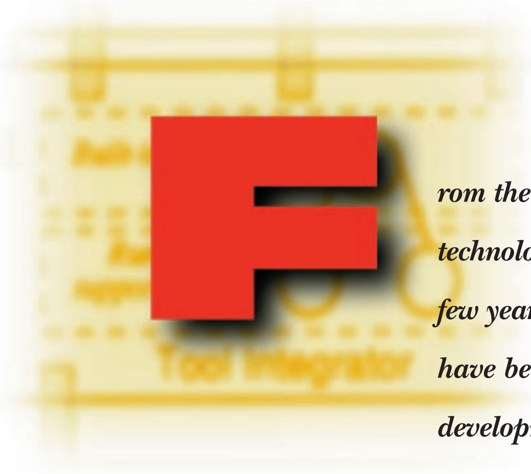
Hypermedia and Cognition:
Designing for Comprehension



From the beginning, hypermedia application design has been driven primarily by technological innovations and constrained by technical feasibility. For the last few years, however, usability methods and results from human factors research have been gaining more influence [17]. Despite this trend toward user-oriented development procedures, issues of cognition and human information processing still are widely neglected and barely influence hypermedia design.

Manfred Thüring, Jörg Hannemann,
and Jörg M. Haake

To discuss the relationship between cognition and hypermedia, it is necessary to distinguish between two kinds of applications: "One encourages those who wish to wander through large clouds of information, gathering knowledge along the way. The other is more directly tied to specific problem-solving, and is quite structured and perhaps even constrained" [20, p. 119]. Applications of the first type appear as browsable databases—or hyperbases—that can be freely explored by a reader. In contrast, applications of the second type take the shape of electronic documents—or hyperdocuments—that intentionally guide readers through an information space, controlling their exploration along the lines of a predefined structure. Each type has its particular advantages and encourages different reading strategies. While the first one is better suited to support unconstrained search and information retrieval, the second one is more adequate for tasks requiring deep understanding and learning. As Hammond points out, it "may be fun and perhaps instructive, to open every door and peer inside, but there are many situations where learning is most effective when the freedom of the learner is restricted to a relevant and helpful subset of activities."

It is this second type of task—*reading a hyperdocument for learning*—that we address in our discussion of "designing for comprehension."

Comprehension of Hyperdocuments

One major purpose—or even *the* major purpose—of reading a document is comprehension, and reading a hyperdocument is no exception. In cognitive science, comprehension is often characterized as the construction of a mental model that represents the objects and semantic relations described in a text [24]. The readability of a document can be defined as the mental effort spent on the construction

If authors want readers to construct relationships exceeding the level of local coherence, *they have to incorporate cues at net level that increase global coherence.*

process. If we want to increase the readability of a hyperdocument we must assist readers in the construction of their mental models by strengthening factors that support this process and by weakening those that impede it. Two factors in particular are crucial in this respect: *coherence* as positive influence [23] and *cognitive overhead* [3] as negative influence on comprehension.

Coherence

Empirical studies have shown that a reader's ability to understand and remember a text depends on its degree of coherence. Therefore, psycholinguistic research emphasizes the relation between coherence and information processing: A document is coherent if a reader can construct a mental model from it that corresponds to facts and relations in a possible world [12].

Two types of coherence are especially important for constructing a mental model. To understand the relation between clauses and sentences, readers infer "small scale" connections that link pieces of information together and thus establish *local coherence.* In addition, readers infer "large scale" connections that are conclusions drawn from several clauses, sentences, paragraphs, or even chapters. Such conclusions establish the *global coherence* of a text [24]. They summarize the meaning of diverse chunks of information as an abstract "macroproposition" and thus represent the common topic of the chunks. During comprehension, a hierarchical "macrostructure" is built up that is an important part of the reader's mental representation since it comprises the main ideas of the document [24]. Empirical studies of linear text indicate that establishing coherence at a local and global level is facilitated when a document is set out in a well-defined structure and provides rhetorical cues reflecting its structural properties [24].

Applying this result to hyperdocuments implies that authors should provide cues for both types of coherence at two levels, i.e., at the node level (within nodes) and the net level (between nodes). At the *node level,* authors can rely on their usual writing skills. For instance, they can increase local coherence by explicitly linking clauses with conjunctions and they can increase global coherence by aggregating sentences into paragraphs or chapters. At the *net level* however, authors need skills that go beyond those for writing linear text. To establish coherence between nodes, they have to provide cues in hypertext that parallel cues for both local and global coherence in traditional text.

In order to *increase local coherence* at net level, authors should limit "the fragmentation characteristic of hypertext" [16, p. 22]. This characteristic seems to be endemic to hyperdocuments and results from the segmentation of information into disjunct nodes and their display in separate windows. Fragmentation may result in a lack of interpretative context and thus lead to the impression that the hyperdocument is an aggregation of loosely linked pieces of information rather than a coherent whole. Authors can take two measures to reduce this impression:

- They can represent semantic relationships explicitly between nodes that indicate what the contents of the nodes have to do with each other. In this respect, a link between two nodes in a graphical browser can be regarded as fulfilling a function analogous to a conjunction in a linear text and hence its label should indicate the appropriate semantic relation.
- They can provide a context in which the actual node is displayed together with its predecessor. The preservation of context conveys a sense of continuity across nodes that is very important for comprehension. In attempting to understand the content of a new node, readers try to extract its information and relate it to the content of other nodes they have visited. In psycholinguistics, this activity is regarded as a basic process of comprehension, called "given-new-strategy" [2]. With respect to hyperdocuments, it facilitates the construction of an integrated mental representation of information units that may be distributed over different nodes. One way to support the given-new-strategy in hyperdocuments is by preserving the content of the predecessor of the current node. When readers can see the "given" information of the previous node together with the "new" information of the current node, they can detect semantic relations between both sources more easily. Thus they are supported in joining the content of both nodes in a common mental representation.

If authors want readers to construct relationships exceeding the level of local coherence, they have to incorporate cues at net level that *increase global coherence.* Such cues should help the reader to identify the major components of the hyperdocument and the way in that these constitute its overall structure. For this purpose, authors should integrate two types of components into their hyperdocument:

- In analogy to such cues as “paragraph” or “chapter” in linear text, they can aggregate information into higher order units—e.g., “composite nodes” [8]. These enable readers to identify important document components at net level and to represent them in terms of a macrostructure. This structure can be regarded as representing the “gist,” or essence, of a document since it captures the major components and their relations.
- For reducing the mental effort of comprehension, it is not sufficient to simply impose a coherent structure on a document; it is also necessary to convey that structure to the reader. This can be accomplished most efficiently by providing a comprehensive overview of the document components and their relations in terms of graphical maps or browsers.

Providing a means for structuring, overview, and reduction of fragmentation will significantly increase the coherence of a hyperdocument. This will facilitate the construction of a mental model in the course of reading and thus lead to better understanding. However, more can be done to support comprehension by additionally reducing the negative influence of “cognitive overhead” [3].

Cognitive Overhead

With respect to reading a hyperdocument, Conklin characterized cognitive overhead as “the additional effort and concentration necessary to maintain several tasks or trails at one time” [3, p. 40]. The reason for cognitive overhead lies in the limited capacity of human information processing [13]. Every effort additional to reading reduces the mental resources available for comprehension. With respect to hyperdocuments, such efforts primarily concern orientation, navigation and user-interface adjustment.

The terms “orientation” and “navigation” imply the conception of hyperdocuments as spaces (see also Dieberger and Bolter as well as Marshall and Shipman in this issue) where readers can move from one piece of information to the other. This “travel metaphor” certainly contributed to characterizing the most cited problem of hypertext readers as “disorientation.” It occurs when readers do not know where they are, how they got there or where they should go next.

Readers need knowledge about the overall document structure and must keep track of their moves through that structure. Even for smaller hyperdocuments this can result in a considerable memory load if no external orientation cues are given. In order to provide cues that appropriately capture the net-like structure of most hyperdocuments, authors may employ graphical presentation formats that give a visual impression of the “information space.” An example of how this can be accomplished for World-Wide Web documents is given in Kahn, appearing in this issue. In general, such orientation cues should enable readers:

- To identify their current position with respect to the overall structure;
- To reconstruct the way that led to this position; and
- To distinguish among different options for moving on from this position.

A visual presentation fulfilling these requirements

Table 1. Design principles and their relation to cognitive design issues.
Issues marked by “+" are of primary concern for the principle.

Principles	Design issues addressed
P1: Use typed link labels	I1+ I2
P2: Indicate equivalencies between information units	I1 I2+
P3: Preserve the context of information units	I1 I2+
P4: Use higher order information units	I3+
P5: Visualize the structure of the document	I1 I2 I4+ I8 I9
P6: Include cues into the visualization of structure which show the reader’s current position, the way that led to this position and navigational options for moving on	I5+ I6+ I7+
P7: Provide a set of complementary navigation facilities which cover aspects of direction and distance	I8+ I9+
P8: Use a stable screen layout with windows of fixed position and default size	I10+

also should provide an overview of the document structure to increase global coherence. Besides simplicity of the user interface, an important argument supporting this claim is the close relation between comprehension and orientation. Empirical studies summarized in [5] revealed a correlation between comprehension and memory for location. One interpretation of this result is that memory for content and memory for spatial information are different aspects of the same mental representation, i.e., the reader's mental model. Hence, all factors that facilitate the construction of such a model by reducing mental effort or increase a model's quality by improving its completeness and consistency can be expected to affect both comprehension and orientation.

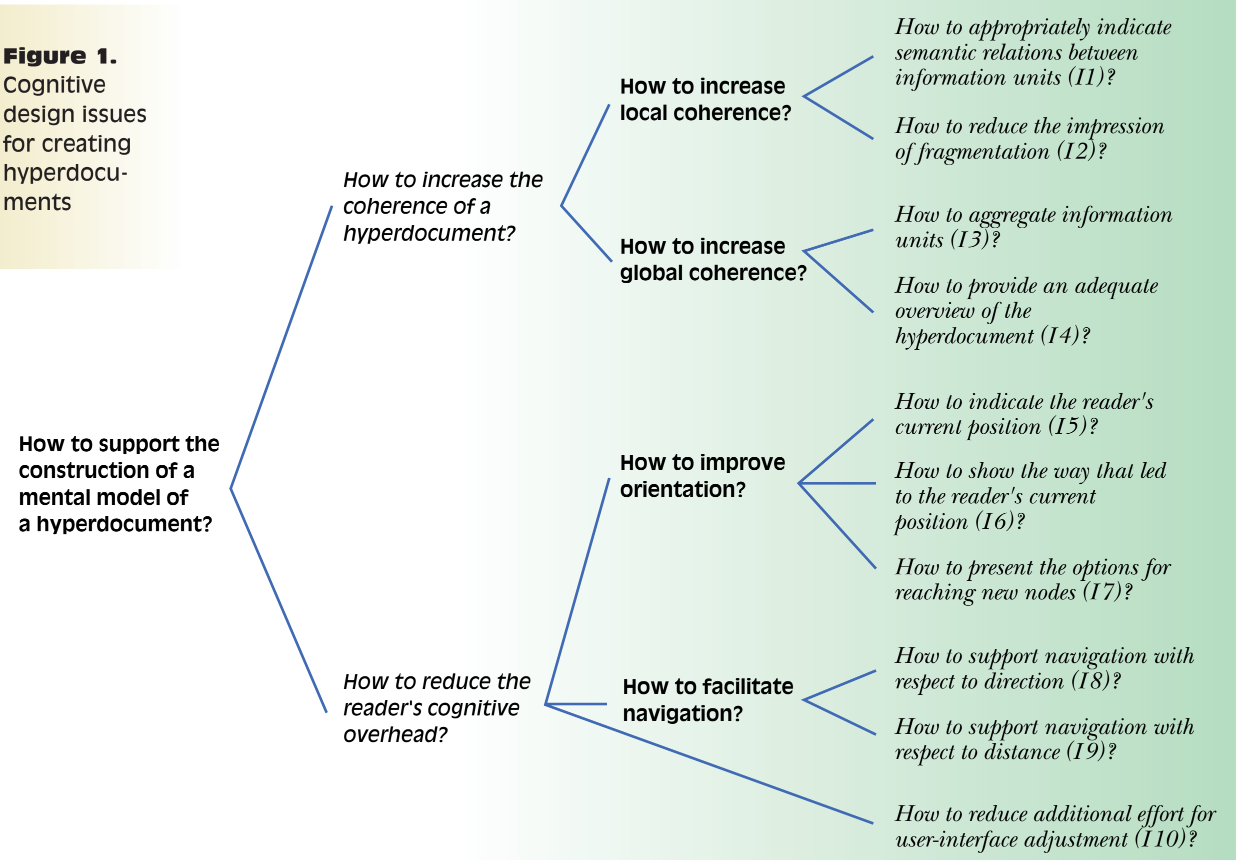
While orientation facilities are meant to help readers *find* their way, navigation facilities enable readers to actually *make* their way. To reduce the readers' effort when acting on their navigational decisions, authors should provide facilities that cover an adequate spectrum of possible movements without enforcing complicated series of actions. Two aspects of navigation are crucial in that respect: *direction* and *distance*.

With respect to *direction*, one can distinguish forward and backward navigation. While forward navigation occurs when readers seek new information by moving to a node they have not opened yet, backward navigation occurs when they try to find old information by moving to a node they have already visited. If the document structure has the shape of a layered net (resulting from the nesting of composite nodes), both types of movements can also include navigation on the vertical dimension: readers move up when an action carries them to higher layers and they move down to lower layers of the document.

With respect to *distance*, one can distinguish between steps and jumps. In making a step, readers simply follow links, that is, move from the current node to a node that is directly linked to it. In making a jump, readers reach a node that is not immediately connected to their current position. This is particularly important for backward navigation, that is, when readers have "traveled" some distance and want to get back to a specific node that is no longer visible.

In addition to supporting orientation and navigation, an adequate interface for hyperdocuments also has to cope with the third potential source of cognitive overhead: *user-interface adjustment*. Effort required for this activity may be influenced by various interface features. Examples include the need to move, resize, or manually close windows on the screen, and the necessity to switch from one presentation format to another (e.g., from the presentation of contents to the presentation of structure). A number of empirical studies demonstrate the effects of such features on various kinds of user performance. For example, several experiments investigating the impact of different window layouts showed that tiled windows—as opposed to

Figure 1. Cognitive design issues for creating hyperdocuments



overlapping windows—are easier to use and lead to higher accuracy and speed in accomplishing certain tasks [1]. To minimize the effort for interface adjustment, an author should carefully consider which of such activities are really indispensable and which can be avoided by a more appropriate user-interface design.

Cognitive Design Issues and Principles

In summary, the readability of hyperdocuments can be improved by supporting the construction of a mental model in terms of a dual approach:

- Authors can *increase document coherence,* thus facilitating the construction of semantic relations between information units.
- Authors can *reduce cognitive overhead,* thus freeing processing capacities that otherwise would have been bound by orientation, navigation, and user-interface adjustment.

As discussed in the previous section, this approach leads to a number of design problems authors must solve to support readers in their attempts to understand hyperdocuments. These problems can be formulated as *cognitive design issues* and represented as a hierarchy (see Figure 1).

Figure 1 shows that the overall design issue of "how to support the construction of a mental model of a hyperdocument" can be broken down into a number of subissues. The most specific subissues (I1 to I10) are shown on the right side of the figure. As answers to these issues, we propose *eight design principles.* These principles can enhance design methodologies[1], such as PHD [19] or RM (Isakowitz et al. in this issue), and design models, such as HDM [7] or OOHDM (Schwabe and Rossi in this issue). They may serve as guidelines for hypermedia authoring and as a basis for deriving evaluation and quality criteria supplementary to those discussed by Garzotto et al. and Johnson (both in this issue). Our eight principles and their relation to the specific design issues of Figure 1 are listed in Table 1. The eight principles are our answers to the cognitive design issues in Figure 1.

Typed link labels (P1) should be used to represent semantic relations between information units [23]. For instance, labels such as "explain" point out that one node contains an explanation for the content of another node. *Indicating equivalencies* (P2) helps to reduce the impression of fragmentation. Equivalencies may arise when tokens representing the same node are displayed in different windows simultaneously. Fragmentation also is addressed by *preserving the context* of a node (P3), that is, by showing neighboring nodes in the document structure and their relationships.

While the first three principles aim to increase local coherence, the fourth and fifth principles concern global coherence. *Higher-order information units* (P4), such as composite nodes [8], can be used to aggregate information thus lending more structure to the document. *Visualizing this structure* (P5) provides a most useful overview and helps the reader to identify major topics and their relations. Together, both principles support the reader's attempt to build a macrostructure and thus increase coherence at a global level.

In addition, visualizations of structure, such as graphical maps or browsers, improve orientation and navigation. In combination with *cues* for the reader's current position, recent path and further navigational options (P6), they form a suitable basis for orientation. By integrating specific navigation facilities to cover aspects of *direction and distance* (P7), maps and browsers can support navigation. Finally, a *stable screen layout* (P8) reduces the effort for interface adjustment, and together with principles five to seven, aims to diminish cognitive overhead.

The extent to which the eight guidelines can be followed is not only constrained by the potentials of the reading environment, but also by the functionality and data objects of the authoring environment that is employed. In our own work, we have used the SEPIA system [21, 22]. The sidebar by Streitz in this issue shows a version of this system that supports cooperation between multiple authors.

SEPIA includes a *construction kit* of objects for coherent hyperdocument structuring and presentation [23]. The kit provides different types of nodes and links, and is enhanced by a taxonomy of link labels. Basic node types are atomic nodes to represent text, graphics, video, etc., and different kinds of composite nodes to aggregate nodes and links, thus enabling authors to build layered hypertext nets. Basic link types are point-to-point links and embedded links. Documents that are created with the construction kit can be automatically mapped to SEPIA's presentation interface SPI [9]. The result can be regarded as one way—among many others—in which a hyperdocument can be structured and presented according to the principles we have proposed. The sidebars by Kahn and Balasubramanian et al. in this special section, for example, apply some of these principles to Mosaic.

SPI: SEPIA's Presentation Interface

To illustrate the main features of SPI, we use a hyperdocument that was developed in the preparation phase of the POLIKOM research program [10]. In this program, a spectrum of telecooperation systems will be developed to support collaboration between distributed government agencies and ministries. The example document represents the debate about designing the parliamentary area in Berlin, the capital of Germany. It contains descriptions of submitted proposals by different architects, of requirements, and of constraints leading to a

---

[1] Recent approaches to methodologies for hypermedia design and development are given in [11] and [18].

complex net of issues, subissues, contrary positions, and opposing arguments. Based on this information, a final recommendation is made and justified. The document was created from existing linear documents by using design objects and naming conventions of SEPIA's construction kit. Mapping the document from SEPIA's authoring environment to its presentation interface SPI results in a hyperdocument that adheres to our eight design principles.

The interface is illustrated in Figure 2. It provides several navigation and help facilities: the buttons "Navigator" and "System Info" at the top of the screen, and the arrow-shaped buttons at the bottom right portion of the screen. The screen is partitioned along two dimensions: On the horizontal dimension, it separates *structural information* on the left side from *content information* on the right. On the vertical dimension, it distinguishes currently *active nodes* in the bottom area from their immediate *predecessors* displayed in the top area. This results in two "structure windows" and two "content windows" with fixed location and a default size.

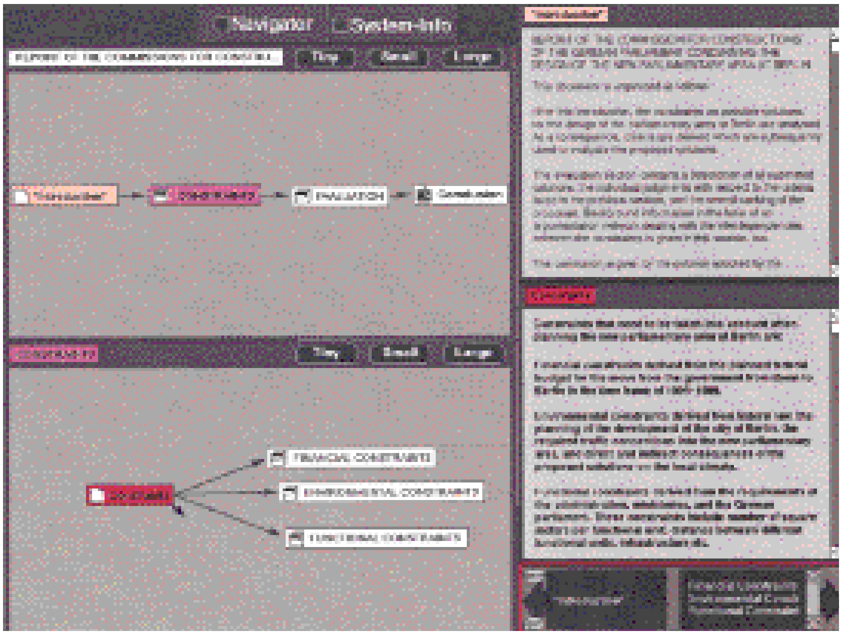


Figure 2. SEPIA's presentation interface (SPI)

Structure windows display nodes and links created with SEPIA's construction kit. Since structures built from the kit may be pretty large, both windows can be zoomed by pressing the buttons "tiny," "small" and "large" of the top panel. *Content windows* show text, graphics, pictures, etc. contained in atomic nodes. They are equipped with a scroll bar and display the node name in a field separated from the text so that it remains present when the content is scrolled.

The most crucial characteristic of SPI is the *integrated presentation* of structure and content that results from the combined application of our eight principles. To illustrate the impact of these principles on the design of SPI, let us discuss the relation of each of them to the corresponding interface features.

Displaying Typed Links

SPI displays the names of typed links (P1). SEPIA's construction kit supports linking by a taxonomy of link types [23]. Link names are verbs that can be used to represent semantic relationships between connected information units as short sentences. In Figure 3 for example, the link "supports" between the nodes "Budget" and "Small solution" constitutes a meaningful phrase that helps the reader to construct a semantic relation between both nodes. When authors employ the taxonomy and label their nodes accordingly, they can create comprehensible structures of high local coherence at net level.

Indicating Equivalencies

SPI indicates *equivalencies* of information units (P2). The combined presentation of structure and content entails the necessity to indicate which content displayed in a "content window" corresponds to a graphical placeholder in a "structure window." Besides identical names, SPI visualizes such equivalencies by identical colors. In Figure 2, for example, the actual composite node "Constraints" in the top structure window and the label "Constraints" of the bottom structure window are both orange. This points out that the bottom window displays the content of the orange node contained in the top window. Further equivalencies exist between graphical browsers and content windows. For example, the currently activated atomic node "Constraints" is represented by the red rectangle in the bottom structure window and by the red label of the bottom content window.[2] This indicates that the right window displays the content of the red node in the left window.

Colors in SPI help to detect equivalencies at first glance. They increase the coherence of a hyperdocument at net level and supplement the linguistic cues visually. In addition, colors are suitable means to support orientation (see P6 later in this article) and therefore also contribute to the reduction of cognitive overhead.

---

[2] Note that in this example the label "Constraints" is used twice: once as name for a composite and once as name for an atomic node that serves as starting point for reading the sequence of nodes in that composite.

Preserving Context

SPI preserves the context of active nodes (P3). For both atomic nodes and composite nodes, preservation works according to a simple principle: The window at the bottom presents the currently activated node while the window at the top presents the previously activated node. Whenever a new node is opened, it is displayed in the bottom window and the former content of this window moves to the top.

The reader can scroll both windows simultaneously and watch the contents of both. This feature of SPI aims to increase coherence by supporting the "given-new-strategy" of reading. Since new information and previous information are visible simultaneously, it is much easier for the reader to construct meaningful relations between both information units. This construction reaches across node boundaries and thus helps to reduce the impression of fragmentation.

Context also is preserved for *composite nodes.* In Figure 2, the upper-left window displays the context of an activated composite node ("Constraints") and the lower-left window displays its internal structure. The relation between both windows is analogous to the relation between the two content windows: while the bottom window presents the structure in which the reader is actually located, the top window presents its predecessor. With respect to composite nodes, the preservation of context aims to increase the global coherence of hyperdocuments. It helps the reader recognize which nodes are contained in a composite and thus facilitates the mental representation of the document structure.

Presenting Higher-Order Information Units

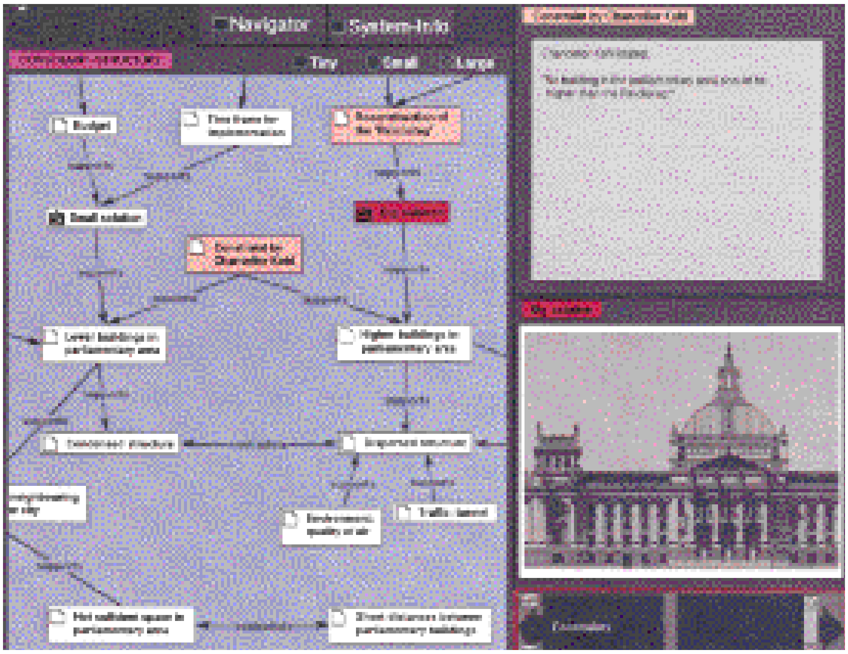
SPI displays higher-order information units (P4). These are two different kinds of composite nodes provided by SEPIA's construction kit: *sequencing nodes* and *exploration nodes.* Each represents a specific part structure and entails a different type of browsing behavior. For their presentation, we distinguish between two graphical modes, called pathview and netview.

In the *pathview,* sequencing nodes are presented (see Figure 2). These nodes give authors the opportunity to define sequential, branching, and conditional paths [25]. While sequential and branching paths are static, conditional paths are dynamic. They depend on the reader's previous actions, that is, at a specific point in a conditional path, the reader cannot reach all next nodes, but only a computed subset.

In the *netview,* exploration nodes are presented (see Figure 3). They consist of a net of nodes and links that can be visited in any order. Links have no impact on navigation, but are exclusively used to indicate semantic relations by their labels. If the exploration net contains no composite nodes, the lower-left window is automatically expanded and covers the complete left side of the screen. The reason for this design decision is obvious from our example in Figure 3. Since the exploration node does not contain different layers that have to be displayed, the space of the former upper-left window can be used to show a larger proportion of the net.

The combination of both views offers interesting possibilities with respect to tailoring a document to the needs and particularities of a specific readership. Paths enable the author to guide a reader in the way

Figure 3. Exploration node displayed in the netview



the author considers best—suggesting the reader ought to use them whenever it is believed that parts of the document should be read in a specific order. In contrast, exploration nodes enable the author to give readers the freedom of leaving a predefined sequence of nodes. This opportunity could, for instance, be provided to access information that is not central, but offers interesting background information. Employed in this way, pathview and netview allow for distinguishing between central facts arranged in a coherent sequence and additional information that can be freely explored.

SPI supports this distinction by an important constraint: Exploration nodes can only be reached from the pathview by activating a link embedded within a sequencing node. Such "exploration links" are indicated by a short blue arrow (see "Constraints" in Figure 2, for example) and their traversal opens an exploration node displayed in the net view. When finished with the exploration, the reader is automatically taken back to the anchor of the exploration link. This feature of SPI reestablishes the context of reading sequences after exploration and ensures the readers can continue to follow the coherent path from where they left it.

Presenting Structure

SPI visualizes the document structure (P5). Due to the limitations of the computer screen, visualizations of structure, such as maps or graphical browsers, have to cope with two problems: document complexity and document size.

The node types provided by SEPIA's construction kit allow the construction of nested paths and layered nets. These structures may grow rather complex, since each composite node can contain another path or net. To reduce the complexity of the structure presented, a specific type of fisheye view [6] is implemented in SPI's structure windows. It constrains visualization to those parts of the overall document structure most relevant with respect to the reader's actual point of interest: the structure surrounding the reader's current node and its predecessor.

Since SEPIA does not limit the amount of nodes and links in a composite, the number of information units in both structure windows can be rather large. Therefore, both windows provide a zooming functionality. By pressing one of the three buttons at the top panel ("tiny," "small," "large"), the internal scale of the window can be enlarged or diminished. Additionally, the reader can move the whole graph in any direction to make hidden parts of the structure visible.

These features of the structure windows enable the reader to view all parts of the document structure that are relevant with respect to the reader's current location. Since both structures and the actual node text are simultaneously available, the reader is supported in relating content and structure. Thus the impression of fragmentation is reduced and the construction of a coherent mental model of the document is facilitated.

Orientation Cues

SPI integrates orientation cues into the visualization of structure (P6). This is accomplished by using four different colors in the graphical browsers (see Figure 2):

1. *Red* indicates the reader's actual atomic node (e.g., "Constraints" in both bottom windows).
2. *Pink* indicates the node that has been visited immediately before, but is no longer activated (e.g., "Introduction" in both top windows).
3. *Orange* indicates the reader's actual composite node (e.g., "Constraints" in the left top window).
4. *White* indicates all nodes that have not been opened yet (e.g., "Financial Constraints" in the left bottom window).

The consistent variation of colors helps readers to see where they are (red or orange), where they have been (pink), and where they can go for new information (white). These visual cues make crucial information for navigation accessible "at a glance." Thus they help readers to keep track of their moves without any additional effort and reduce the memory load required for orientation.

Navigational Facilities

SPI provides a set of complementary navigational tools addressing direction and distance (P7). The set consists of three facilities: a button panel, the graphical browsers in the structure windows and a tool called the "Navigator."

The *button panel* is made up of two buttons located at the bottom of the user interface. Clicking on the right button moves the reader one step forward; clicking on the left button moves the reader one step back. Both buttons can handle multiple navigational options such as branches in a path. When the reader arrives at a branching point, the forward button displays a list of all alternative next nodes (see Figure 2). If the reader has already passed a branching point, the back button displays the name of the prior node and the name of the most recent branching point. This arrangement allows the reader to return directly to the most recent branching and is extremely useful if the reader wants to jump back and take another route.

Graphical browsers support navigation in the two structure windows. Readers navigate by clicking on nodes and are constrained by the kind of composite node that is currently active. The content of a sequencing node is displayed in the pathview, i.e., the reader has to follow predefined paths step by step. In the browser of Figure 2, for example, the reader could reach the node "Constraints" only after visiting its predecessor "Introduction." The content of an exploration node is displayed in the netview (see Figure 3), i.e., the reader can jump to any node at any time simply by clicking on it.

The Navigator is a history function that handles the structural properties of layered nets and nested paths by providing three kinds of information:

- It shows the history of a reading session by chronologically listing each node that has been visited during a session, thus enabling readers to easily reconstruct their paths. Atomic nodes in an exploration node are marked as "explored" while all other nodes are simply listed.
- It shows the currently activated atomic node that is simply denoted by the last name on the list.
- It shows the number of hierarchical levels of the document and the reader's current position within that hierarchy. The number of levels is given by a scale at the top and the bottom of the Navigator. In Figure 4, this scale indicates that the presented document has four levels (starting at level 1). The name of each node in the list is indented in order to indicate the level to which the node belongs. For example, "Constraints" is at level 1 while "Evaluation" is at level 2. Thus readers can immediately recognize their position in the hierarchy and know how much deeper they can go.

Movements within the Navigator are accomplished by simply clicking on a name in the list, that is, readers can step or jump back to any node they have already visited. Since such a node can be located at a different level of the document structure, the Navigator supports upward and downward moves.

Together, the button panel, graphical browsers, and the Navigator provide a set of complementary navigation facilities that cover all relevant aspects of direction and distance. They help to avoid cognitive overhead since without adequate direction and distance support, the user would have to resort to awkward maneuvers and clumsy detours.

User-Interface Adjustment

SPI has a stable screen layout with windows of fixed position and default size (P8). This stability relieves the reader from the burden of opening, closing, positioning, and resizing objects displayed in SPI. However, the interface is not inadjustable, that is, the reader can (re)scale the size of nodes and links displayed in structure windows.

Another advantage lies in the simultaneous availability of content and structure information. There is no need for the reader to look for maps and tables of contents or to switch from text to structural overviews and back again. Since both types of information are continuously present, such activities become superfluous and cognitive overhead for accessing structural information is avoided.

Summary and Future Research

Two factors are particularly crucial for increasing the readability of hyperdocuments: coherence as positive influence and cognitive overhead as negative influence on comprehension. Since both factors are likely to affect the reader's construction of a mental model, readability can be improved by a dual approach: On the one hand, authors can increase the coherence of their document thus facilitating comprehension. On the other hand, they can reduce cognitive overhead thus freeing information processing capacities that otherwise had to be used for orientation, navigation and user-interface adjustment.

To support authors in following this dual approach, we proposed eight principles that provide solutions to a number of cognitive design issues that follow from the two factors. These principles can be met by a variety of different user interfaces and in fact a number of systems, such as SuperBook [14], Aquanet [15] and gIBIS [4], share important features with SPI. However, none of them starts from an analysis of cognitive design issues and realizes our design principles in total. What distinguishes SPI from other interfaces is its cognitive foundation and the explicit integration of components for increasing coherence with means for reducing cognitive overhead that follows from this approach.

SPI is particularly tailored to support learning from hyperdocuments with a structure that is intentionally defined by a single author or an authoring team. However, many of its features also should help readers to explore hypermedia applications with a structure that is continuously growing out of the contributions of many, independently acting authors. The best known example of such a "browsable hyperbase" evolving over time

Figure 4.
The Navigator

is the World-Wide Web. This global information space offers a tremendous amount of information that is both fascinating and incredibly complex. Since navigation and orientation in the Web suffer from the lack of overviews and context (see the sidebars by Kahn and V. Balasubramanian et al. in this issue), SPI features, such as context preservation, visual orientation cues and graphical maps, might be helpful enhancements to interfaces such as Mosaic.

Until now, we have not investigated the adequacy of SPI in systematic studies, but first experiences with several readers support the assumption that our design decisions increase comprehensibility and reduce the effort required for orientation and navigation. In this respect, the graphical browsing facilities and the Navigator are particularly appreciated by readers. Of course, anecdotal evidence is insufficient and more research is required to prove the correctness of our underlying theoretical model of comprehension, the value of the design principles, and the effectiveness of SPI as a corresponding interface. Currently, an evaluation study is under way that will provide deeper insights on an empirical basis and give justified answers to our question of how to design for comprehension.

References

1. Bly, S.A., and Rosenberg, J.K. A comparison of tiled and overlapping windows. In *Proceedings of CHI '86*. ACM Press, New York, 1986, pp. 101–106.
2. Clark, H.H., and Haviland, S.E. Psychological processes as linguistic explanation. In D. Cohen, Ed., *Explaining Linguistic Phenomena*. Hemisphere, Washington, 1974, pp. 91–124.
3. Conklin, J. Hypertext: An introduction and survey. *IEEE Comput. 20*, 9 (Sept. 1987), 17–40.
4. Conklin, J., and Begeman, M.L. gIBIS: A hypertext tool for team design deliberation. In *Proceedings of Hypertext ´87,* ACM Press, New York, 1987, pp. 132–143.
5. Dillon, A., McKnight, C., and Richardson, J. Space–The final chapter or why physical representations are not semantic intentions. In C. McKnight, A. Dillon and J. Richardson, Eds., *Hypertext–A Psychological Perspective*. Ellis Horwood, New York, 1994, pp. 169–191.
6. Furnas, G.W. Generalized fisheye views. In *Proceedings of CHI '86,* ACM Press, New York, 1986, pp. 16-23.
7. Garzotto, F., Paolini, P., and Schwabe, D. HDM–A model-based approach to hypermedia application design. *ACM Trans. Info. Syst. 11,* 1 (Jan. 1993), 1–26.
8. Halasz, F.G. Reflections on NoteCards: Seven issues for the next generation of hypermedia systems. *Commun. ACM 31*, 7 (July 1988), 836–852.
9. Hannemann, J., Thüring, M., and Haake, J.M. Hyperdocument Presentation: Facing the Interface. *Arbeitspapiere der GMD, 784,* GMD, St. Augustin, 1993.
10. Hoschka, P., Butscher, B., Streitz, N.A. Telecooperation and telepresence: Technical challenges of a government distributed between Bonn and Berlin. *Informatization and The Public Sector 2* (1992) 269–299.
11. Isakowitz, T., and Thüring, M., Eds. Methodologies for designing and developing hypermedia applications. In *Proceedings of the Workshop on Hypermedia Design and Development at the ACM European Conference on Hypermedia Technology (ECHT '94),* Working Paper Series Stern # IS-94-8, New York University, New York.
12. Johnson-Laird, P.N. Mental models. In M.I. Posner, Ed., *Foundations of Cognitive Science*. MIT Press, Cambridge, MA, 1989, 469–499.
13. Kahneman, D. *Attention and Effort*. Prentice-Hall, Englewood Cliffs, N.J., 1973.
14. Landauer, T., Egan, D. Remde, J., Lesk, M. Lochbaum, C., and Ketchum, D. Enhancing the usability of text through computer delivery and formative evaluation: The SuperBook project. In C. McKnight, A. Dillon and J. Richardson, Eds., *Hypertext–A Psychological Perspective*. Ellis Horwood, New York, 1994, 71–136.
15. Marshall, C.C., Halasz, F.G., Rogers, R.A., and Janssen, W.C. Jr. Aquanet: A hypertext tool to hold your knowledge in place. In *Proceedings of Hypertext'91*, ACM Press, New York, 1991, pp. 261–275.
16. Marshall, C.C., and Irish, P.M. Guided tours and online presentations: How authors make existing hypertext intelligible for readers. In *Proceedings of Hypertext´89*. ACM Press, New York, 1989, pp. 15–26.
17. Nielsen, J. *Usability Engineering*. Academic Press, Boston, 1993.
18. Schuler, W., Hannemann, J., and Streitz, N.A., Eds. *Designing User Interfaces for Hypermedia*. Springer, Heidelberg, 1995.
19. Schuler, W., and Thüring, M. *Pragmatic hypertext design (PHD).* Arbeitspapiere der GMD, 813, GMD, St. Augustin, 1994.
20. Stotts, P.D., and Furuta, R. Hypertext 2000: Databases or documents? *Electronic Publishing 4*, 2 (June 1991), 119–121.
21. Streitz, N.A., Hannemann, J., and Thüring, M. From ideas and arguments to hyperdocuments: Travelling through activity spaces. In *Proceedings of Hypertext´89* , ACM Press, New York, 1989, pp. 343–364.
22. Streitz, N.A., Haake, J.M., Hannemann, J., Lemke, A., Schütt, H.W., Schuler, W., and Thüring, M. SEPIA: A cooperative hypermedia authoring environment. In *Proceedings of the ACM Conference on Hypertext (ECHT 92),* ACM Press, New York, 1992, pp. 11–22.
23. Thüring, M., Haake, J.M., and Hannemann, J. What's ELIZA doing in the Chinese Room? Incoherent hyperdocuments–and how to avoid them. In *Proceedings of Hypertext´91*. ACM Press, New York, 1991, pp. 161-177.
24. van Dijk, T.A., and Kintsch, W. *Strategies of discourse comprehension*. Academic Press, Orlando, 1983.
25. Zellweger, P.T. Scripted documents: A hypermedia path mechanism. In *Proceedings of Hypertext´89*. ACM Press, New York, 1989, pp. 1–14.

About the Authors:

MANFRED THÜRING is executive manager of the Institute for Organisation and Automation (BIFOA) at the University of Cologne. Current research interests include cognitive science, hypermedia application design and computer-supported cooperative work. **Author's Present Address:** BIFOA, Universitätsstr. 45, D-50931 Köln, Germany; email: thuering@bifoa.uni-koeln.de

JÖRG HANNEMANN works as project manager at the Institute for Organisation and Automation (BIFOA) at the University of Cologne, specializing in hypermedia and CSCW. Current research interests include the cognitive foundation of authoring environments and the design of hypermedia applications. **Author's Present Address:** BIFOA, Universitätsstr. 45, D-50931 Köln, Germany; email: jhanne@bifoa.uni-koeln.de

JÖRG M. HAAKE is head of the OCEAN department of the German National Research Center for Computer Science and has been responsible for the development of hypermedia-based cooperative systems. Current research interests focus on hypermedia design and architectures for open, collaborative hypermedia systems. **Author's Present Address:** GMD-IPSI, Dolivostr. 15, D-64293 Darmstadt, Germany; email: haake@darmstadt.gmd.de; WWW: http://www.darmstadt.gmd.de/publish/ocean/people/haake.html