CHAPTER 4

USER INTERFACE ISSUES

1. Introduction

The promise of hypertext lies in its ability to produce complex, richly connected and cross–referenced bodies of information. However, it can also become a complex system of tangled webs, confusing both authors and readers. According to Conklin, disorientation and cognitive overhead are the two most challenging problems related to hypertext. He feels that these two problems "may ultimately limit the usefulness of hypertext." [Conklin, 1987].

1.1 Disorientation

The problem of disorientation or "getting lost in space" arises from the need to know where one is in the network, where one came from, and how to get to another place in the network. In traditional text, it is not easy to get lost. There is the table of contents of topics with page numbers, the index with keywords and page numbers, and also bookmarks. However, in a complex hypertext network, with thousands of nodes and links, it is more than likely that the reader will get lost.

1.2 Cognitive Overhead

Cognitive overhead is the additional mental overhead on authors to create name, and keep track of nodes and links. For readers, it is the overhead due to making decisions as to which links to follow and which to abandon, given a large number of choices. The process of pausing (either to jot down required information or to decide which way to go) can be very distracting. It can become a serious problem if there are a large number of nodes and links.

All hypertext systems provide the basic capability of following a uni–directional link to a target node. However, the true potential of hypertext cannot be realized by this approach alone. Considerable amount of research efforts are underway in universities and the computer industry to develop better tools and methods to exploit the full potential of hypertext and also to solve or minimize the problems of disorientation and cognitive overhead.

2. Designs for Navigation

2.1 Graphical Browsers

Graphical browsers serve as overview displays for large bodies of information, especially in a hypertext system. NoteCards from XEROX and gIBIS from MCC both provide these browsers with which users can scroll through the entire network as well as rearrange the nodes. Both systems provide facilities to view the contents of the browser at different levels of detail. For large networks, the highest level of detail about the structure will be provided. The user can zoom in to see any portion of the browser in detail but cannot look at the details of the entire network because of screen space limitations [Utting & Yankelovich, 1989].

Graphical browsers help reduce disorientation by providing a two–dimensional spatial display of the hypertext network. They also help minimize cognitive overhead by showing a small part of the network. They also provide an idea about the size of the network which help users estimate the number of nodes and links in the system. If the user needs an idea about the contents of a target node prior to visiting it, some kind of a "peek" mechanism can be provided. Though general browsing is a low cognitive load operation, it is inefficient for directed search tasks or fact retrieval. Also, a graphical browser itself can become a tangled web if the hypertext network is large and if the display has to be updated because of dynamic changes in the network.

2.2 Web Views

In Intermedia, a hypertext system developed at Brown University, a web is defined as a network of documents or portions of documents linked together. Initially, the designers provided users with Web Views (similar to graphical browsers) called global map, local map, and local tracking map. A global map displayed every document in a web and the links between them. However, a global map worked well only with small webs. The local map was useful in focusing on the document of interest and its neighboring documents. The local tracking map dynamically updated the focus as documents were opened and activated.

Studies showed that the global map and the local tracking map were not of much use to users. Disoriented users need a sense of context and location. The need was felt to display no only spatial information ("Where can I go from here ?") but also temporal information ("How did I get here ?") [Utting & Yankelovich,1989]. The Web Views were modified to provide both spatial and temporal context in a flexible and non−intrusive manner. Three navigation tools were developed: a path, a map, and a scope line. Most often users would like to know, in advance, the amount of material in the web. This would help them decide whether to continue reading a document or to return later. The scope line informs the user about the number of documents and links in the web. Paths and maps are discussed in other sections below.

2.3 Maps and Overview Diagrams

Maps serve to improve spatial context in a hypertext network. The Intermedia view of a map is a local tracking map that displays all the documents or nodes linked to the current document which is dynamically updated. This ensures that correct information is always available; it also allows link previewing. Selecting a link marker will highlight the corresponding link line allowing the user to get an idea of the target without actually following the link [Utting & Yankelovich, 1989].

Overview diagrams, both at the local and global levels, serve as excellent navigational aids. Global overview diagrams provide an overall picture and can also serve as anchors for local overview diagrams. Local overview diagrams provide a fine−grained picture of the local neighborhood of a node. Overview diagrams for large systems might become complex and might introduce navigational problems of their own [Nielsen, 1990a]. Nielsen feels that in order to reduce the propagation of links from the local diagrams to global diagrams, weights can be assigned to the links based on their relevance to the user; this will trim the edges of the graph at the global level. Also, anchors can be differentiated, graphically, based on their estimated relevance to the users. Statistical analysis of the most frequently traversed links by a number of users might provide valuable insights into developing system−generated overview diagrams.

2.4 Paths and Trails

In most current hypertext systems, readers have a problem trying to understand the material presented because they view it in the wrong order or they simply cannot comprehend easily. The concept of a path can help solve this problem by allowing authors to determine an appropriate order of presentation for a given audience. It will reduce both disorientation and cognitive overhead since users will follow a pre−defined path which will also narrow down their choices. The concept of paths or trails was first enunciated by Bush in his classic paper, "As We May Think". He called a trail a sequence of links through a memex.

"The process of tying two items together is the important thing.....Before him are the two items to be joined, projected onto adjacent viewing positions....The user taps a single key, and the items are permanently joined.....Thereafter, at any time, when one of these items is in view, the other can be instantly recalled merely by tapping a button....Moreover, when numerous items have thus been 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....trails do not fade....Several years later,.....tapping a few keys projects the head of the trail. A lever runs through it at will, stopping at intersecting items, going off on side excursions." [Bush, 1945].

The idea of trails was implemented as paths by Trigg in his TEXTNET system. In Intermedia, a path is a list of documents users visited earlier in a browsing session. The display of a path consists of the name of the document, an icon indicating the type of event (opening or activating documents), and a timestamp indicating when the event occurred. A user's path is saved when closing the web and restored when opening the web, the next time. Thus, a path can be used to collect all interesting documents to form a linear document that can be preserved in printed form [Utting & Yankelovich,1989].

The Scripted Documents System, developed at XEROX, uses paths which are procedural and programmable. The items in the path (nodes) can be "active" entries or scripts which can do computations, execute programs etc. The entries provide the content while the path provides sequencing. Paths can be created and edited using path editors. Readers can get local and global views of relevant paths. Playback mechanisms are supported which allow users to play back a path either single−stepped or automatic. Different kinds of scripts provide different paths and can be used to create presentations for different classes of audiences [Zellweger,1989].

2.5 Guided Tours and Tabletops

The concept of paths was extended to "guided tours" in NoteCards [Trigg,1988]. A guided tour is a system−controlled navigational tool that can be entered and exited at the user's will. Progress can be monitored using maps or overviews of the hypertext database. In NoteCards, a guided tour can be accessed through a graphical browser (which displays a portion of the hypertext network in terms of cards and links) by both authors and readers; the "stops" on the guided tour are sets of cards arranged on the screen according to a particular layout, also called the "tabletop". Thus, a guided tour is a graphical interface to a network or path of tabletop cards, connected by links. A tabletop is a snapshot of cards currently on display, including their positions, shapes, scrolled locations of their contents and any overlapping. The decision about what cards to include in the tabletop is left to the author. Tabletops, in addition to saving individual card layouts, can be used to make online presentations and hard−copy screen shots or transparencies for offline presentations. Editing tools are available for creating and managing both tabletops and guided tours.

A reader has necessary controls to start a guided tour, traverse a path of tabletops in sequence, jump arbitrarily to any tabletop, go back and forth between tabletops or reset the state of the guided tour. Facilities are also available to peek into tabletops in a path without actually "opening" them. Guided tours, I would add, are like "hypertext within hypertext". They help bridge the communication gap between hypertext authors and readers. They make the contents of a document intelligible to people who are not already familiar with it − things like a description of their contents, the rationale behind their contents, an explanation about the context of their use and a history of their construction. A demonstration can be carried out by the author, in absentia, using guided tours. Trigg refers to this advantage of remote pointing as "remote deictic reference".

Nielsen feels that even though a guided tour seems to limit the very purpose and potential of hypertext, it can be used to introduce the concept of hypertext to new readers and for small systems. Different levels of guided tours can be designed for different types of users [Nielsen, 1990b].

2.6 Backtracking, History Lists, Timestamps, and Footprints

Backtracking allows visiting previously visited nodes. Backtracking in a linear fashion and also the ability to arbitrarily jump to previous nodes helps people to bail out of difficult situations. The most preferred method is the path−following principle which allows traversing, in reverse order, those nodes that were previously visited since this approach relies on the user's memory of his or her own navigation behavior. The structure−oriented feedback approach allows users to directly jump to a node without backtracking.

However, experiments have shown that combining these two methods might lead to confusion [Nielsen, 1990a].

Backtracking has been provided in system such as Guide and HyperCard. However, most backtracking mechanisms do not support the feature of going forward to the node from which the backtracking was initiated. Also, backtracking can confuse users if the interface is not consistent. In a document designed with Guide when readers returned from a backtracking operation the display looked different making them wonder whether that was the node they were viewing before taking off on the backtrack [Nielsen, 1990b]. Hence, Nielsen suggests that backtracking mechanisms must fulfill two requirements − "It should always be available, and it should always be activated in the same way."

A textual history list mechanism was developed for NoteCards which maintains an ordered list of each notecard that was examined in a particular session. Users can select an item from the list and look through a browser. Nodes that have already been visited in a session are marked with a plus sign, one for each time visited. Visual indicators such as the plus sign or checkmarks or asterisks serve as "footprints" on overview diagrams and help users to avoid returning to nodes that have been recently visited. Some implementations of the history list allow users to customize the list for future interaction. A variation of the history list called the history tree shows the users "how" they traversed a set of linked nodes, the digressions, and multiple visits to nodes. Both the history list and history tree can be saved and annotated with text and graphics [Utting & Yankelovich,1989].

HyperCard has a graphical history list called the recent list which has miniature snap−shots of the last forty−two nodes visited. Clicking on a miniature brings that card to the display. Of course, this method makes the assumption that an individually may not be able to remember the name of the node but may remember the "look" of the node. Some usability studies have shown that it is very difficult for users to distinguish these miniatures from one another [Nielsen, 1990a].

Nodes that are visited can be timestamped (along with the accumulated time spent at each node) and maintained in a chronological order [Nielsen, 1990a]. This feature will help users, at a later date, to recognize whether the node was visited before and if so, the duration allowing users to change their viewing behavior accordingly. The Document Examiner supports a history of commands executed and a history of documents examined. The command history allows people to see previously executed commands and the document history is in the form of bookmarks for each document that was previously opened.

2.7 Arbitrary Jumps, Landmarks, and Bookmarks

Arbitrary jumps or gotos can be provided enabling users to go to any node in the system. This can be accomplished by zooming in and out using an animated iris that opens for anchored jumps and closes for return jumps. "Landmarks" or prominent nodes can be provided which can always be accessed from anywhere in the system. The concept of a bookmark is similar to a history list except for the fact that a bookmark is placed by the reader only if a particular node will be of interest at a later date. Hence, a bookmark list is smaller more manageable and more relevant to the user.

Experiments have shown that random jumps from anchors leading to multiple destinations can be very confusing to users [Nielsen,1990a]. This can be avoided by listing the possible destinations when an anchor is activated allowing the user to choose a predictable path. In Intermedia, this problem has been solved by displaying a single link icon (instead of multiple links) which can be quickly queried to show the specific links, their names, and their destination nodes.

2.8 Embedded Menus

Embedded menus, as opposed to explicit menus, allow the user to select a word or item embedded within the text of a node and can be selected using a touch screen, cursor keys or mouse pointer. In The

Interactive Electronic Encyclopedia System (TIES), selectable items are highlighted directly in the text, a method now called touchtext [Koved & Shneiderman, 1986]. The embedded menus in TIES (now called HyperTIES) can be traversed using cursor keys and can be selected either by clicking or by a keystroke; a new node or piece of information will be retrieved. At every node in the network, the user can request a return to a previous article using other navigation mechanisms. An extension of the embedded menus approach can be found in MIT's Spatial Data Management System (SDMS) where users can select a graphical area (such as a territory in a map) and retrieve a more detailed map from the database. The user can undo the effects of the selection process by returning to less detailed maps.

Embedded menus are a better way of indexing for hypertext systems since they emphasize the understanding of concepts. They highlight semantic relationships rather than physical relationships. They provide meaningful task domain terms (as opposed to computer domain terms) and concepts, thereby reducing disorientation [Marchionini & Shneiderman, 1988]. Embedded menus reduce the "loss of context" feeling by being part of the information being displayed. They provide information hiding (layers below are shown only when requested). The extent of "embeddedness" can be varied depending on the skill level of the user. They are suitable for learning−by−browsing systems as in museums. Further research is required regarding the negative aspects of using highlighted embedded menus. It is possible that they may cause disruption, reducing speed and comprehension.

2.9 Fisheye Views and Spiders

Furnas implemented a fisheye view algorithm which is similar to looking at a scene with a wide angle lens − things of greater interest will be at the center, while items of lesser interest will be on the periphery. This algorithm generates an image of the neighborhood by computing a relationship between a priori importance of a node and the distance between that node and the current position in the hypertext network [Carlson, 1989].

Thoth−II treats hypertext as a directed graph with semantics [Collier, 1987]. Nodes themselves do not contain text. Pieces of text are connected to the nodes by text links while the nodes are connected to each other by labeled value links; text can contain embedded links, called lexical links, to other nodes. "Spiders" is a directed graph browser in Thoth−II where a global map is created dynamically as a user browses through linked nodes. As the user interacts with the structure that is being viewed, new graphic objects (nodes and links) are created. Activating a node expands it showing links to other nodes which themselves fan out to other nodes. Thus, the network expands or fans out in two−dimensional space creating "spiders" on the display. The window on which this structure is displayed can be moved around to view other parts of it falling outside the viewing area. The browsing mode allows the user to browse through the graph and manipulate nodes and links whereas the text mode allows the user to view the textual pieces attached to the nodes. The disadvantage with this system is that the whole screen can get filled up with spiders very quickly as the user clicks on various nodes.

2.10 Roam and Zoom Techniques

Hypertext navigation is also restricted by the physical limitations of the display screen. The inability to view large amounts of information at any one time due to the small size of computer displays falls under the category of context−in−the−small problems [Nielsen,1990a]. This includes the issue of readability, when the contents of a node do not fit into one screen and have to be carried over to other screens. Larger displays can only partially solve this problem. Conventional scrolling techniques use arrow keys or paging keys allowing only vertical movement to various places on the screen. While scroll bars do allow two−dimensional navigation, it is not easy to focus on a particular region of interest.

In an effort to improve the display of large two−dimensional spaces, researchers at the University of North Carolina, Chapel Hill, developed two similar direct manipulation techniques. Both use a miniature of the entire information space to assist the user in remembering his or her location. The first technique allows the user to rapidly "roam" over the space while the second allows "zooming" into a particular region while roaming [Beard & Walker, 1987].

The entire information space can be shown, in miniature form, in a map window occupying a small part of the display. A wire−frame box or rectangle inside the window shows the portion of the information space displayed on the main display. The main display is the actual viewport into the information space. Thus, the map window provides a clear sense of location within the information space. the size of the wire−frame can be changed using the mouse thereby zooming in and out of the region. Also, the wire−frame box can be dragged around the map window thus roaming around the information space. Experiments proved that the roam and zoom features were significantly faster than vertical and horizontal scroll bars thus improving performance.

2.11 Conceptual Space Navigation

Some researchers feel that learning systems should have some amount of disorientation and cognitive overhead in order to facilitate exploration and learning [Mayes et al., 1990]. They feel that while most researchers are concentrating on navigation through information space, very little work has been done on navigation through "conceptual space". They contend that simply following links to nodes does not necessarily provide effective learning − "they do not tell us ABOUT anything, but only WHERE it is."

In learning systems, disorientation in conceptual space is required sometimes in order to explore and learn. Thus, users need to be guided not only by system information but also by discovery. The issue of cognitive overhead has not been found significant in learning systems [Mayes et al., 1990]. The authors say, the very question "what to do next, will enrich the process of learning rather than detract from it." A learning−by−browsing system called StrathTutor was developed where the links were computed based on how the nodes were related to each other, conceptually, in terms of certain attributes. This not only allows navigation through conceptual space but also the ability to "interrogate" the system by designating a combination of attributes that are meaningful at a particular instant of exploration; the system will respond by giving the learner a guided tour of all nodes satisfying those attributes.

StrathTutor also provides another facility called the "quiz" where the learner is asked to play a game in which he or she is asked to identify the nodes that have common attributes. This helps the learner create his or her own personal view of the underlying conceptual space. The content search and structure search mechanisms, suggested by Halasz, will be especially useful in learning systems.

3. Usability and Evaluation of Hypertext

Wright feels that the following five issues have to be examined while evaluating hypertexts [Wright, 1991]:

1. Adequacy of the content and the interface.

2. Acceptability to readers.

3. Adaptability by readers for the task in hand.

4. Skills of the readers as information users.

5. Costs of production and dissemination.

3.1. Adequacy of the content and the interface

Hypertext design involves a mixture of decisions relating to the content, functionality, display, and control. The usefulness of hypertext depends on its purpose, ease of navigation, and the population domain. The authors have to determine tradeoffs among various design principles. For example, for a learning system the authors would have to not only look into the outcomes of using hypertext but also the quality of learning. In short, there is no upper limit on adequacy.

3.2. Acceptability to readers

There is also the issue of adequacy once the users have learnt to accept a hypertext system. For people seeking more information (say, after an initial exposure to a tourist information kiosk), the restrictive design of the application may pose a limitation (say, the tourist would like to know, immediately, about some good restaurants near a place of historic interest) – in fact, the display may not be large enough to display more details without chunking information. An overview mechanism (for browsers) and a query mechanism (for information seekers) may be more appropriate in such a situation.

3.3. Adaptability by readers for the task in hand

Hypertext systems have to be appropriate to the tasks that the users are trying to accomplish. There should be facilities for annotation and placing bookmarks. Hypertext interchange (the compatibility of file formats) is an important criterion of adaptability and hence acceptability. With the increasing use of hypertexts for collaboration, their evaluation should include the assessment of benefits to the group as a whole as well as the individuals.

3.4. Skills of the readers as information users

Hypertext evaluation should include an evaluation of the demand a system may make on the reading skills of the users. This could mean the evaluation of users reading linear text (on print). The system should extend the range of cognitive activities that readers will engage in (that is, provide an insight into strategies which they may not have thought of). The following could be some cognitive extensions :

Ability to search using keywords (general IR).

Changing the very nature of the work being done by creating some sort of an interdependency between the system and the users.

Evolution of new ways of thinking with the continued use of hypertext.

3.5. Costs of production and dissemination

The cost effectiveness of having information in hypertext form should be considered. Conventional IR systems allow only retrieval while hypertext systems help users integrate units of information retrieved.

Evaluation should also include looking for the availability of automated tools or cognitive prostheses to overcome cognitive overhead and disorientation. Readers adapt different navigation strategies based on the complexity of the task at hand. Such tools might help with a variety of subtasks related to hypertext usage. These subtasks might include specifying search targets and planning the order in which information will be sought. Tools are also required to store and manipulate the information found in order to integrate with other work. However, the effectiveness of such tools might be reduced due to the following :

Readers may not realize they need help.

Readers may not know how to use such tools.

Readers might blindly follow certain procedures without understanding their need.

Hence, the design and evaluation of hypertext systems should include the understanding of how people use their cognitive resources to handle information.

Nielsen also has discussed four categories for the evaluation of hypertext documents [Nielsen, 1991]. These are:

a. Utility

This is a measure of whether the hypertext document actually helps a user perform the intended task. This has to be compared with performing the same tasks with linear text. Many empirical tests have shown that readers exhibit poorer performance with hypertext than with paper documents.

b. Integrity

This is a measure of the completeness of the document – whether it is up to date and not misleading and if it is easy to maintain.

c. Usability

Usability can be measured as a combination of these factors:

Ease of learning – It is a measure of how fast a reader can start learning and navigating through a hypertext document. Good presentation structure and graphic design are key determinants.

Ease of use – It is a measure of how fast a reader can locate information. Appropriately defined links, search mechanisms, backtracks, landmarks and other navigational aids can greatly increase efficiency of use.

Error handling – It is a measure of how many errors a reader makes and how easy it is for them to recover from such errors.

d. Aesthetics

This is a measure of how pleasing is the system to the user.

4. Summary

In a true hypertext system, users must be able to move freely through the system according to their needs, without getting lost either spatially or cognitively. The facilities to navigate through a hypertext database must be at least as rich as those available in books. Some of the designs currently available to navigate through hypertext were reviewed in this chapter. Usability issues and evaluation criteria were discussed. When the initial excitement about hypertext dies down and systems become more common, better navigation techniques and more systematic evaluation measures will emerge both from developers and users. This will be based on the organizational setting, the targeted task domain, the typical user population, and the desired outcomes of navigation.

References

[Beard & Walker, 1987]. Beard, David & Walker, John. Navigational Techniques to Improve the Display of Large Two–Dimensional Spaces, University of North Carolina at Chapel Hill Technical Report TR87–031, November 1987.

[Bush, 1945]. Bush, Vannevar. As We May Think, The Atlantic Monthly, July 1945.

[Carlson, 1989]. Carlson, Patricia Ann. Hypertext and Intelligent Interfaces for Text Retrieval, The Society of Text, MIT Press, 1989.

[Collier, 1987]. Collier, George H. Thoth−II : Hypertext with Explicit Semantics, Proceedings of Hypertext '87 Conference, November 1987.

[Conklin, 1987]. Conklin, Jeff. Hypertext : An Introduction and Survey, IEEE Computer, September 1987.

[Halasz, 1988]. Halasz, Frank. Reflections on NoteCards : Seven Issues for the Next Generation of Hypermedia Systems, Communications of the ACM, July 1988.

[Koved & Shneiderman, 1986]. Koved, Larry & Shneiderman, Ben. Embedded Menus : Selecting Items in Context, Communications of the ACM, April 1986.

[Marchionini & Shneiderman, 1988]. Marchionini, Gary & Shneiderman, Ben. Finding Facts vs. Browsing Knowledge in Hypertext Systems, IEEE Computer, January 1988.

[Mayes et al., 1990]. Mayes, Terry, Kibby, Mike & Anderson, Tony. Learning about Learning from Hypertext, Designing Hypermedia for Learning, NATO ASI Series, Volume F67, Springer Verlag, 1990.

[Nielsen, 1990a]. Nielsen, Jakob. The Art of Navigating through Hypertext, Communications of the ACM, March 1990.

[Nielsen, 1990b]. Nielsen, Jakob. Navigating through Large Information Spaces, Hypertext and Hypermedia, Academic Press, 1990.

[Nielsen, 1991]. Nielsen, Jakob. Panel Discussion − The Nielsen Ratings: Hypertext Reviews, Proceedings of Hypertext '91, 1991.

[Trigg, 1988]. Trigg, Randall. Guided Tours and Tabletops : Tools for Communicating in a Hypertext Environment, ACM Transactions on Office Information Systems, October 1988.

[Utting & Yankelovich, 1989]. Utting, Kenneth & Yankelovich, Nicole. Context and Orientation in Hypermedia Networks, ACM Transactions on Information Systems, January 1989.

[Wright, 1991]. Wright, Patricia. Cognitive Overheads and Prosthesis: Some Issues in Evaluating Hypertexts, Proceedings of Hypertext '91, 1991.

[Zellweger, 1989]. Zellweger, Polle T. Scripted Documents : A Hypermedia Path Mechanism, Proceedings of Hypertext '89 Conference, November 1989.