A model for hypertext authoring based on accessibility

Arrie van Berkel
Groningen

After a short survey of the problems hypertext authors have to deal with when they take the users' navigation and comprehension difficulties into consideration, it will be evident that an author should have permanent access to the mental basics of the textual architecture. Therefore the authoring process will be divided into three basic components, Tasksetter, Planner and Editor, which are interactive in the sense that they are bi-directionally linked to each other. As a transitional component from Planner to Editor we introduce Idea-nodes, which is a network of ideas generated from the Planner that can be directly linked with nodes in the Editor. The same holds for the (standard) structural component. This component can also be linked to the Editor as a homepage. In this way a hypertext author has optimum accessibility to structural and conceptual information.

1 Introduction: hypertext structure and navigation

The production of an interactive hypertext is not only the decomposition of linear text or adding simple connections between informative textual units. Its effectiveness is highly dependent of the structure or architecture of the hypertext. Without an explicit structure the user is not able to navigate efficiently, which prevents her or him from building up a mental representation of the information that s/he is looking for. The hypertext and especially its navigational structure is the user interface of the program that is behind it. So the author has the obligation to create an interface that is user-friendly in the sense that it enables the user to make a structured mental representation of the information (cf. Van Berkel/Lappenschaar 1993, 30). Research in human-computer interaction shows that users are being helped by a user interface design that will provide users to adapt the model of the system tasks and the obtained information to their cognitive model. From this it follows that a hypertext author must rely on a users' mental model to build a hypertext. The author has to create a system that "teaches" the user how to execute the tasks (cf. Van Berkel/Lappenschaar 1993, 25).

A hypertext architecture can be structured in different ways such as tree, pyramid and web structures, in which the separate textual elements can be linked. According to Shneiderman (1989, 24) the author's primary task is to build a textual architecture on which the user can rely from the start. Such a root document or introductory article must be presented in such a way that the user is able to create a mental picture of the structure, the extent and the range of the information in the hypertextual environment. The actual users need to know what is and what is not in the textual database. Shneiderman proposes three models for such a root document (1989, 126):

> 1. a survey of relations with all main concepts: the glossary strategy
> 2. a hierarchic approach in which the relations in the root document are the main categories: the top-down strategy
> 3. the root document is a table of contents of the main concepts: the menu strategy.
> In websites the root document is either the homepage or it follows immediately after the home page. One of the author's tasks is to ground the architecture on one of those strategies.

A hypertext program (e. g. an HTML-editor) supposes that users know which and how much information they have to select and how to execute a search. Nevertheless it is well-known that users get lost in hyperspace. These navigation problems may be caused by several factors (cf. Van der Geest 1993, 58):

> 1. the user does not know which path to follow or what option will be the next
> 2. the user does not know which path leading to the desired information to choose or the user forgets his search goal during navigating
> 3. the user misses the necessary indications to interpret and comprehend the content of the document; the user misses titles, headers, tabs, anchors, buttons, and other structure and navigation indicators.

Hypertext users are often unable to create a mental picture of the information that has been ordered in a nonlinear way. This also has to do with a wrong estimation of the users' preknowledge that would help them to integrate information into existing knowledge or the processing of a new mental scheme. Therefore a hypertext must help its users by activating their preknowledge. Thus a hypertext author has to solve many problems in a different way to that which writers of linear texts are used to. The nonlinearity of hypertext authoring requires more access to the metastructural and metaconceptual framework: the architecture.

2 Accessibility during the authoring process

Nonlinearity implies recursivity in planning, formulating and revising. We assume that the abstract representation and data processing in the process of planning differ from the textual realization of the formulating and the revising processes. It follows that a hypertextual architecture must be divided in a planning device and an editing device. According to Flower and Hayes (1980) we have to deal with a task environment which includes general information about goals, audience, and themes are. For hypertext authoring we propose an accessible component, the Tasksetter, in which general information about the architecture is also included. The accessible components for the authoring process, Tasksetter, Planner, and Editor will be based on the model as represented in Figure 1.

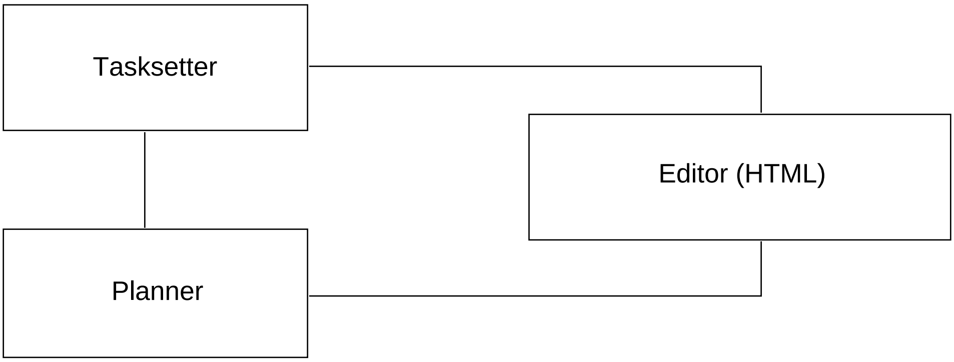

Fig. 1: Accesible Components for the Authoring Process

2.1 The Tasksetter

In the task environment the author must rely on what the authoring tasks imply. Task definitions generate goals for the authoring process. Furthermore, the author must formulate rhetorical goals for local goals in hyperdocuments. During the authoring process an author will check ideas produced thusfar (planning), the texts, anchors, links and hyperlinks produced thusfar and compare them with the goals. The task environment can be supported by answering the following questions: what am I writing about, what do I want to say about it, how can I link the separate units of information and what do I want to achieve and with what motivation. The structure of the authoring environment must give immediate access to the task environment as well as to the Planner and the Editor. From all components of the authoring environment the author must be able to change elements in the Tasksetter. Moreover, the Tasksetter must stimulate the generation of rhetorical definitions and local aims by its accessibility.

2.2 The Planner

The Writing Task Model as proposed by Flower and Hayes (1980 and 1981) distinguishes three subprocesses in the planning phase: generating, organizing and goal-setting. We already mentioned that an author can put her or his goals in the Tasksetter. The Planner only contains specific information concerning the processes of generating and organizing. This enables the author to establish and actualize the internal representation from an abstract level (which is already more or less present in the Tasksetter) to more concrete structural and conceptual information. For practical reasons it would be helpful if the Planner has many graphic features to schematize the organizational or the planned architecture of the hypertext in progress. In order to get explicit elements that are part of the Planner we will go further into some theoretical aspects of the process of idea structuring.

Haas (1990) establishes that writers mostly make notes that are immediately connected with the content of the text. Furthermore it turned out that structural and emphasizing notes were produced during all phases. A good Planner will give access to a configuration of idea-nodes which means that the author will have permanent access to content structured notes and preferably to graphically presented structural notes. Content notes enable authors to sum up and summarize ideas about topics and topic related (external) information. Having permanent access may be used as a tool for getting and structuring information in such a way that continuous feedback from the ongoing authoring process, i. c. writing nodes, attributing types of nodes and linking nodes (building up the architecture) is possible. Each idea-node can be activated from a series of other nodes. Such a node can be imported into the whole architecture and linked to other nodes or menus where it can be elaborated at an appropriate time. This also prevents the author from getting lost in her or his own hyperspace even in the case of nonlinear "patchwork" writing.

Most hypertexts, especially hypertexts that are designed for a World Wide Web site, have a more or less standard superstructure or architecture. This does not count for external hyperlinks because those links have no consequences for the architecture within the site. A standard structure can be enforced by an existing textual database structure or by other standardized textual structures like informative guides of any kind. Other more or less standardized structures are produced by existing tables of contents and indexes. All these structural elements that will be used for structural and semantic linking as part of the whole architecture may be inserted in the Planner. As a permanently accessible tool for the architectural design of a hypertext, the Planner must enable the author to execute the following operations:

(1) instiation and a rearrangement of a new idea-node

(2) flexible adaptation of existing idea-nodes

(3) creation of flexible schemes and (standardized) structures

(4) specification of structural and semantic links between nodes in the hypertext under construction.

Authors have also to deal with problems on the level of formulating, anchoring and linking and hyperlinking. These heuristic problems are strongly connected with the generating and structuring heuristics within the Planner we already discussed.

In the previous paragraph we also discussed the internal organization of the Planner with emphasis on cognitive planning processes concerning organizing and generating. The internal organization of the Planner also provides the accessibility of the Tasksetter and the Editor. Therefore the Planner must be linked with the Tasksetter, as was shown in Figure 1.

2.3 The Editor

Hypertext authoring on the formulating level can be done with an ascii-editor, any word processor or a special HTML-editor using the special HTML-markups. A special editor has the advantage that the markups are already preprogrammed, which makes the HTML-format much easier. But what is more important is that most HTML-editors allow the Tasksetter and the Planner to be integrated in the program itself so that optimal access is possible. Moreover, the HTML-editor provides a helpful tool for anchoring and linking. So it is possible to import or copy a root document or a homepage directly from the Tasksetter or the Planner, which allows a total top-down processing of the hypertext (cf. Van Berkel 1995). Therefore we shall mean HTML-editor when we use the word ‘Editor’.

The Editor should have enough options to revise the structural and conceptional structure according to the flexibility of the Planner and the Tasksetter. Even when the definitions in the Tasksetter are being changed the Editor should be flexible enough to actualize the changes in the text, the anchors and the links. Therefore, as we already mentioned before, the Tasksetter must also be linked with the actual hypertextual page. This allows authors to check their actual authoring against the definitions in the Tasksetter.

Another useful option in the Editor is the linkage with the separate idea-nodes. The idea-nodes can be linked from the Planner as separate nodes. The Planner in this case is a meta-rootdocument. When a homepage is already a copy of the Planner, it is possible to link idea-nodes as separate planning elements that can be worked out as subpages later on (see Figure 2). Thus, the interaction between Tasksetter, Planner and Editor preferably as an integrated system in the Editor leads to an optimal accessibility to all tasks that authors have to perform.

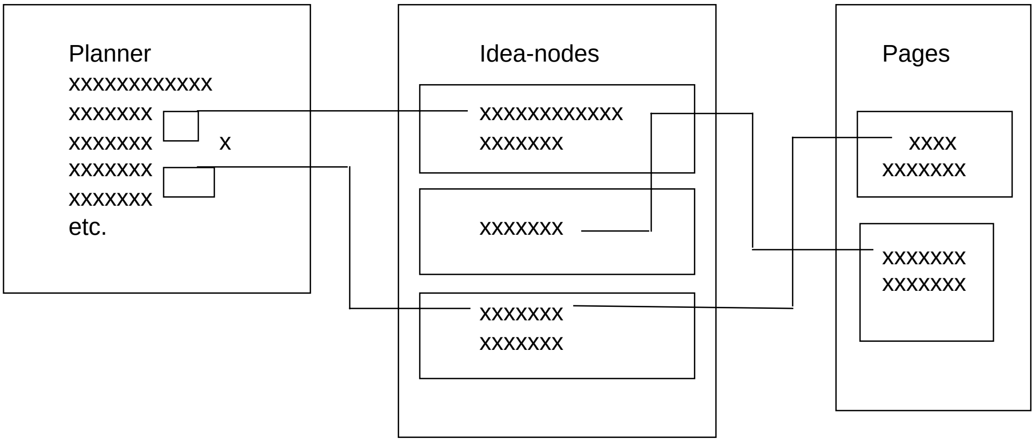

Fig. 2: Linkage between nodes

3 An extended model of accessible authoring components

When we summarize what has been discussed about the accessibility of the different components that can be implemented in a HTML-editor, we get the following list that is also presented as Figure 3:

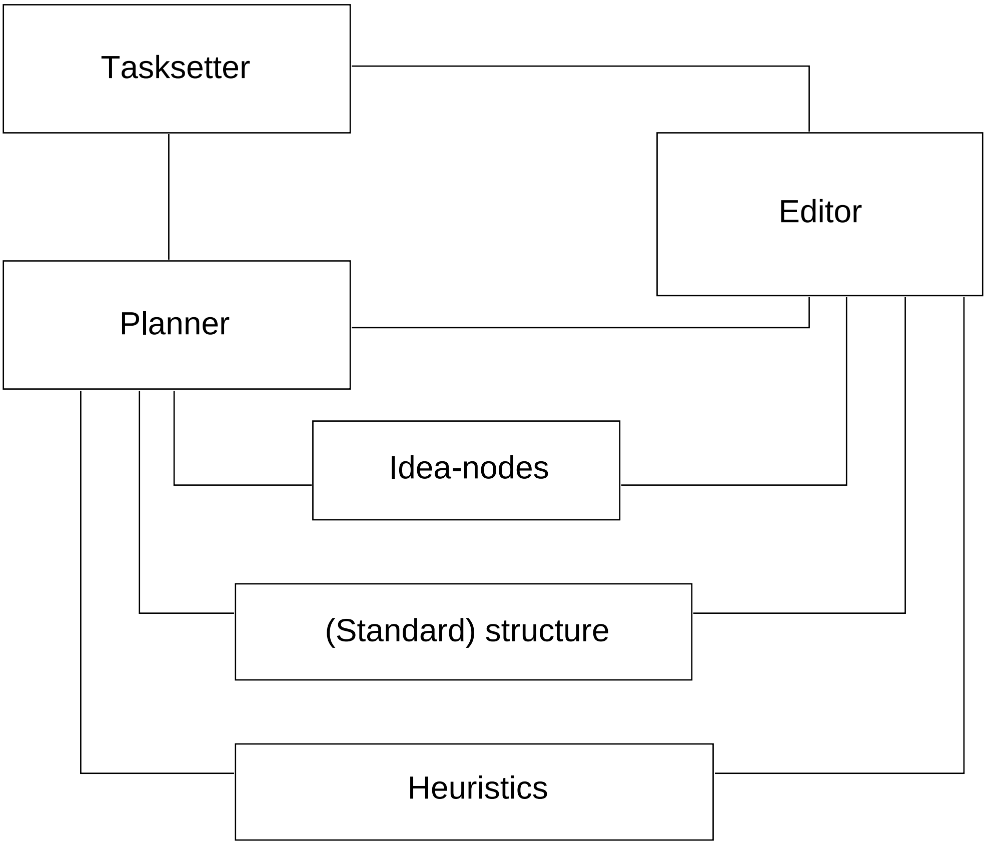

Fig. 3: Model of accessible authoring components

(1) the basic components are Tasksetter, Planner and Editor

(2) the Tasksetter and the Planner are linked with each other

(3) both the Tasksetter and the Planner are linked with the Editor

(4) the Planner is linked with the Idea-nodes in which each idea-node is linked to another

(5) the Planner may generate a (standard) structure which can be linked to the Editor

(6) the whole authoring process is controlled by heuristics.

The processing in Tasksetter, Planner, and Editor must be interactive and recursive which enables an author to get optimum access and control during the authoring process.

References

van Berkel, Arrie (1995): Hypertext as a Tool for Planning in the Writing Process. In: Jakobs, Eva-Maria/ Knorr, Dagmar/ Molitor-Lübbert, Sylvie (Hrsg.): Wissenschaftliche Textproduktion. Mit und ohne Computer. Frankfurt/Main u. a.: Lang, 169-177

van Berkel, Arrie/ Lappenschaar, Bert-Jan (1993): Een taalbeheersingsbijdrage aan de implementatie van een user interface [A language communication contribution to the implementation of an user interface]. In: Tijdschrift voor Taalbeheersing 15, 24-35

Flower, Linda S./ Hayes, John R. (1980): The Dynamics of Composing. Making Plans and Juggling Constraints. In: Gregg, Lee W./ Steinberg, Erwin R. (eds.): Cognitive Processes in Writing. Hillsdale NJ: Erlbaum, 31-50

Flower, Linda S./ Hayes, John R. (1981): Plans that Guide the Composing Process. In: Frederiksen, Carl H./ Dominic, Joseph F. (eds.): Writing. The Nature, Development, and Teaching of Written Communication. Vol. II: Writing. Process, Development, and Communication. Hillsdale NJ: Erlbaum, 39-58

van der Geest, Thea (1993): Hypertekst: schrijven en lezen in een niet-lineair medium [Hypertext: writing and reading in a not-linear medium]. In: Tijdschrift voor Taalbeheersing 15, 55- 67

Haas, Christina (1990): Composing in Technological Context. A Study of Note-Making. In: Written Communication 4 (7), 512-547

Shneiderman, B. (1989): Reflections on Authoring, Editing, and Managing Hypertext. In: Barrett, Edward (ed.): The Society of Text. Hypertext, Hypermedia, and the Social Construction of Information. Cambridge MA, London: MIT Press