, Fla., 1987.
12. TOULMIN, S. *The Uses of Argument.* Cambridge University Press, Cambridge, 1958.
13. TRIGG, R. H., AND IRISH, P. M. Hypertext habitats: Experiences of writers in NoteCards. In *Proceedings of the Hypertext '87 Workshop* (Chapel Hill, N.C., Nov. 13–15). ACM, New York, 1987.
14. TRIGG, R., AND WEISER, M. TEXTNET: A network-based approach to text handling. *ACM Trans. Off. Inf. Syst. 4*, 1, 1986, 1–23.
15. TRIGG, R. H., MORAN, T. P., AND HALASZ, F. G. Adaptability and tailorability in NoteCards. In *Proceedings of Interact '87: 2nd IFIP Conference on Human–Computer Interaction* (Stuttgart, West Germany, Aug.). H.-J. Bullinger & B. Shackel, Eds. North-Holland, Amsterdam, 1987.
16. TRIGG, R. H., SUCHMAN, L., AND HALASZ, F. G. Supporting collaboration in NoteCards. In *Proceedings of CSCW '86: MCC/ACM Conference on Computer Supported Cooperative Work* (Austin, Tex., Dec. 3–5). MCC Software Technology Program, Austin, Tex., 1986, pp. 153–162.
17. WEYER, S. A., AND BORNING, A. H. A prototype electronic encyclopedia. *ACM Trans. Off. Inf. Syst. 3*, 1, 1985, 63–88.
18. ZELLWEGER, P. Active paths through multimedia documents. In *Document Manipulation and Typography*, J. C. van Vliet, Ed. Cambridge University Press, New York, 1988, pp. 19–34.

Received June 1988; revised August 1988; accepted August 1988