2

Co

USE OF HYPERTEXT FOR THE DEVELOPMENT OF AN OFFICE REFERENCE SYSTEM ON ECONOMIC ANALYSIS

THESIS

DeAnna L. McMurry
Captain, USAF

AFIT/GCA/LSQ/90S-9

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

9

2

AFIT/GCA/LSQ/90S-9

USE OF HYPERTEXT FOR THE DEVELOPMENT OF AN OFFICE REFERENCE SYSTEM ON ECONOMIC ANALYSIS

THESIS

DeAnna L. McMurry
Captain, USAF

AFIT/GCA/LSQ/90S-9

The opinions and conclusions in this paper are those of the author and are not intended to represent the official position of the DOD, USAF, or any other government agency.

AFIT/GCA/LSQ/90S-9

USE OF HYPERTEXT FOR THE DEVELOPMENT OF AN OFFICE REFERENCE SYSTEM ON ECONOMIC ANALYSIS

THESIS

Presented to the Faculty of the School of Systems and Logistics

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Cost Analysis

DeAnna L. McMurry, B.S.

Captain, USAF

September 1990

Acknowledgements

In completing a major effort like this thesis, several individuals deserve recognition. First, I would like to thank my advisor, Lieutenant Colonel Richard E. Peschke for his guidance, motivation, and good feedback throughout the research development. He helped me go from a hypertext tenderfoot to somewhat of an expert. I would also like to thank two members of the GCA-90S class for their continual concern and encouragement throughout the masters program. Bryan and Dimitri, thanks for the memories!

Most of the credit for this thesis, however, must go to my parents, Floyd and Jean Weelborg of Dell Rapids, South Dakota. Their strength and support started me on the road to success. Thanks.

Finally, and most importantly, I dedicate this thesis to my new husband, Robert Dean McMurry, Jr. His love, patience, and understanding from across the miles rescued me many times during this research and masters program.

Captain DeAnna L. McMurray

Table of Contents

Page

Page

List of Figures

List of Tables

Abstract

This objective of this ~~research~~ thesis was to evaluate the use of hypertext technology for an automated office reference system. A secondary purpose, because of the relative newness of hypertext technology, was to provide a single reference document which describes hypertext and the steps needed in managing or developing a hypertext product. Individuals unfamiliar with hypertext technology can learn how to apply it in an office environment.

The evaluation consisted of two phases, a literature review to discover the hypertext supports for a reference system and a prototype development and test of an Economic Analysis reference system. The literature review includes a review of hypertext features, advantages and disadvantages, applications (including government use), factors affecting usability, and legal concerns. The prototype development answered questions in three areas: economic analysis information required, hypertext development concerns, and applicability of hypertext to the office environment.

The developed prototype, tested using a questionnaire answered by 60 graduate students, showed that hypertext has strong applicability for an office reference system. The results showed the reference system was easy to learn and use, would provide quicker and easier access to information, and users could learn more than through the current paper reference system.

USE OF HYPERTEXT FOR THE DEVELOPMENT OF AN OFFICE REFERENCE SYSTEM ON ECONOMIC ANALYSIS

I. Introduction

This chapter provides an introduction to the research. First, problems with the current reference system for economic analysis are discussed. Second, hypertext is described as it will be used for development of an automated reference system on economic analysis. Next, the specific investigative questions which this research answers are presented. Then the scope and limitations of this research are characterized. Finally, the thesis organization is outlined.

General Issue

Today's office reference system for economic analysis within the Department of Defense is manual--consisting of separate paper files, regulations, guidebooks, policy letters, correspondence, training course textbooks, and other paper documents. The reference system is used by individuals with varying knowledge levels, from those first learning economic analysis techniques to those needing to research the answer to specific questions.

The Base Level Cost Analysis Handbook, developed for the Air Force in 1988, was designed to become a concise reference system for base-level cost analysis tasks. As such, the handbook consists of two books and over 1,000 pages. Two chapters deal specifically with economic analysis, Chapters 8 and 11. These chapters combined are nearly 200 pages long and use many cross references to other chapters of the handbook (26). Therefore to access the needed information on economic analysis, both volumes are required. Although the handbook references over 30 regulations and guides specifically pertaining to economic analysis (26:11-8 to 11-10), none are contained in it. The user must access needed regulations through a different source, either within her office or through a master publications library. Each base level cost analyst is provided a copy of the handbook. Economic analysis information similar to the above handbook is also presented in Volume I of the AFSC cost estimating handbook series, AFSC Cost Estimating Handbook (25).

Within most major commands, the base level cost office within the comptroller organization is the office of primary responsibility for economic analysis. The cost office is responsible for both conducting the economic analysis and identifying offices of collateral responsibility to help formulate alternatives, make assumptions, and provide needed data (23:5). With recent budget cutbacks however, a proposal is under consideration to eliminate or greatly reduce base level cost analyst positions, thus disbanding the current pool of experience for conducting economic analyses.

Within other major commands, such as the Air Force Communication Command, there are no base level cost analysts; responsibility for accomplishing economic analyses falls to individuals within the functional organizations. These individuals typically have no background training or knowledge of the techniques required to conduct economic analyses. They also do not have immediate access to the handbooks referenced above or any of the needed regulations.

Many Department of Defense agencies provide training courses on the techniques used in conducting economic analyses: Sheppard Technical Training Center (STTC), the Air Force Institute of Technology (AFIT), U.S. Army Logistics Management Center (ALMC), Army Management Engineering Training Activity (AMETA), and Defense Systems Management College (DSMC). Course prerequisites often limit the individuals who can attend the training to those working in an economic or cost analyst position, effectively removing the functional individual from consideration for the courses (24:1). Limited quotas are available for many of the courses taught by the above agencies, meaning that individuals responsible for conducting economic analyses may not be able to attend the courses when needed. Often, one individual from each office attends the courses and is expected to train the remainder of her office. If the individual changes jobs before the in-office training can be accomplished, the office no longer has access to the needed knowledge.

The above conditions combined make it difficult to maintain adequate reservoirs, in an office that is required to conduct economic analyses, both of individuals trained in economic analysis techniques and the needed economic analysis information. This can cause delayed proficiency gain for inexperienced workers, lower quality economic analyses, and lost continuity with job changes.

The need for a more concise and more easily delivered package on economic analysis is indicated by the efforts of the Air Force Cost Center; they are working on development of microcomputer-based training modules on economic analysis. At this time, they have developed two computer based training (CBT) modules: "How to Document an Economic Analysis" and "Time Value of Money". Another module due to be released soon is on economic analysis and combines the documentation CBT with economic analysis overview information (22). Appendix A provides a description of each of the CBT modules.

This research examines another method of improving the internal office reference system through automation--hypertext.

Specific Problem

One way to tie separate reference documents together into one concise package is with hypertext. Hypertext is a type of computer database system that supports non-sequentially linked pieces of text. "Rather than turning pages to find more information or using an index to find a topic, it [hypertext] is structured so that wherever more information exists it is linked off a key word or phrase" (61:1). Hypertext provides for machine supported links to cross reference or continuing information.

The setup of a hypertext application works in several steps. First, the existing reference documents are input into the computer database in linear form. Next, these documents are broken into blocks of text that typically express a single idea. Finally, the blocks are linked together in two ways--linear (as originally input) and nonlinear. The nonlinear links allow reference blocks from the same or separate documents to be linked together, providing paths through the database on specific subject areas (e.g. inflation factors). The user of a hypertext system accesses text blocks through a series of windows, following a network of linked information.

This research has a twofold purpose. First, it develops a prototype hypertext economic analysis reference system. Hypertext has already been used on projects of similar scope within other organizations as discussed in the literature review. Secondly, because of the relative newness of hypertext technology this research presents a single reference which describes hypertext and the steps needed in managing or developing a hypertext product.

Research Questions

Three general areas of research questions are answered. The first deals with the information needed for an economic analysis reference system. The questions are as follows:

1. What economic analysis documents are necessary to give a good test of a prototype system?

2. Are these documents available in computerized form, and if not, how can they be computerized?

The second area deals with the hypertext system development. The questions answered here are:

1. What hypertext systems are available or obtainable for educational research?

2. What computer system is more appropriate (mainframe, micro, or mini)?

3. What methodology should be used to break documents into text blocks?

4. What nonlinear links are needed?

The third area deals with the use of the system in the office environment. The specific research questions are as follows:

1. Is a hypertext system compatible with office reference, specifically, does linked text provide easy use in finding and learning information?

2. What capabilities are available to update the hypertext references as new regulations or references become available?

Scope

Due to the substantial number of references available on economic analysis, this prototype reference system deals with a sample that demonstrates the hypertext capability in this area. Because of the many references already available, the prototype uses existing documents; no new text is created. Those types of economic analyses that are typically performed in the operational environment provide the basis for the selection of information. Because of the limited depth of the prototype, a complete analysis of user requirements was not accomplished.

Because the prototype was used to demonstrate the concept of using hypertext for an economic analysis office reference system, prototype testing was not as extensive as for an actual production system. Prototype development usually consists of iterative development and tests of a product. This research was limited to iterative development and one overall prototype test.

Research Development

The research on this topic follows the pattern outlined below.

Chapter I introduces the research. It discusses the current problem with the economic analysis reference system, a possible solution in hypertext, specific research questions, and the scope of the research.

Chapter II contains a review of the literature applicable to the use of hypertext as a reference system.

Chapter III discusses the methodology to be used in developing the prototype reference system.

Chapter IV provides the findings and analysis in development and testing of the prototype.

Chapter V presents conclusions reached through the research and recommends areas for follow-on research.

II. Background

This chapter presents a review of the literature relevant to the development of a hypertext office reference system. First, a brief history of hypertext is reviewed. Second, an overview of the key features of a hypertext system is presented. A discussion of the advantages and disadvantages of using hypertext for accessing large amounts of information follows. Next, the browsing feature is examined in more detail. Two classification schemes are then provided to give the reader a better understanding of the various ways hypertext systems can be used and how reference systems are classified. Several specific applications similar to this thesis project are discussed. Government use of hypertext applications is briefly examined.

Hypertext, from the user's perspective, is discussed in terms of factors effecting usability of hypertext systems. Legal implications of a hypertext document are then discussed. Finally, the chapter concludes with a preview of the future for hypertext.

History of Hypertext

Many people view hypertext as being a recent concept. This is not true; however, it is only recently that the computer technology has been available to support the concept. Hypertext was first described in 1945 by Vannevar Bush, President Roosevelt's Science Advisor in an article for the Atlantic Monthly called "As We May Think" (14). Bush proposed a "memex" machine based upon microfilm and photocells and described the machine's essential feature as the ability to tie two items together.

Although the memex was never developed, Bush is given credit for the original thought of machine supported linking of information (18:10).

Additional work in the area was not performed until nearly 20 years later when Douglas Engelbart and Ted Nelson began work on H-LAM/T (Human using Language, Artifacts, and Methodology, in which he is Trained) and Xanadu respectively. Guide, the first hypertext system for personal computers was released in 1985 (74:88-89). A detailed history of hypertext systems is presented in A Survey of Hypertext (18:7-37) and Hypertext Hands-On! (74:75-94).

Standard Hypertext Features

Manual Hypertext Systems. Although the computer technology supporting hypertext is new, manual equivalent systems have existed for a long time.

> One example is the encyclopedia. As you read an article in an encyclopedia, you find references to related items which will have more information. These references are usually "non-linear"; that is, the other items or articles do not generally directly follow the one you are looking at. You may have to open another volume in the set. You may also be referred to a diagram or picture elsewhere in the encyclopedia. (42)

An encyclopedia set can be considered one large document with many interrelated articles. If the encyclopedia were automated with hypertext it would be a hyperdocument, with all inter-article and intra-article relationships specifically computer supported with links.

Building Blocks. The key feature of a hypertext system consists of machine supported links for connecting information within a nonlinear database (18:3; 31:237; 21:169; 59:27). In addition to this key ingredient, two other essential features characterize hypertext systems.

First, information is broken into "chunks" or nodes that typically express a single idea or concept. While a hypertext system is limited to text information, a hypermedia system may include other elements such as graphics, video, audio, still pictures, and animation.

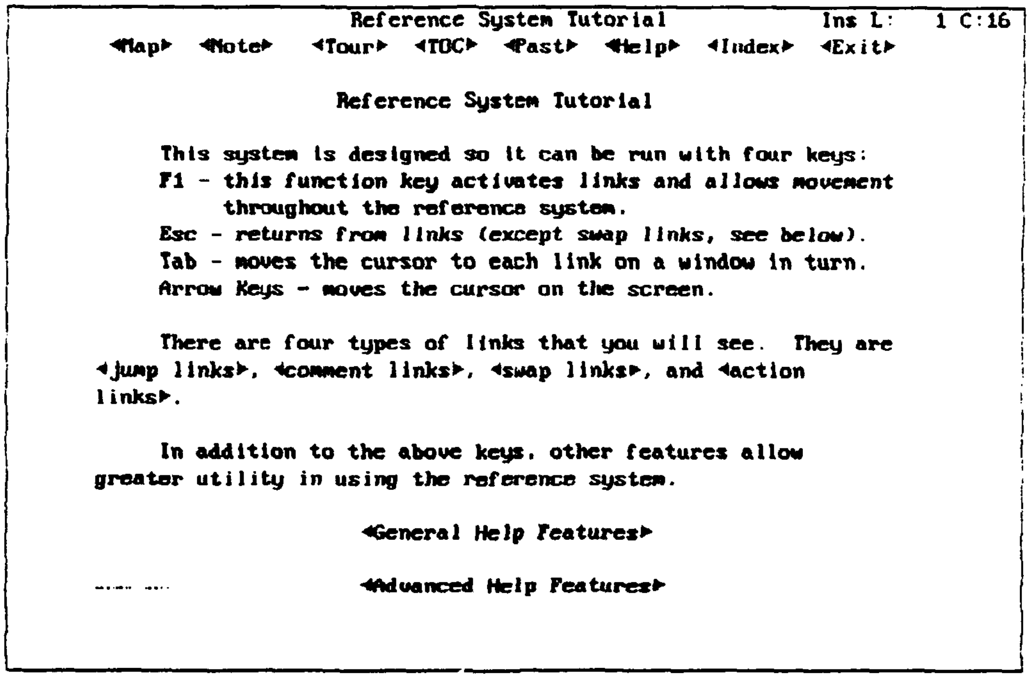
Second, users browse through information within a hypermedia database by following links from node to node. The entire system forms a hyperdocument network in which all the nodes are connected by links (18:6; 76:33; 1:820; 59:27).

Figure 1 displays a window with links and the action that occurs when one link is activated.

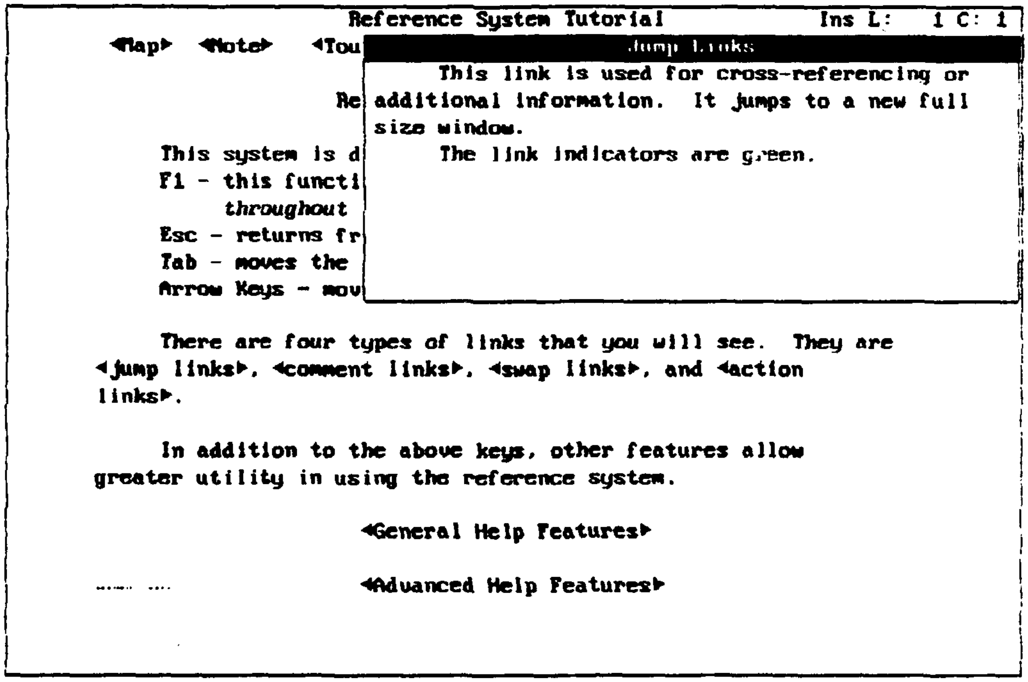
The next portion of this section further explains these building blocks and provides additional information on the average current hypertext system.

Links. "Links are the labels that connect one node (document, article, topic) with another" (74:3). They must have the ability to rapidly transport the user to new places within the hyperdocument, as the decision to follow a link "is associated with a cost (e.g., time, distraction, annoyance) and a reward (e.g., useful information)" (34:882). Akscyn feels that the ability to rapidly traverse a link is the "most important parameter of a hypermedia system" (1:829). Conklin states that a delay of one or two seconds is unacceptable (18:39) and Akscyn's design goal is a response time of less than 0.25 seconds (1:829).

Links have two ends, the link source (reference) and the link destination (referent) (18:41; 31:239). The method of indicating source and destination within the hyperdocument varies by system. Among the



```
                     Reference System Tutorial               Ins L:   1 C:16
      ◄Map►  ◄Note►   ◄Tour►  ◄TOC►  ◄Past►  ◄Help►  ◄Index►  ◄Exit►

                         Reference System Tutorial

         This system is designed so it can be run with four keys:
         F1 - this function key activates links and allows movement
              throughout the reference system.
         Esc - returns from links (except swap links, see below).
         Tab - moves the cursor to each link on a window in turn.
         Arrow Keys - moves the cursor on the screen.

         There are four types of links that you will see.  They are
     ◄jump links►, ◄comment links►, ◄swap links►, and ◄action
     links►.

         In addition to the above keys, other features allow
     greater utility in using the reference system.

                          ◄General Help Features►

                          ◄Advanced Help Features►
```



```
                     Reference System Tutorial               Ins L:   1 C: 1
      ◄Map►  ◄Note►   ◄Tou|                   Jump Links
                          |        This link is used for cross-referencing or
                        Re|additional information.  It jumps to a new full
                          |size window.
         This system is d|        The link indicators are green.
         F1 - this functi|
              throughout |
         Esc - returns fr|
         Tab - moves the |
         Arrow Keys - mov|

         There are four types of links that you will see.  They are
     ◄jump links►, ◄comment links►, ◄swap links►, and ◄action
     links►.

         In addition to the above keys, other features allow
     greater utility in using the reference system.

                          ◄General Help Features►

                          ◄Advanced Help Features►
```

Figure 1. Hypertext Window With and Without Link Activation

ways used to indicate links are: boxed text (39:837), icons (76:35), reverse video, fonts, color (18:41), and whole text (1:829). Regardless of the indication method, only one or two keystrokes or taps of a mouse button should be needed to activate the link (31:239; 18:39; 39:837).

Links can be used for a variety of purposes. They can:

- transfer to a new topic

- show a reference (or go from a reference to the article)

- provide ancillary information, such as a footnote, definition, or annotation

- display an illustration, schematic, photograph, or video sequence

- display an index

- run another program (e.g., a spreadsheet or animation) (74:4)

Some systems allow creation of link types to indicate link purpose, with the number of typed links varying greatly by system, from over eighty (18:15) to two (1:828). These types are used to define the specific relationships between information being linked, e.g., citation, background, solution, correction, methodology, rewrite, argument (18:14), referential, and organizational (parent-child) (18:43). Other properties of links in hypertext documents include: creation, deletion, editing, naming, listing, and directing (18:40; 31:239).

Nodes. Nodes and the links that connect them form a nonlinear network. Nodes also have varying characteristics depending upon the hypertext system. While a node typically expresses one idea, a system's node length varies from screen size (1:827) to whole books (18:48). The variance in node size is based on how the information is presented; article-based or card-based. Article-based systems present entire documents in one node, allowing the user to scroll through the text onscreen. A disadvantage here would be loss of information if the user did not know to scroll down for more information. Card-based systems limit node size to the size of the screen and are nonscrolling. Card-based systems eliminate the problem of scrolling, but information needs to be broken into screen size chunks, thus increasing the conversion time from linear text to hyperdocument (61:5-6).

Regardless of the type of system, node units should be such that:

> a) an individual idea can be referenced elsewhere . . . and
>
> b) alternative successors of a unit can be offered to a reader (e.g., more detail, an example, bibliographic references, or the logical successor). (18:48)

Small nodes allow greater organizational possibilities while large nodes minimize the time and complexity of traversing between nodes (18:49).

Nodes can contain varying information depending upon the system. In a hypertext system, node content is limited to text. In a hypermedia system content is limited only by what the current technology can provide (76:33). Some systems allow nodes to be viewed as composites where related subnodes can be viewed together or individually (18:52; 31:239). Nodes can also be typed, similar to the typing of links, to specify the information contained (18:50).

Browsing. When hypertext was originally thought of by Bush in 1945 it was on the principle of browsing.

> The human mind . . . operates by association. With one item in its grasp, it snaps instantly to the next that is suggested by the association of thoughts, in accordance with some intricate web of trails carried by the cells of the brain. (14:106)

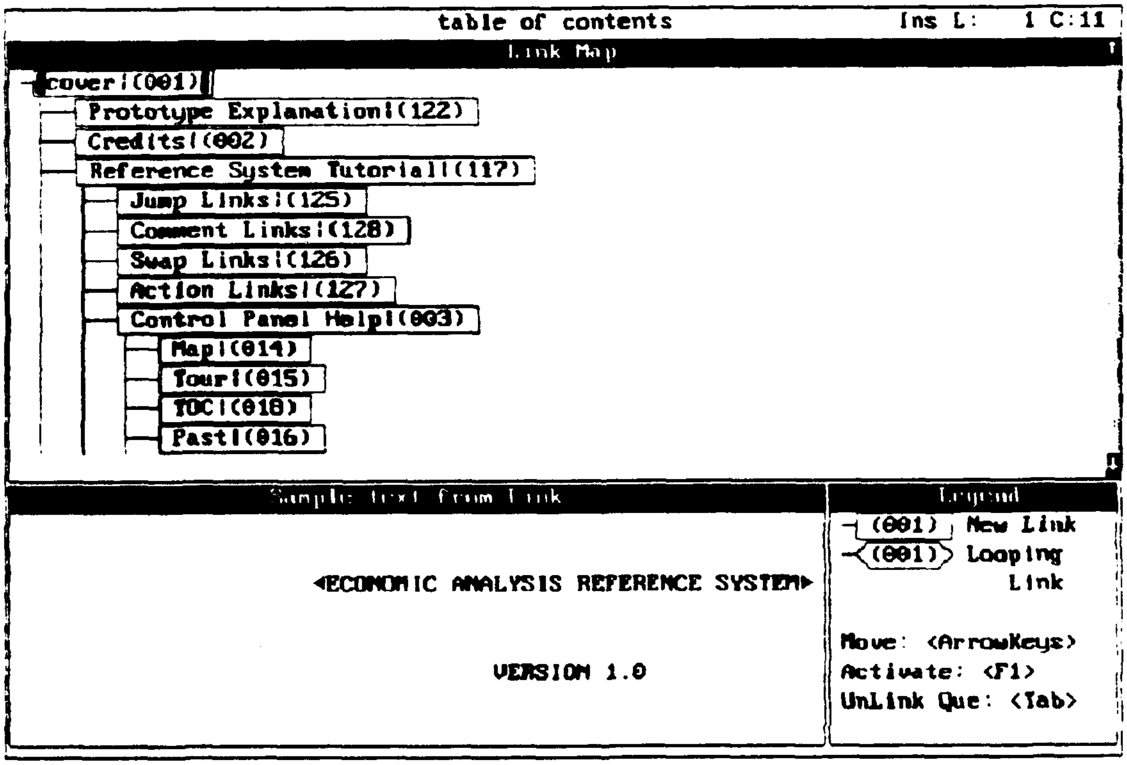
Browsing within a hyperdocument may take place in three ways:

a) by following links and opening windows successively, examining their contents,

b) by searching the network (or part of it) for some string, keyword, or attribute value, and

c) by navigating around the hyperdocument using a specialized graphical browser that displays the network (e.g. with nodes displayed as icons and links as lines between them. (18:7)

Figure 2 displays a graphical browser from the HyperWriter (TM) hypertext software. The nodes are identified both by window name and window number with lines indicating the link structure between them. At the bottom left of the map window the top part of the indicated node is displayed.



```
cover|(001)
    Prototype Explanation|(122)
    Credits|(002)
    Reference System Tutorial|(117)
        Jump Links|(125)
        Comment Links|(128)
        Swap Links|(126)
        Action Links|(127)
        Control Panel Help|(003)
            Map|(014)
            Tour|(015)
            TOC|(018)
            Past|(016)
```

<ECONOMIC ANALYSIS REFERENCE SYSTEM>

VERSION 1.0

```
(001) New Link
(001) Looping
      Link

Move: <ArrowKeys>
Activate: <F1>
UnLink Que: <Tab>
```

Figure 2. Graphical Browser (43)

Average Current System. As discussed above, the key features of hypertext (links, nodes, and browsing) vary between systems. The average current hypertext system, however, has the features shown in Table 1.

TABLE 1

Features of a Hypertext System (39:840)

Feature	Description
Nodes:	Typed (text, graphics . . .), implemented using a type hierarchy
Links:	Binary, bidirectional; labeled but not typed. Anchors can be whole nodes or points/regions within the node
Overview:	Browsers containing node/link diagrams of the network
Hierarchies:	Special support for hierarchical networks
User Interface:	Multiple windows; mouse/menu driven
Search/query:	Slow, full-text string match
Storage:	Standard files or relational DBMS [Data Base Management System]

As with any computer software technology, hypertext has certain advantages and disadvantages for use in specific applications. The next section considers advantages and disadvantages of using a current hypertext system for reading and structuring of information.

Hypertext Advantages and Disadvantages

Advantages. Conklin describes many advantages in the use of hypertext for accessing large amounts of information.

1. ease of tracing references: machine support for link tracing means that all references are equally easy to follow to their referent, or backwards to their reference;

2. ease of creating new references: users can grow their own

> networks, or simply annotate someone else's document with a comment (without changing the reference document);
>
> 3. information structuring: both hierarchical and non-hierarchical organizations can be imposed on unstructured information; even multiple hierarchies can organize the same material;
>
> 4. global views: browsers provide table-of-contents style views, supporting easier restructuring of large or complex documents; global and local views can be mixed effectively;
>
> 5. customized documents: text segments may be threaded together in many ways, allowing the same document to serve multiple functions;
>
> 6. modularity of information: since the same text segment can be referenced from several places, ideas can be expressed more modularly, i.e. with less overlap and duplication;
>
> 7. consistency of information: references are imbedded in their text--if the text is moved, even to another document, the link information still provides direct access to the reference;
>
> 8. task stacking: the user is supported in having several paths of inquiry active and displayed on the screen at the same time, such that any given path can be unwound to the original task;
>
> 9. collaboration: several authors may collaborate, with the document and comments about the document being tightly interwoven (this feature is just beginning to be explored). [numbers added] (18:56)

Disadvantages. Conklin goes on to describe two classes of problems with hypertext, implementation problems and problems endemic to hypertext. The implementation problems vary from system to system and include, "delays in display of referenced material, restrictions on names and other properties of links, lack of or deficiencies in browsers, paucity of . . . powerful specialized "views" of networks . . . (18:56).

Two problems are endemic to hypertext systems, disorientation and cognitive overhead. Disorientation is caused by the great number of dimensions through which a reader can move. Conklin cites two solutions, "graphical browsers and query/search mechanisms" (18:56-57). Cognitive overhead is related to the mental "overhead of meta-level decision making" of which link/path to follow. The user has to decide if a specific link is worthwhile to follow. This problem is eased through:

> having the cross referenced node appear very rapidly . . ., providing an instantaneous one to three line explanation of the side reference in a pop up window . . ., and having a graphical browser which shows the local subnetwork into which the link leads. (18:59-60)

Other authors confirm the problems of disorientation and excess overhead in hypertext (1:831; 18:56-59; 6:261; 39:841-845; 81:398).

Both implementation and endemic problems are related to the area of browsing to find information in a hyperdocument. Because of the significance of the browsing problem, the next section concentrates on techniques that hypertext systems use to reduce disorientation while browsing.

Orientation Techniques

The worst effects of disorientation can be remedied by orientation tools that are prominent, ever present, and predictable (7:43). The following tools work together in various hypertext systems to provide the user with orientation cues.

Graphical Browsers. The most common method of browsing a hypertext system is through a graphical browser (see Figure 1). This browser displays the structure of the hyperdocument as graphs which provide contextual and spatial cues to the user about the nodes he is viewing and how they are linked to each other (18:7). A browser may be global or local. In the global mode, it displays all links available in the system. While in the local mode, only those links that emanate from the current node are shown (76:39). A global map is helpful for small hyperdocuments, but as networks become large (over 50 links) global maps "become quite overwhelming" (76:39-40).

Some systems display this network permanently on part of the screen (6:256) and others allow it to be brought up as an individual node (39:837). Systems may automatically compute the graphical browser (39:837), require the author of the hyperdocument to create them (7:39), or not provide them (1:830).

Certain factors combine, however, to make it impossible to alleviate the disorientation problem with a browser alone:

a) large number of nodes,

b) large number of links,

c) frequent changes in network,

d) slow or awkward response to user control inputs,

e) insufficient visual differentiation among nodes and/or links, and

f) non-visually oriented users . . . (18:57)

Because of this many hypertext designers have added additional orientation tools to their systems.

Bookmark. Bookmarks allow the reader to mark or unmark pages as they navigate a hyperdocument. If they get lost within the document, they can easily flip back to a page they have marked (7:39-40).

Filters. These are typically used to mask the detail of a network so that the user is presented with a manageable level of complexity. They often provide a capability for the user to shift the level of complexity as desired (18:57, 83:71,76).

Guided Tour. This concept is present in many hypertext systems, although by different names: trails, paths, active path (81:400), tour, and guide (83:65). Guided tours provide an ordered list of nodes to follow which allow the user to "read the material in the suggested order as if it were a linear document" (18:15).

Indexing. Two methods of indexing can be used, alphabetical and hierarchical. Indexing allows the user to look up an alphabetized list of topics and key words included in the hyperdocument. A hierarchical index provides a table of contents that displays the overall structure of the hyperdocument (74:11-12).

Path History. This feature provides reversibility by maintaining and displaying a list of previously reviewed nodes. A user can quickly jump back to any of these nodes by activating the link within the path history (74:15). Some systems automatically record the path followed so that the user can backtrack at any time (1:830). A related concept is the use of "breadcrumbs". A small marker is displayed in a node to show the user they have already seen the material. These crumbs allow the user to call up a list of recently visited pages at any time and to jump to any such page. The system limits the length of the list to avoid disorientating the user with a large menu. "Thus, crumbs represent pages . . . read recently, and imaginary birds remove breadcrumbs the reader leaves unvisited for more than thirty pages" (7:42-43).

Search and Query. Two methods are used for search and query, content search and structure search (39:842). The content search examines all links and nodes independently to locate a match for a keyword or string (31:240). A structure search looks at the hyperdocument structure to match a given pattern (6:262). Desirable features in a search function include the ability to specify "case-sensitivity (i.e., whether to ignore upper/lower case), boolean expressions (terms combined with AND, OR, NOT), and "wild card" characters that do not have to be matched" (74:12).

Tabletops. Again, these are described by several different names depending upon the hypertext system: fileboxes (39:838), composites (6:258), webs (18:29), collections (18:31), and stacks (18:32). Tabletops compile related nodes into one group which can be saved, edited, and recalled as a group at any time (81:402-403).

Hypertext Classifications

Halasz' method for classifying hypertext systems (as described by Nielsen) specifies four dimensions as shown in Table 2. This classification scheme allows identification of hypertext systems over the entire scope of hypertext development and use. It can be used to give an overall picture of any hypertext system.

Conklin, on the other hand, classifies hypertext systems as shown in Table 3. The scheme in Table 3 is a combination and further clarification of two categories in Table 2, scope of the user target and browsing versus authoring.

The classification in Table 3 is important because it clearly identifics the end purpose of a hypertext system. The reference system proposed in this thesis is categorized as a browsing system.

TABLE 2

Hypertext Classification Scheme 1 (59:28)

generation	original (1945 to 1980) and current (1980s)
scope of the user target	single user, work group, corporate division, and the whole world
browsing versus authoring	information presentation versus network creation and manipulation
task specificity	general, some task specificity, and task specific

TABLE 3

Hypertext Classification Scheme 2 (18:9-10)

macro literary systems	. . . large on-line libraries . . . [for] all publishing, reading, collaboration, and criticism . . . within the network.
problem exploration tools	tools to support early unstructured thinking on a problem, in which many disconnected ideas come to mind . . .
browsing systems	similar to macro literary systems, but smaller scale - systems for teaching, reference, and public information, where ease of use is crucial.
general hypertext technology	general purpose systems designed to allow experimentation with a range of hypertext applications--most commonly applied to reading, writing, collaboration, etc..

Hypertext Applications

As discussed above, hypertext can be used in many different ways. Because this research is centered on the use of hypertext for an office reference system, the applications described below are primarily browsing systems, systems that are used for teaching and reference. Appendix B discusses other applications that are possible with hypertext. Another source that describes many applications of hypertext is Hypertext Hands-On! (74:19-33, 107-115).

Navy Aircraft Carrier System. The USS Carl Vinson uses a hypertext management information system called ZOG ("which stands for nothing, but is short, easily pronounced and easily remembered" (68:461)) that supports four online applications: Ship Organization and Regulation Manual (SORM), interactive task management system, technical manuals, and user interface for AirPlan. Each application is described in more detail below.

The SORM is designed to be accessed online by human and agent (application programs) users.

Agents use the SORM to support interactive task management.

> Tasks described in the SORM are copied and their generic portions are instantiated to reflect the particulars at hand (times, people, etc.). These specific structures are then used to track the status of these tasks and generate reports . . . (56:303)

Technical manuals for aircraft and weapons elevators are supported online and integrated with video disk material. This allows a videodisk player to play applicable portions of the disk to provide further explanations of the task.

AirPlan is an artificial intelligence system that assists in launch and recovery of aircraft. AirPlan's function is to notify decision makers when current options change (e.g. proposed landing site no longer attainable due to fuel shortage). ZOG provides input and output interface to AirPlan; allowing users to update scenarios through ZOG and then receive information back in ZOG frames (nodes) (56:303-304).

Hypertext Medical Handbook. The Washington University School of Medicine in St Louis, Missouri, is working on an application that will provide a hypertext medical handbook. Two areas of concern are being focused on: how to assist the authors and editors of the handbook and how can information be retrieved from an online manual.

To automate the paper manual, four steps were required. First, the manual was partitioned into individual nodes based upon the original hierarchical structure of the paper manual. Then each node was assigned a title based upon the nodes first six words (this was later used to label links). Next a parser was used, creating automatic links between nodes to maintain the linear relationships. The final step was further indexing and retrieval.

Several methods are available to the user for finding information in this hypertext medical handbook: pattern matching (i.e key word search), simple graphical browsing, and heuristics for finding optimal locations to start graphical browsing.

The developers of this system are also looking at further enhancements to increase the capability of the system: highlighting for emphasis, annotation, page turner links, permanent path marking, bookmarks, clipboards, and agenda-keepers (34:880-886).

Other medical hypertext projects are also being developed, a drug information database at Johns Hopkins Hospital, a textbook of internal medicine (InforMed Project) at Columbia-Presbyterian Hospital, and a knowledge management system for diagnosis and pathophysiology at Harvard Medical School (74:24).

Other Reference Books. The Oxford English Dictionary (OED) has been keyed in to the computer as a hypertext system and is currently available on compact disk, read only memory (CD-ROM) (65:871-879, 63:292). Research on an electronic encyclopedia is being conducted by the University of Maryland and the University of Washington (83:63-88; 73:189-194).

Educational Applications. Hypertext and hypermedia are being used in several educational applications. Brown University's Institute for Research in Information and Scholarship (IRIS) is completing a research project (Educational Software Program (ESP)) designed to explore the possibilities of non-lineal thinking in human cognition. Courses in English Literature and Plant Cell Biology were augmented with hypertext (76:35; 82:891; 84:12-25; 5:67-88) to supplement information provided in regular classroom instruction. The literature class hypertext includes biographical sketches, explanatory essays, portraits, art works, and timelines (84:18-20; 76:35). The biology class uses linking of online lecture notes to cell models and experiment diagrams so students can "better integrate the material" (76:35).

Several departments at the University of Southern California are working on a joint project (Project Jefferson) to:

> provide a computerized program that will integrate the assignments of the Freshman Writing Program classes, the pedagogical goals of the instructors, the teaching of library and research skills, and the ability to access online information with a minimum of training. (46:34)

Cornell Medical College has loaded its second-year curriculum into a hypertext system which also allows access to a library of related graphics and pictures. Their final goal is to have the entire curriculum in a single hypertext system with access to textbooks and reference books (69:171).

Project Perseus at Harvard University explores how hypertext can improve research in the classics. Instructors and students are working on an extensive system of linked classical literature to encourage and speed the research process (19:51-55; 46:34).

Other hypermedia projects in education and business are discussed in Interactive Multimedia by Ambron (3).

Department of Defense Use of Hypertext

Hypertext use and research within the Deoartment of Defense (DOD) appears to be in the initial stages yet. The literature indicated only three specific DOD uses of hypertext.

The first, and earliest, indication of DOD use was aboard the U.S.S. Carl Vinson. This ten year project, started in 1980, aboard the USS Carl Vinson provided a computer-assisted management system (1; 2; 56; 68). It is described in detail earlier in this chapter.

The second use of hypertext began with the Hypermedia Laboratory, in 1988, under the Defense Applied Information Technology Center. The overall goal of the laboratory was to "create artificial intelligence and hypermedia based programs that make the human-machine interface more human in the machine's responses" (48:1). One specific project of the lab's was to develop a desktop environment with artificial intelligence and hypertext for the DOD Gateway Information System (DGIS) (48; 50;

49). DGIS, as the core computer system of the Defense Applied Information Technology Center (DAITC), serves two basic purposes. First, it serves the DOD community by aggregating information from a variety of information systems into a useful product. Second, it serves as a prototype environment for enhancing information access within the DOD community (51:1). Another related project, on a super mini-computer, allows the user to jump around in a database using hypertext and has a goal of developing a hypertext template to allow non-programmers to create their own applications (10). The Hypermedia Laboratory has since been deactivated (9).

The third use was a two year project, begun in 1988, designing a system to provide rapid access to relevant text, forms, and graphics from Army regulations and training literature. The application uses table of contents, indexing, browsing, and suggestions for further reading to aid the user in finding the needed information (79; 67). The purpose of this document management system was to assist Army personnel in: "learning specific concepts and their interrelationships, authoring a document of their own, or preparing specific forms" (79:569-570).

Other than the three examples above, the only indication that the DOD is using hypertext appears in acknowledgements of articles. The Army, Air Force, and Navy are contracting for general research at private institutions and companies (75; 36; 32; 20; 15; 27; 35; 30).

Factors Affecting Hypertext Usability

The literature in this area consistently identifies one fact: more research is required; few empirical studies have been done; and "extremely few studies exist of the real world impact of using hypertext systems, possibly because hypertext has not been used all that much yet for real world tasks" (58:245). Because of the limited research on hypertext usability, this section presents several articles that, while not specifically dealing with hypertext, may bear on its use. This review is not intended to be all inclusive, rather it presents a brief commentary in each area of concern. The developer/author needs to be aware of these areas in the design of a hypertext system.

When considering whether to use an automated hypertext system for office reference or learning, one of the first things that must be discussed is the affect of presentation method on user reading, comprehension, and selection.

Oborne cites studies reviewing the difference between reading from screen versus paper, which hypothesized that reading from a computer screen would take longer than paper and reader comprehension would be lower. Five studies supported a screen disadvantage; three studies did not support the screen disadvantage; and, one study indicated a screen advantage. Among the reasons postulated for the screen disadvantage were unfamiliarity with reading from the screen, differing page layouts, and contrast differences between paper and screen (62:2-3).

Oborne's own study, designed to control the above factors, showed no difference in reading speed and comprehension between screen versus paper display and negative versus positive character display. Preferences cited by the research subjects, however, were for paper presentation with dark characters on a light background, indicating that it was "most pleasing", "easiest to read", and "preferred more" (62:5-6). This agrees with other studies reviewed by Oborne which cited faster and more accurate reading with reduced contrast ratios between characters and their backgrounds or with dark characters on a light background (62:3).

Another experiment compared performance based upon lines of text displayed on the screen: 9, 18, and 34. Dense single spaced text of 34 lines per screen was least preferred while performance times for the 9 and 18 line versions were similar (73:192).

Granda, in a study of 229 computer users, found that user selected help gave better performance and users of hard copy help performed better than online users (37:37). The same study reviewed the use and effectiveness of several help sources: human, paper, and various types of online information. Each source had differing selection rates; human help was selected by 54 to 63 percent of the users, printed documents by 58 percent, and online by 51 to 61 percent. The time to find the required information varied according to source; human assistance required the longest time (24 minutes), followed by printed (10 minutes) and online assistance (9 minutes) (37:39-40).

The research then looked at each of the three sources and characterized them according to availability, depth of knowledge, and diagnostic capabilities. It was felt that these factors would affect when each help source is selected and the amount of success achieved. Availability of online help is high (provided it is installed) while human and printed availability is varied. Depth of knowledge for printed help is high while human and online are varied. Diagnostic capabilities of humans are high while online is low to medium and printed is low. Overall, human assistance was rated highest, followed by printed documents, and lastly, by online documents (37:40-42).

Granda concluded that four areas account for the more frequent use and higher ratings for human help sources.

> 1) Humans are interactive. . . . [I]n response to . . . linguistic, visual, and auditory cues, humans can quickly change the conversation to meet immediate requirements . . .[;] they can tailor statements to maximize the degree of interaction. . . .
>
> 2) Humans are selective. A human consultant can provide only the information required; other, irrelevant information can be excluded. . . .
>
> 3) Humans can engage in query at multiple levels of dialogue. The user can ask the human consultant questions to define terms and clarify ambiguities at many levels.
>
> 4) Humans can make assessments. A human consultant can ask the user questions to ascertain the level of understanding and diagnose the difficulties. (37:42)

Granda further states that online information could be improved if made more like a human source, with interaction, selectivity, query format, and assessment capability (37:43).

The next step in assessment of a hypertext system concerns the structure of the document itself. Several articles present information in this area. Responses from attendees at the 32nd International Technical Communication Conference indicated that format and the ability to move through an online information system were the most important characteristics of an online help system (38:26). Information with well-defined structures and clear signals of shifts from one part of the structure to the next have proven to increase reader understanding and learning. A poor presentation order can, on the other hand, impair learning (17:110). This might cause a problem in hypertext systems, where users are given more control over the structuring of material.

Based upon database research, Raymond concludes that the structure of a hypertext document can be made more effective by reducing the number of steps required to navigate the document. He suggests this be done through the use of multi-menus which permit a broader base of comparison up front and give the user multiple paths and levels that can be explored (66:254). Other researchers support this idea of breadth overview and further add that menus of link descriptions, global and local maps, indexes, and table of contents may be helpful (52:335-336; 57:300, 79:572-576).

There is disagreement in the literature on how these menus should be presented, however. Akscyn recommends the use of breadth first frames (explicit menus) to provide a tree-like overview of all descendent frames, saying that this provides for strong hierarchical structure (2:118-119). Koved, rather than supporting explicit menus, as Akscyn recommends, provides empirical evidence for the use of implicit embedded menus. Implicit embedded menus differ from explicit menus in that highlighted words within text blocks become menu items (47:312).

In a test comparing explicit tree-like menus to implicit embedded menus, embedded menus showed more questions were answered correctly, fewer screens were viewed, and subjects preferred embedded menus. A second test compared the use of embedded menus to page-turning commands (first, next, etc.). The subjects again preferred embedded menus even though more screens needed to be viewed than with the page-turning method. Another experiment combined embedded menus with a pruning technique (to trim currently irrelevant text). With this "novel" technique, fewer pages were viewed, less time was spent reviewing each page, and less than half the time was required to solve problems (47:316-317).

Koved acknowledges one negative aspect of embedded menus, however. With embedded menus, a frequent user of the hyperdocument would be required to traverse familiar paths to access the details needed. He recommends shortcuts or command language as a way to bypass unneeded menus (47:316).

IBM's study by Granda also considers the different information needs based on user knowledge level. Granda classifies help users along a continuum by their cognitive state, learning, solving, or refreshing. He based this continuum on descriptive user responses describing problems encountered, "I was <u>learning</u> to do X . . . I was <u>solving</u> the problem of making Y work . . . [and] I was <u>refreshing</u> my memory on the Z facility . . . [underlines added] (37:39).

Study results showed that learners were less focussed in their information search and both learners and solvers required more time to find the needed information (37:39-40). This was supported by Gilfoil, who tracked the amount and type of help that was used while learning a computer system. He found that users seek general help while learning and more specific help as knowledge grows (37:36). Based on these results, Granda concludes that learners and solvers could benefit from greater guidance in information search. For online information to be effective, he says it "should be designed to account for the different [cognitive] characteristics" of learning, solving, and refreshing (37:43).

A final consideration in the provision of an online information reference is, "will people use it?" Grice identifies three factors that are critical if online information is to be used.

> 1. Useful-- . . . It must support their work and be the type of information they need in the format they want it.
>
> 2. Easy to use-- . . . [I]t must be developed based upon what has been learned from analyzing the audience for the information and the tasks that they will perform. Once the information is developed, it must be reviewed and tested, revised based upon the reviews and tests, and tested again to ensure that it is accurate and suitable.
>
> 3. Possibly attractive and enjoyable--but function and usability must not be sacrificed for the sake of appearance alone. Of course, online information can, and should, be designed to be attractive as well as useful. (38:34)

Grice then goes on to discuss designer choices that will affect the above factors in the success of the online information. These are: familiarity of subject, support of tasks, ease of learning, navigability, ease of modification, and appearance (38:35).

Familiarity is related to the users comfort with computers. For individuals unfamiliar with computers, an online presentation with the same conventions and format as printed paper will make users more at ease. Also, conventions that are similar to what users have experienced in the past may make the new system easier to learn. All frames should look and act the same for "achieving consistency, familiarity, and ease of navigation" (38:35-37).

Support of users' tasks is the key ingredient in making a system useful. Grice states that the tasks described should match those the users perform in the order they perform them. Further, the online help should be easy to get to and get out of (38:37-38).

Ease of learning and first use create a first impression on the users that affects whether they will use the online information again. A poorly designed information product could "scare people away" so they may not use it. Factors that influence the ease of learning are: information match with users expectations, degree of user guidance offered, task complexity, and familiar and consistent conventions (38:38).

Navigability, as previously mentioned, is one of the disadvantages of using a hypertext document. Grice describes several actions that the designer can take to make the document more navigable.

- Guideposts, such as "screen 5", "screen 1 of 8," . . .

- Actions that can be taken ("push key x to do this" . . .)

- Identification of keys that can be used to take action, such as ESCAPE, END, QUIT, RETURN, or HELP.

- Identification of the overall context (such as a structure diagram or stack map).

- Consideration of most appropriate use of disruptive versus nondisruptive functions (such as Help). (38:39)

Grice states that ease of modification tends to make the users more involved and comfortable with the use of online information. Modification to color, menus, abbreviations, commands, and message text can often be allowed without having a major affect on the online information (38:39).

Appearance of an online information system incorporates four factors, font, density, color, and special effects. Grice cites smaller type font and mixed case as the preferences in online documents. Dense text slows reading on screen. Users like "color for a reason" rather than indiscriminate use of color; excess use of color can be a distraction. Special effects include sound, animation, blinking, color changes, and text buildup. As with color, overuse of these techniques tend to be a distraction (38:40-41).

Although each of the above studies provides information individually that is important in the design and use of a hypertext system, one noteworthy study by Nielsen compiles the results of 30 different papers and 92 benchmark measurements affecting the usability of hypertext. Because of the low number of studies on any given aspect of usability, Nielsen found that a meta-analysis comparing the results of studies addressing the same issue was not possible. Rather, his research focused on reviewing study results on various factors effecting usability to determine the most important issues (58).

Based upon analysis, he narrowed the 98 factors affecting usability to the "matters that really matter"--those with effects greater than 20% between the two conditions studied. The results of his research are shown in Appendix C (58:240-245).

Nielsen found the factor having the greatest effect was based on individual differences--age of the user; older users were over 100 percent less likely to use a hypertext system than those 20 years of age or less. Items rated 2, 3, and 6 also reflect individual differences, leading Nielsen to conclude that individual differences have the most important effect on hypertext system usability. Of the top ten effects, two (effects 8 and 10) relate to differences in the user's task, leading Nielsen to conclude that the user's task is second in effect on hypertext usability. Finally, five effects (effects 4, 9, 12, 13, and 15) related to different behavior when using hypertext compared to using paper or another computer system. Nielsen concludes from this that studies covering usability aspects of other medium should not be applied directly to hypertext (58:243-245). Based upon the above conclusions, Nielsen makes the following recommendations:

> - Individual differences among users[:] . . . make sure to test with subjects who are representative of your intended users, since different people will perform very differently. What is best for one group of users may not be best for another.
>
> - The effect of different tasks: People having different tasks will use hypertext systems in different ways, so different hypertext mechanism are needed to support different tasks. . .
>
> - [O]ne cannot always rely on having the results of usability studies of non-hypertext computer systems transfer to apply also to the usability of hypertext systems, so you are advised to exercise caution if you try to do so. (58:246)

Legal Implications

Depending upon how hypertext technology is used, it may cause difficulties with the copyright law. Federal copyright law provides five distinct rights, "the right to reproduce a work (e.g., to "copy") . . ., the right to adapt or modify a work (i.e., to create a "derivative work"), and the rights to publicly distribute copies of a work, to display it (e.g., on a computer terminal), or to perform it" (44:368).

Copyright law, however, does provide for "fair use".

> [T]he fair use of a copyrighted work, . . . for purposes such as criticism, comment, news reporting, teaching (including multiple copies for classroom use), scholarship, or research, is not an infringement of copyright. In determining whether the use made of a work in any particular case is a fair use the factors to be considered shall include:
>
> (1) the purpose and character of the use, including whether such use is of a commercial nature or is for nonprofit educational purposes;
>
> (2) the nature of the copyrighted work;

> (3) the amount and substantiality of the portion used in relation to the copyrighted work as a whole; and
>
> (4) the effect of the use upon the potential market for or value of the copyrighted work. (44:370)

When hypertext involves the automation of already existing material the copyright law must be considered.

Future of Hypertext

Hypertext is still in the growing stages; much has been written about issues that must be considered in the design of the next generation of hypertext systems (18:60-63; 21:173,175; 39:841-851; 82:893-895; 83:65-66). Some individuals see that hypertext will incorporate techniques from simulation, artificial intelligence, natural language processing, and cognitive science (55:25; 83:65-66; 70). Prior research has been completed at AFIT in this area by Florian and Liddle, who both developed applications combining hypertext with artificial intelligence (33; 54).

Summary

This chapter presented the early history of hypertext, with its birth in 1945 with Vannevar Bush, who foresaw browsing of entire libraries with a machine. Further work was delayed for twenty years, however, due to technology limitations. The features of hypertext (links, nodes, and browsing) were explained and examined regarding how they differ among hypertext systems. Links were highlighted as the key feature, with their critical capability for supporting rapid movement within the hyperdocument. Orientation tools for browsing were reviewed, with emphasis on graphical browsers as the most common method. Features vary depending upon the intended user, scope, and application.

There is no standard hypertext system. Features vary depending upon the intended user, scope and application. Classifications of hypertext systems are also not standard with the industry. In general, the development of hypertext has followed individual paths depending upon the needs of the designer and user.

Several specific applications of hypertext were discussed, including: aircraft carrier management information system, medical handbook, reference books, and various educational applications. Hypertext is being used within the Department of Defense, however, there is no mechanism to keep track of the applications or findings.

The usability of hypertext is effected most by two factors, individual differences among hypertext users and differences in tasks.

Copyright law may or may not be important in the development of a computer application. A hypertext application developer needs to be aware of the legal implications, however, so they can be considered upfront.

The future of hypertext is seen with the overlapping of several different computer technologies to provide greater versatility and useability.

III. Methodology

This chapter explains the methodology used to answer the key research question areas posed in Chapter I, information needed for the reference system, hypertext system development, and hypertext applicability for use with an office reference system. In addition, it presents a review of the literature pertinent to the development of the prototype. Prototyping is examined as it is used in the software development life cycle and justification is provided for its use in this research. The remainder of the chapter follows the steps in prototyping: identify basic needs, develop working model, demo in context, implement revisions, and prototype done.

Prototype Development

The first step in software development is defining the product life-cycle model (29:37). This model should encompass all the activities needed to define, develop, test, deliver, operate, and maintain the final software product. A well defined life cycle model

> . . . provides a basis for categorizing and controlling the various activities required to develop and maintain a software product. A life-cycle model that is understood and accepted by all concerned parties improves project communication and enhances project manageability, resource allocation, cost control, and product quality. (29:37)

For this thesis the standard waterfall life-cycle model is modified to include prototyping. The waterfall life-cycle model as described by Boehm is shown in Figure 3.

Most software development practices specify that system requirements be totally defined prior to initial system design. Prototyping, on the other hand, allows the definition of an initial set of needs, followed by quick implementation, and reiteration of the design as requirements are further defined (11:5). An effective prototype "must work, realistically approximate the actual system's functionality, be ergonomically sound, and be documented" (11:58). It should also be large enough to permit both general and lower level perspectives. The final prototype should define requirements adequately for both the user and developer (11:58-63).

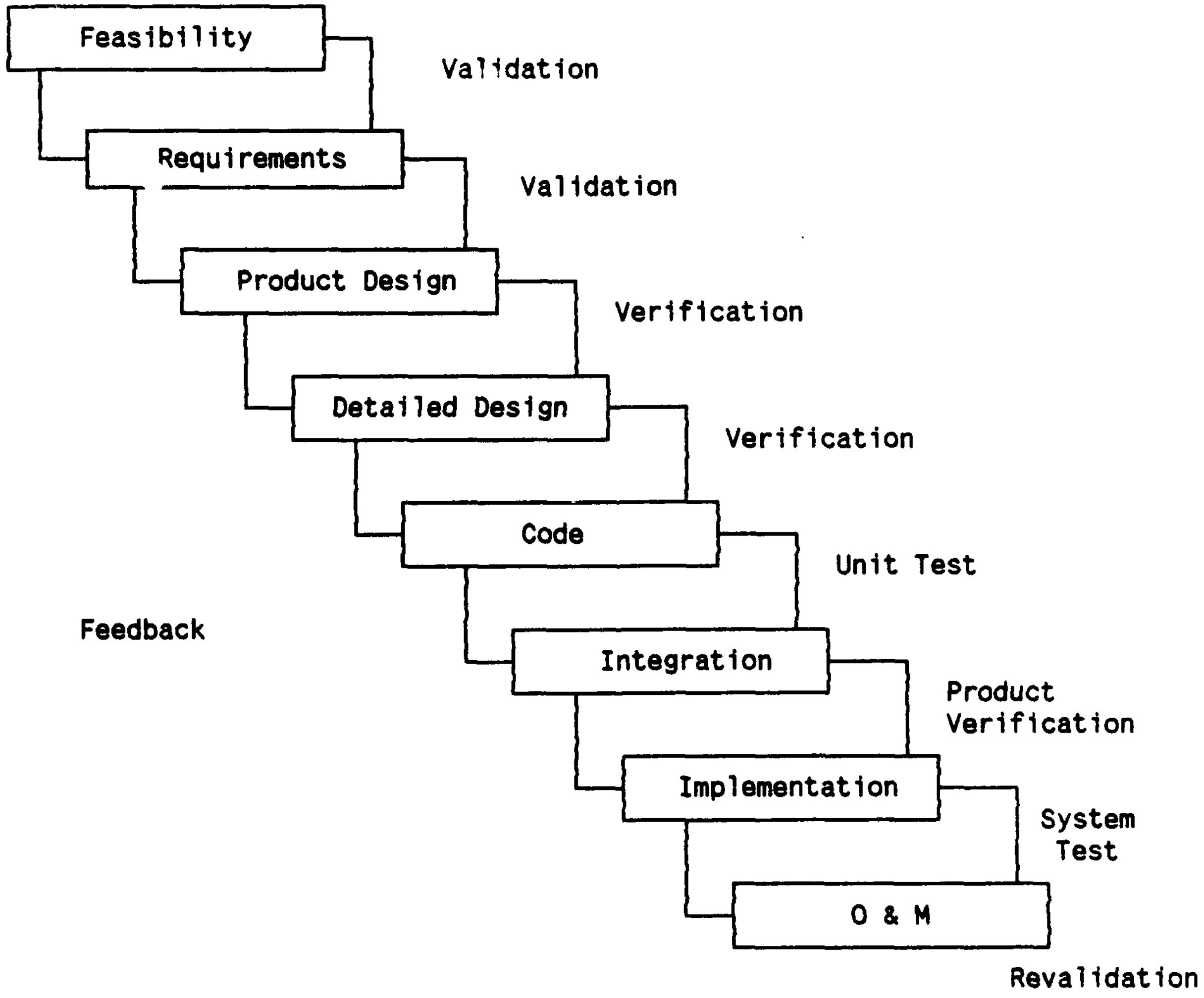


Figure 3. Waterfall Model of Life Cycle (12:36)

Prototypes are considered part of the requirements definition stage of the life cycle. An example of the requirements definition using prototypes is shown in Figure 4. As Figure 4 shows, prototyping requirements definition fits between feasibility and product design of the waterfall model in Figure 3.

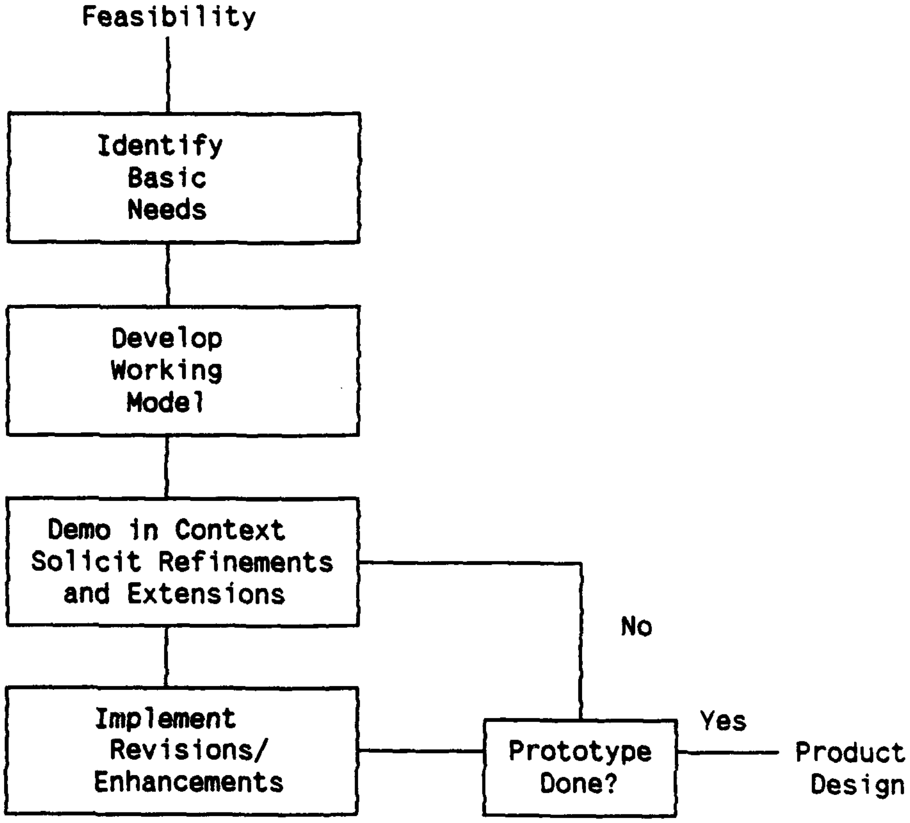


Figure 4. Requirements Definition by Prototyping (11:8,37)

Bielawski states that prototyping is important for large knowledge base problems because it gives four advantages over development of an entire system first.

> 1. A prototype enables the developer to judge whether the system is feasible. Being unable to get the system to perform well on a subset of the intended problem suggests that the system will not be capable of handling the entire problem. . .

> 2. A prototype allows the developer to test the suitability of the development tool that has been selected. Testing even an incomplete system may reveal that the knowledge representation scheme, . . . or its user-interface capabilities may be inadequate for the problem at hand. . . .
>
> 3. A prototype will suggest the amount of time required to build the whole system, an estimate is essential for determining cost/benefit ratio. As a result of the prototyping experience, developers may discover that their initial time estimate was either too optimistic or too pessimistic.
>
> 4. If developers need to gain support of a supervisor before building an expert system [or hypertext system], one means of doing so is to present the supervisor with a working prototype of the system. To suggest the potential the potential of a fully developed system, a prototype often makes a more convincing argument than even a cogent and well-prepared verbal or written presentation of the concept. (8:161-162)

Prototyping is used for this thesis because of the following benefits:

- Provides a vehicle for validating requirements.

- Provides a facility to permit assessment of the impact of the system on the whole user environment.

- Permits early life cycle testing of human/machine interfaces.

- Alleviates project communication problem. (11:38-44)

Prototypes are also readily modified to explore different behaviors and can grow in functionality to accommodate new demands (53:32).

Regardless of the above advantages, however, prototyping is only a correct strategy if the following conditions exist:

- All requirements cannot be prespecified.

- Quick build tools are available.

- Inherent communication gap between project participants exists.

- Active system model is required.

- Rigorous approach [prespecification] is correct once requirements are known.

- Extensive iteration is necessary, inevitable, desirable, and to be encouraged. (11:28)

Due to the relative newness of hypertext, many office personnel are not familiar with its capabilities. A working model is essential to help define hyperdocument requirements and bridge any communication gaps. By selecting a system that does not require programming skills, iterative models can be quickly modified as more information becomes known about the requirements. A complete review of user requirements was beyond the scope of this research, so to communicate with potential future users an active system model was essential. Once an office is familiar with hypertext capabilities, prespecification of the requirements would be appropriate, with product development following the waterfall life-cycle model.

Identify Basic Needs

The first step in a hypertext development project is ascertaining that the planned application is suitable for hypertext. Three attributes should be satisfied, "large body of knowledge separable into smaller components, interrelated components, and user needs only a slice at any time" (72:127). These attributes were satisfied by economic analysis, so identification of needs progressed to the next steps.

Further identification of the basic needs for the prototype development consists of hypertext software selection and design of the knowledge structure.

Hypertext Software Selection.

The hypertext software was selected based upon several criteria. Some of the many features that should be considered in system selection are:

- management of nodes . . . an index of all the nodes/articles that have been created, . . . will be of great benefit.

- indication of links . . .should be simple and easy to manipulate. Marking a phrase or region can usually be accomplished easily, but then it should also be easy to indicate the destination of the link. . . . For text articles, the link buttons should move with the words.

- the range of editing functions available (i.e., copying, moving, insertion, deletion, global change within an article, etc.).

- the availability of lists of link names, index terms, synonyms, etc.

- the range of display formatting commands available.

- the availability of search/replace functions for making global changes across multiple nodes.

- control of color (text, background): color can make the text look attractive, but it can also be distracting. Since users are very different in their preferences and tasks, it should be possible for users to reset color usage parameters.

- the capability to easily switch between author and browser modes to test ideas.

- accessing CD-ROM, videodisc, or other devices: new devices are emerging regularly with remarkable storage capabilities. It should be possible to access information on a variety of devices.

- capability to export files to other systems.

- operability on a local area network.

- multi-user, network & distributed databases.

- version keeping (can old versions of an article be stored?).

- graphics & video facilities: are there embedded graphics editors and mechanisms for exploring the videodisc.

- collaboration: can more than one person edit the database at one time? Can components of the database written by several people be merged into one hypertext database.

- data compression: compression algorithms can reduce the size of the database and facilitate distribution of disks or dissemination by electronic networks.

- security control: is there password control for the database or parts of it.

- encryption: can sensitive nodes by encrypted?

- reliability: does the software perform without bugs, and without loosing data.

- integration with other software/hardware.

- browser distribution: does every user of the hypertext have to acquire a copy of the full system or can the browsing part be included with the database. (72:121-124)

In the final selection of a hypertext system for use in an office reference capacity, most of the above features would be critical. For the prototype development, however, not all features were considered essential. For the prototype the following were considered important: index of nodes, easy management of links, wide range of editing functions, capability of easy switching between editor and browser, version keeping, data compression, reliability, and browser distribution. Also, several additional features were added to the above list for this prototype development.

Ease of development was considered critical. The hypertext system should not require the application developer to possess programming skills, thereby allowing quicker development and iteration. Simple non-programming link development should take less time and fewer error corrections than programming link development. This might also be important for selection of an office reference system, thereby making the software more usable by the entire office.

Although an office reference system may be more appropriate based on another computer, to aid prototype development the requirement for a microcomputer-based hypertext system was added. The appropriate computer level (mainframe access through DDN, minicomputer, microcomputer) for the economic analysis reference system was determined during interviews with major command economic analysis experts (Appendix D). A microcomputer based system also provided better capacity to test the developed reference system. The microcomputer system allowed test volunteers to review the prototype on their own computers.

Many commercial companies are consolidating hypertext and artificial intelligence into one software package. Since the purpose of this thesis was to test hypertext capabilities only, a package with hypertext capability only was desired. For a review of the capabilities gained through tieing these two technologies together, however, the reader is referred to the two AFIT theses mentioned in Chapter II (54; 33).

The literature indicated many companies provide hypertext software packages (Appendix E). Unstructured telephone interviews were conducted with the companies and demonstration copies of disks and documentation were requested so the systems could be evaluated. Also, the KMS hypertext system which is available in the Air Force Institute of Technology Engineering School on the Sun Workstation was considered.

Knowledge Structure Design. Design of the knowledge structure typically consists of four steps: "specify goals, market niche & audience, decide on scope of coverage, identify list of topics and components, and choose traversal structures" (72:127-128).

As specified in Chapter I, a complete analysis of user requirements (i.e. goals, audience) was beyond the scope of this research. The scope of the prototype included a adequate number of reference documents to test the capability of hypertext when used in an office reference system. The prototype focused on adapting the selected reference documents to hypertext (i.e., breaking into nodes and linking).

An initial list of topics and components was identified through structured telephone interviews with economic analysis experts from several major commands and schools: Strategic Air Command, Military Airlift Command, Tactical Air Command, Air Force Cost Center, Sheppard AFB Cost Analysis Technical School, and Air Force Institute of Technology Professional Continuing Education (PCE). The interviews determined what reference documents are used to prepare economic analyses, the most frequent types of economic analysis performed, and the knowledge base of the preparer. Other questions concentrated on the most common problems in economic analysis preparation, with the prototype then concentrating on better methods of tying together documents to solve the problems. The specific questions asked are shown in Appendix D.

To test the concept of prototype development, an economic analysis instructor from the Air Force Institute of Technology functioned as the prototype system user. Discussions helped determine the final selection of reference documents and the traversal structure for the hypertext system.

Due to the large number of reference documents available on economic analysis, selection of the ones to include in the office reference system required some thought.

> Selecting documents to incorporate in a hypertext document is not a simple problem. The goal should not be to build a hypertext encyclopedia, a hypertext reference manual, or a hypertext version of system documentation. Instead, the goal should be to enhance the performance or enjoyment of people performing their jobs, using hypertext when it is appropriate

> to improve the access to and the usefulness of information. This view implies that the choice of documents to include in a hypertext should be dictated by a user and task analysis. Only when the intended users must combine information from more than one document to meet the information needs of the tasks they carry out is a multi-document hypertext called for. . . .
>
> So when should two documents be integrated in a multi-document hypertext? The motivation for combining more than one document in a hypertext is the conviction that each document adds value to the others, that the whole is greater than the sum of the parts. . . . (36:51-52)

Document selection was not a great problem due to the scope of this research prototype. In a full reference system on economic analysis, however, the selection of documents would be a greater concern.

Develop Working Model

Steps in Development. Shneiderman lists several steps in development of a hypertext document:

- Collect or create material.
- Develop a style sheet for writing articles and creating links.
- Ensure appropriate cross referencing to related concepts.
- Arrange for editing of text and graphics.
- Secure legal permissions.
- Create database in proper formats . . . (72:128)

The analysis in this stage can required the most effort of prototype development, as was the case with the Army document management system discussed in Chapter II. The document management prototype required an initial heavy load of manhours to develop an understanding of the documents and the selection of concepts for the link structure was found to be the most difficult part of the development. The developers of the Army manual project found the most time consuming effort, however, was the scanning, proofreading, and correcting of selected information (67:10).

For the economic analysis reference system prototype the documents identified in knowledge structure design were collected and reviewed. A flatbed scanner was used for documents not available in computerized form. The scanner transformed printed paper copies to a computerized file. Since existing reference documents were used, a style sheet was not required. Appropriate cross referencing within and between documents was determined after establishing the basic documents in the hypertext system. Cross referencing is explained in more detail later in this chapter. Legal permission for use was secured where necessary. In creating the actual hypertext database, the following information became important.

Development Considerations. Creating a hypertext document in the proper format requires additional considerations beyond concerns for good writing. The following list, developed by Shneiderman, is based upon years of experience with hypertext document development. Each of these requirements were considered in the development of the prototype.

> 1. Know the users and their tasks: Users are a vital source of ideas and feedback; use them throughout the development process to test you designs. . . . Study the target population of users carefully to make certain you know how the system will really be used. Create demonstrations and prototypes early in the project; don't wait for the full technology to be available.
>
> 2. Meaningful structure comes first: Build the project around the structuring and presentation of information, not around the technology. Develop a high concept for the body of information you are organizing. Avoid fuzzy thinking when creating the information structure.
>
> 3. Apply diverse skills: Make certain that the project team includes information specialists (trainers, psychologists),

content specialists (users, marketers), and technologists (systems analysts, programmers), and that the team members can communicate.

4. Respect chunking: The information to be presented needs to be organized into small "chunks" that deal with one topic, theme, or idea. Chunks may be 100 words or 1000 words but when a chunk reaches 10,000 words the author should consider restructuring into multiple smaller chunks. . . .

5. Show inter-relationships: Each document should contain links to other documents. The more links contained in the documents, the richer the connectivity of the hypertext. Too few links means that the medium of hypertext may be inappropriate, too many links can overwhelm and distract the reader. . . .

6. Be consistent in creating document names: It is important to keep a list of names given to documents as they are created; otherwise, it becomes difficult to identify links properly. Synonyms can be used, but misleading synonyms can be confusing.

7. Work from a master reference list: Create a master reference list as you go to ensure correct citations and prevent redundant or missing citations. Some hypertext system automatically construct this list for you.

8. Ensure simplicity in traversal: Authors should design the link structure so that navigation is simple, intuitive, and consistent throughout the system. Movement through the system should be effortless and require a minimum of conscious though-t. Find simple, comprehensible, and global structures that the readers can use as a cognitive map. Be sensitive to the possibility that the user will get "lost in hyperspace" and develop the system so recovery is simple.

9. Design each screen carefully: Screens should be designed so they can be grasped easily. The focus of attention should be clear, headings should guide the reader, links should be useful guides that do not overwhelm the reader. Visual layout is very important in screen design.

10. Require low cognitive load: Minimize the burden on the user's short-term memory. Do not require the user to remember things from one screen to another. The goal is to enable users to concentrate on their tasks and the contents while the computer vanishes.

11. Early reviews: Subject the database to technical, legal, and management reviews as early as possible. As the database becomes larger, changes become more difficult to make.

> 12. Maintain multiple perspectives: When authoring, try to balance the technical requirements of the system with the user's perspective and the organizational use of the database. (72:125-126; 74:62-63)

Know the Users. The current manual reference system is used by individuals with a wide range of knowledge in economic analysis, the same would be true of any automated system developed. In addition, users of an automated system would have a wide range of knowledge and experience with computer operation. The range of knowledge and experience in both economic analysis and computer use were considered in the design of the prototype. Three factors were important in this area:

> . . . Strive for consistency. . . . Consistent sequences of actions should be required in similar situations, identical terminology should be used in prompts, menus, and help screens, and consistent commands should be employed throughout. . . .
>
> Enable frequent users to use shortcuts. As the frequency of use [and knowledge of economic analysis and computers] increases, so does the desire to reduce the number of interactions and increase the pace of interaction. Abbreviations, special keys, hidden commands, and macro facilities are appreciated by frequent knowledgeable users. . . .
>
> Reduce short-term memory load. The limitation of human information processing in short-term memory ("seven plus or minus two chunks") requires that displays be kept simple, multiple page displays be consolidated, frequent window motion be reduced, and sufficient training time be permitted for codes, mnemonics, and sequences of actions. Where appropriate, online access to command syntax forms, abbreviations, codes, and other information should be provided. (71:61-62)

Structure Comes First. According to literature, the introduction of a hypertext document needs to convey the subject and structure of the information contained, much like a book's table of contents. Three strategies are possible:

> a. Make the root document an overview that contains links to all major concepts in the database [glossary strategy].

> b. Adopt a hierarchical approach in which the links in the root document are major categories [top-down strategy].
>
> c. Organize the root article as a list or table of contents of the major concepts in the database [menu strategy]. (72:126)

Inter-relationships of Documents. Since the economic analysis reference system contains more than one document, the amount of integration between documents must be considered. As discussed in Chapter II, excess linking causes problems of disorientation and cognitive overload. In an early hypertext model, linking of multiple documents was so extensive that structures of the individual documents disappeared (36:55). If regulations are to be incorporated into the reference system, however, separate structures must be maintained. Glushko presents a framework for linking in multi-document hypertext by considering four classes of links.

> Explicit Intra-document links Links of this class explicitly connect two parts of the same document, and include footnotes, "See also" cross-references, and pointers to figures, illustrations, or other nontextual components. These links are the easiest to identify when converting existing texts to hypertext and they are probably the most usable and useful as well. . . . As a result, when printed documents are converted to electronic ones it is essential to exploit this sort of knowledge by capturing the explicit intra-document links first.
>
> Implicit Intra-document links These links are those that are part of the logical structure of the document but which may be impossible to make explicit in the printed form because of limitations in the medium, the nature of the writing and production process, or publishing conventions. For example, . . . the first appearance of a Glossary item in the text can readily be linked to its definition in the "back of the book" Glossary in electronic form. We think that links of this type pose little risk of disorientation or cognitive overload because they follow naturally from the printed version of the document, so we consider them nearly as important as the explicit intra-document links.
>
> Explicit Inter-document links Like the explicit links within a document, these links are easy to identify because they follow presentation conventions in the printed document and are often

> collected in the reference or bibliography sections. Yet we think that they pose more challenges for the hypertext designed and reader than intra-document links, because it is much harder to predict the extent or usefulness of the information at the end of the link. An author may cite another document for many different reasons and the cited documents may add little value. To link or not to link hinges on the issue of complementarity . . .
>
> Implicit Inter-document links These are the links that might be closest to the vision of hypertext, namely links that are not explicit between related documents but that can be extracted by careful and creative analysis of the two texts and the relationship between them. But, . . . there is no consistent "rhetoric of hypertext" that makes it easy for a reader to understand what such links mean and what is likely to appear at the link destination. . . . (36:57-58)

Glushko emphasizes that the intra-document links be established before attempting to use inter-document links (36:58). In the Army field manual project, the creation of a linked database turned out to be the most difficult portion of the entire development effort (67:12).

Demo in Context (Test and Evaluation)

Prototype testing and evaluation is dependent upon how many iterations have been performed. Testing of early iterations concentrates on:

> macro acceptance of the thrust of the model by the user. Are we in the ballpark? . . . [D]etection of gross oversights. Did we miss a record type? [and] user familiarity and comfort in operating the model. (11:72)

Testing of later iterations concentrates on "discovering missing or incorrect function, testing ideas and suggestions, [and] improving the user/system interface" (11:72). Since the developed prototype was tested only once, it concentrated on the level of macro acceptance by the user.

The design of a hypertext system affects the useability of the reference system. The author (of the hyperdocument) is responsible for providing the right information with the right nodes and links. The design should provide two characteristics to the user, accessibility and relevance.

> Information is accessible if users can find it both when they know what they are looking for and when they do not. Information is relevant if it is the appropriate type and size of information module, both comprehensible and useful. The key to accessibility and relevance for specific user groups is a carefully engineered matrix of information, an array that enable users to move efficiently on clearly mapped paths through nodes of information . . . (41:47)

In testing computer systems for office use, three measurable factors are considered paramount, ease of learning, low error rates, and subjective satisfaction (71:17). Ease of learning refers to the time required to learn the commands to use the system; error rate refers to the errors made in carrying out a specified set of tasks; and subjective satisfaction refers to how much users liked the overall aspects of the system (71:14-15).

Based on the above information, a questionnaire was developed for testing the prototype; it is shown in Appendix F. Volunteers were solicited from the Air Force Institute of Technology community. Student and faculty volunteers were asked to use the prototype on any computer for any amount of time and then answer the questionnaire. AFIT students from the both the 1990 and 1991 classes were asked to participate in the test, thereby attempting to include a broader spectrum of computer knowledge in the test.

Office applicability (update capability of hypertext to accommodate regulation changes) was also partially tested during system development.

The researcher tested this capacity by adding, deleting, and updating nodes of text to see how the hypertext network reacted.

Implement Revisions

Comments from the first demonstration test were reviewed and used in modification of the structure and format for the final prototype. Modification concentrated on improving the user interface and general format of the prototype.

Summary

Chapter III discussed the methodology used to improve the economic analysis reference system through hypertext automation. The use of prototyping was explained and justified. This prototype research involved a four step process. Identification of basic needs included the hypertext software selection and design of the knowledge structure. Development of the working model followed specific steps, and at the same time, incorporated lessons learned over years of prior hypertext research. Demonstrate in context and testing was done through a questionnaire based upon those factors important in system usability. Finally, one iteration in prototype development was made based upon comments of the demo in context.

IV. Results and Discussion

This chapter describes in detail how the methodology of the previous chapter was followed. It is separated into six sections that follow the prototyping requirements definition as shown in Figure 4 and explained in Chapter III. It further incorporates and answers the research questions posed in Chapter I.

Hypertext Software Selection

From discussions with hypertext software companies and reviewing demonstration packages and documentation the HyperWriter hypertext software was selected for the prototype development. Five factors contributed to the selection of HyperWriter:

1. No programming knowledge was required, although a simple scripting language could be used if desired.

2. The program runs on an IBM compatible microcomputer.

3. Single use system (i.e., not combined with artificial intelligence).

4. Browser distribution is unlimited.

5. Link creation is very easy.

The selection of HyperWriter (TM) for this research does not indicate or imply that it would be the best software for development of a total economic analysis reference system. HyperWriter was selected because, based on the limited number of software packages reviewed, it appeared to best fulfill the needed criteria to demonstrate the concept. Appendix G provides an overview of HyperWriter's requirements and capabilities.

A microcomputer hypertext-based system was selected for the prototype due to simplification of the development and testing effort. The interviews with economic analysis experts from the major commands indicated that a microcomputer-based system would also be most appropriate for use throughout the Air Force. A potential problem exists however, due to the varying hardware and software configurations in use in the target offices for the economic analysis reference system (13; 78; 40; 77).

Knowledge Structure Design

Interviews with economic analysis experts from various organizations revealed varying information about economic analysis. The most frequently identified reference sources were AFR 173-13, U.S. Air Force Cost and Planning Factors, AFR 173-15, Economic Analysis and Program Evaluation for Resource Management, AFP 178-8, Economic Analysis Procedures Handbook, The Base Level Cost Analysis Handbook, U.S. Air Force Military Family Housing Economic Analysis Manual, and Military Construction Program Economic Analysis Manual. Additional locally developed unique information was also used. The most frequent types of analyses performed were Military Construction Program and Military Family Housing Projects. The general knowledge level of individuals preparing analyses varied from untrained/uneducated to being very experienced with analyses techniques and processes. There appeared to be no consistent problem area with analyses; problems cited were in documentation, misuse of indices and factors, and logic (13; 78; 40; 77).

Those reference documents indicated by the interviews were gathered and used as the basis for discussions with the user. The user suggested the course text for QMT 345, Economic Analysis for Cost & Pricing (64) as another source of information. From the discussions, it was decided to center the prototype on information from The Base Level Cost Analysis Handbook Chapter 11, QMT 345 course book, and regulations (26; 64; 23). An additional document used in the prototype was an economic analysis checklist developed by HQ MAC (28). The specific thrust of the prototype was to be on the broad concepts of economic analysis including terms, definitions, and monetary techniques (45).

Develop Working Model

Documents and Structure. The documents selected for the prototype (23; 64; 26; 28) were gathered and prepared for use. The QMT 345 course book was already available in computer form but the others were not. A checklist developed by HQ MAC scanned with minimal errors. Chapter 11 of the handbook was scanned in with some success. The majority of the text scanned with only minor errors (10 to 15 per page) resulting from slightly illegible print. Pages that had lines, underlines, smaller font, nonstandard symbols, or dense text resulted in many more errors per page, and in some cases, made the page unscannable. The format for AFR 173-15 (small font, two columns, condensed print, bold-face, and lines) made it virtually unscannable. Although the current regulation was not available in computer format, a draft of an upcoming revision of AFR 173-15 was available on the computer. Therefore the draft regulation was used.

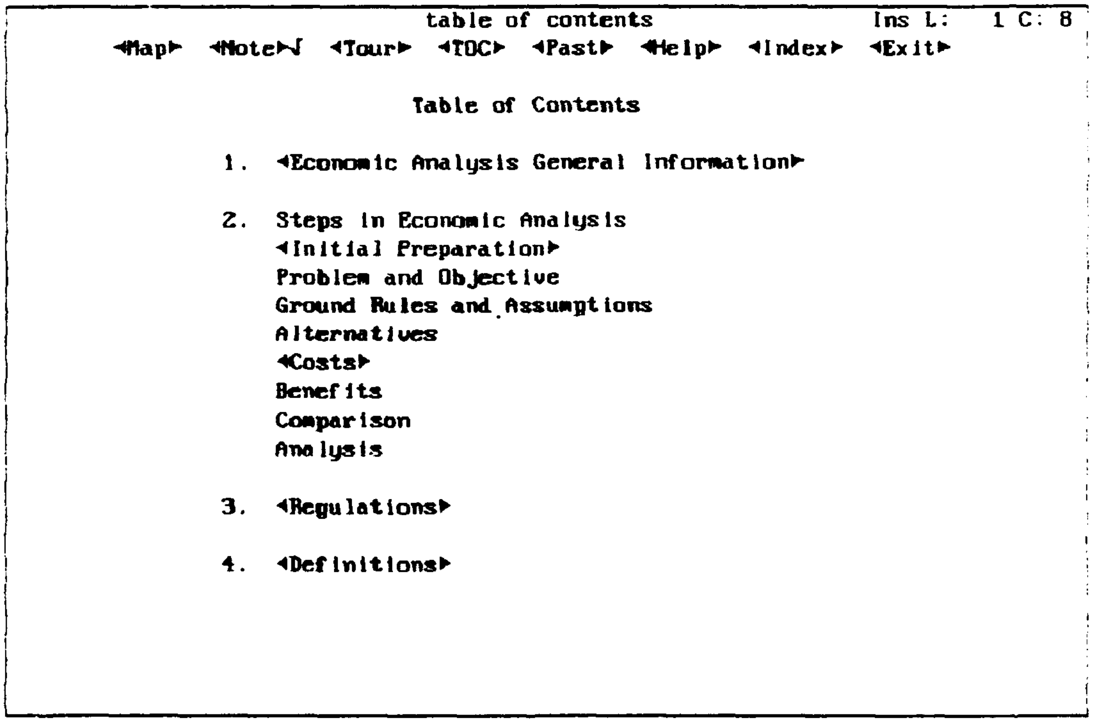
While the scanning of good quality text was fast and required minimal corrections, poor quality text resulted in the need for many hours of correction. Even with the limited number of references used, an estimated 30 hours were spent in initial corrections and formatting of the text. The above problems support one of the lessons learned during a previous hypertext development effort.

> If you are going to rely on scanning in large amounts of text, invest in the best quality scanning system you can afford. The one we selected was not as good as we expected, and that slowed us down considerably. (67:14)

All references selected for the prototype were developed by government sources except one, the handbook. Because an outside contractor developed this for the Air Force the copyright laws was a concern. Discussions with the agency who the handbook was developed for, however, revealed that the handbook could be reproduced or automated (16) without violation of copyright law.

The development considerations discussed in Chapter III were important in the next step of creating the database in the proper format. This document and link structure analysis would probably have been the most labor intensive part of the development effort except for software problems which are discussed below. An estimated 30 hours were initially spent on document analysis prior to importing any information into the hypertext system and another 20 were required once the documents were imported. More time spent before importing the text could have saved time later, however, as the researcher found herself reconsidering and changing the link structure while developing the prototype. Peter Beck of The Analytic Sciences Corporation, who has developed hypertext applications in the past, also found this to be one of the important lessons learned when starting a hypertext project (4).

System Development. Because ,e wide range of knowledge of expected users made consistency, shortcuts for frequent users, and reduced memory load important, certain features were used in the prototype. A single control panel was used to provide new users with an interface that was easy to learn and which appeared on every window of the reference system. The features in the control panel were designed to simplify recovery if the user gets lost. The top of Figure 5 shows the control panel used in the prototype. Quicker access to these and additional features were made available for frequent users through a pull-down menu system and function keys. Reduced memory was facilitated in three ways, through HyperWriter's online help system, a help system developed as part of the prototype, and a consistent format of windows with titles.



```
                Table of Contents

1.  ◄Economic Analysis General Information►

2.  Steps in Economic Analysis
    ◄Initial Preparation►
    Problem and Objective
    Ground Rules and Assumptions
    Alternatives
    ◄Costs►
    Benefits
    Comparison
    Analysis

3.  ◄Regulations►

4.  ◄Definitions►
```

Figure 5. Prototype's Table of Contents

The overall structure for the reference system was based on the steps in economic analysis and a menu strategy with table of contents is the starting point for further browsing. The main portion of Figure 5 shows the prototype's table of contents. Further structure for the reference system follows a card-based design which limits node information to what would fit within each screen.

Once the selected documents were imported into the hypertext system and linearly linked as nodes of text, additional links were added within and between documents to show inter-relationships. Explicit and implicit intra-document links were used freely and explicit inter-document links were used where specifically indicated by wording of the text.

The first iteration of the prototype file was 88K and contained 160 windows with 250 links.

Software Problems. As mentioned above, some software problems were encountered in the development effort. Overall, HyperWriter was extremely easy to learn and use; the researcher learned to use the system in one weekend. There are a few bugs in the programming that need fixing, but nothing that prevents adequate use of all the intended functions. The problem comes in the error checking algorithms of HyperWriter.

One action that may cause damaged files is the cutting, copying, or pasting of links (61:94). During restructuring and normal edit functions on the imported documents cutting, copying, and pasting is normal and frequent. The first step in any of these operations is marking a block of text for action. Being human means that eventually a link will accidently be marked for one of the operations causing damaged files. If HyperWriter then queried the user before performing the damaging operation no inadvertent file damage would occur. HyperWriter, however, lets the user blithely corrupt the files, which occurred several times.

A second undocumented action that also caused erratic results and damaged files occurred during importing of text. HyperWriter requires ASCII file format for imported text. Again, it doesn't query the user if the file format is incorrect. Although the file will not import correctly some unseen characters are imported into that HyperWriter window, causing later problems during cutting, copying, and pasting.

Fixing these bugs and adding error query algorithms, the HyperWriter program should allow very quick and easy development of an application. NTERGAID is working on a utility that will allow a developer to recover from corrupted files. At the current time the only way to restore a corrupt file is to send it to NTERGAID.

Demo in Context

One iteration of the prototype was reviewed and tested. In the design of a full reference system, however, early and frequent reviews would be critical.

Questionnaire Results. AFIT student volunteers were solicited to test the prototype. Of the 60 students who volunteered 57 returned questionnaires, for a 95 percent response rate. The results of the questionnaire are shown in Appendix H.

Experienced computer users were the predominant respondents in the test (88 percent). This is probably not representative of the Air Force community who would be using the reference system. The volunteers ranged in grade from O-2 to O-4, ranged in age from 25 to 35, and all were either currently working on a masters degree or had recently completed one. A complete assessment would be needed to determine the individual characteristics of the end users of an economic analysis reference system, but undoubtedly the range in grade, education, and age would be greater than those testing the prototype.

Few volunteers had problems installing the software or running the prototype with the simple installation instructions (Appendix I). The majority felt that the opening screen was sufficiently explanatory and adequate to allow them to activate links to the online help system.

The researcher's intention in designing the online reference system help was that it be totally self-sufficient (i.e., no documentation manual required). Based on the responses to questions about the online help, only 34 percent felt that a users manual would not be required. In addition, 58 percent indicated that the instructions on how to use the system needed improvement and 29 percent felt that the online help needed improving. It is not clear, however, whether an improvement in the online instructions and help would lead more individuals to state that a users manual isn't required. Even with the indications that more help was required, 67 percent found the system easy to learn and 63 percent found it easy to use. Only 23 and 16 percent respectively felt that the system was both difficult to learn and use.

The control panels that were provided to assist the beginning user received mixed reviews. The table of contents function was the most popular control icon (85 percent found it useful) and the least useful was the past history function (27 percent found it useful). The other control icon functions in order of usefulness were hypertext help, note, map, tours, and index.

The testers felt that a hypertext system would provide a useful office reference system (79 percent). In that capacity, it would be most useful for re'iewing familiar material, followed in order by browsing for general information, first time learning of concepts, looking up an answer to a specific question, and research purposes.

The time required to find and learn information on a hypertext system compared favorably to the use of any other sources of information. Users stated that the system would help them find information more easily (83 percent), would take them less time (75 percent), and they would learn more (69 percent). If they had access to a hypertext reference system, 44 percent expected to use it at least once or twice a day.

The features most liked about the system were data links (70 percent liked), functions (53 percent) and speed of processing (53 percent). In contrast, the least liked features were instructions on how to use the system (58 percent stated improvement was needed), depth and breadth of data (30 percent), and online help instructions (28 percent).

Some volunteers who tested the system also provided comments about the prototype. Their comments are summarized in Appendix J.

Update Capability. The software selected for prototype development allows users to add annotations to a hypertext document. Because of this capability, update is simplified somewhat. As changes to regulations are published, indicating the change is as simple as typing information in a window called from the control panel. Total revisions to regulations are not as simple. A complete redevelopment of that document and existing links to it would be required. Any cutting of currently existing links could cause file damage as discussed above. Other hypertext system's capability in this area would vary however. Any hypertext system would have to be reviewed individually to determine its support for updating.

Implement Revisions/Prototype Done

Comments received in the demonstration and test were incorporated into the prototype. Specific revisions concentrated on supplementing and improving the online help system. The final prototype developed in this research effort is available from the author.

Summary

Chapter IV presented the results from building and testing the economic analysis office reference system. The HyperWriter hypertext software was selected because of its ease of use, microcomputer-base, and unlimited distribution of applications with the runtime browser. Four documents were used in the prototype with the structure and links determined by the researcher.

The demonstration in context and test showed the prototype was easy to use and learn but additional work was required in either supplementing the online help or developing a users manual. These comments were implemented and the final prototype is available through the author.

V. Conclusions and Recommendations

This research has shown that hypertext technology can be used to provide a very practical office reference system. The ease of both use and learning, demonstrated with the prototype, shows that a hypertext reference system can be effectively used in an office environment; finding information would be quicker and users would learn more. In addition, this research demonstrates that a hypertext reference system can be used by individuals with a wide range of experience and reference requirements, from initial learning to reviewing of material. The remainder of this chapter draws conclusions about the research and makes recommendations for further research.

Conclusions

Three general areas of questions were answered through this research, reference documents appropriate for the prototype, hypertext system development, and office environment compatibility.

Reference Documents Needed. The first concern was the determination of appropriate economic analysis documents to give a good test of the hypertext prototype system. Four documents were selected for the prototype development, a regulation, a handbook, a course textbook, and a checklist. Each provided a different unique structure and covered varying detailed information about economic analyses. Some interrelationships existed between the documents but each also serves a distinctive purpose.

The second challenge was getting the selected documents in computerized form. One document was available in computer form, scanning was attempted on the other three to simplify and speed this step. While one document scanned near perfectly, the others achieved varied success. Scanning appears to be an option only with "clean" copy, meaning no underlines, lines, condensed text, font differences, nonstandard characters, columns, or graphics. If any of the above conditions exist in quantity, it will probably be quicker to reenter the information rather than correcting nearly illegible scanned text.

Hypertext System Development. Many commercial hypertext software systems exist at varying prices. Although a system was available on the SUN workstation, it was ruled out due to the difficulties in development and testing. The system selection was based not on cost but on system capability. Educational pricing is available from many software companies, however.

Based on interviews with several major commands, the most appropriate system base is a microcomputer. Even this could cause problems due to varying hardware/software configurations at different bases. Before development of a full reference system, a more complete analysis would be required to determine the full range of current hardware/software configurations and varying hypertext system compatibility with the configurations.

The methodology used to break documents into text blocks (nodes) must start before any system development. Three structures are recommended to simplify the interface and access, glossary strategy, top-down strategy, and menu strategy. Based on the structure of the selected documents, and economic analysis in general, a menu strategy with table of contents was the most appropriate. Nonlinear links need to be established to indicate intra-document relationships first. The next step is establishing of explicit inter-document links where relationships clearly exist and the existence of the link provides useful information.

Differences between individual users and tasks should be a primary consideration in development of the user interface. While much research has been done on improving this interface, caution must be used in applying the results of studies on other computer mediums directly to hypertext. More research in real world applications is needed before the total usability of hypertext can be known.

Office Environment Compatibility. Based upon the prototype test, hypertext use is compatible with the office reference environment. The reference prototype system was found to be easy to use and learn and capable of providing quick access and learning of information.

Update capabilities are available within the hypertext software selected for the prototype. Annotation nodes allow the user to indicate office unique requirements or regulation changes. Complete revisions to regulations will require some redevelopment, with the effort required depending on the hypertext system base used.

Recommendations for Future Research

Four general areas exist for further research. The first deals with government use of hypertext capabilities. Research on hypertext is being accomplished in pockets of organizations throughout the government, each working in a vacuum. An indepth review of organizations involved, applications tested, real-life systems, and research findings could be a starting point for greater use of hypertext capabilities.

The second recommendation deals with hypertext software. One system was used in this prototype, but many others exist that may have been more appropriate for use. Further research should be conducted into what software systems are available, what individual software requirements and abilities are, and how each software system might fit into the needs of the government. The determination of an Air Force supported hypertext system could save money and aid many offices in making use of hypertext capabilities where appropriate.

The third and fourth recommendations are further developments on the prototype of this research. The third recommendation is for a complete user needs assessment regarding an economic analysis reference system. It should identify the range of individual capabilities, specific economic analysis reference documents required, and existing hardware/software configurations. This needs to be accomplished before any further development of an office reference system for economic analysis.

The fourth recommendation is to continue the development and testing of this prototype. Specifically, further research needs to be done on how to improve the user interface for an office reference system. What types of capabilities need to be added to make an office reference system that can be used by individuals with a broad range of both computer and knowledge base training and experience.

Summary

Chapter I described the current problem with the manual economic analysis reference system, indicating that improvement might be possible through the use of hypertext. Chapter II provided an overview of what hypertext is, how it is being used, and factors affecting usability. Chapter III defined the methodology used to develop and test the prototype reference system. Chapter IV provided comments and findings from the development process and this chapter explains how the research questions were answered through each step of prototype development. Four areas for further research were also presented.

In addition to the prototype development and testing, this research was designed to provide a single first reference on the capabilities and concerns relating to hypertext.

Appendix A: CBT Modules on Economic Analysis

1. Title: CBT - How to Document an Economic Analysis

a. Version: 1.0

b. Description: The purpose of this CBT is to acquaint the student with the need for preparing a final document which pulls together all of the work done in an economic analysis for a team that will review and make final decisions on problems concerning allocation of scarce resources. By the end of the lesson, the student should be able to correctly place all of the elements of an Economic Analysis in their proper order.

c. How to obtain: Download from AFCSTC Bulletin Board or via DDN [Filename: EADOC.ARC] or call OPR for copy.

d. OPR: Paula Spinner, SAF/FMCE, AV 227-1152, (202) 697-1152

2. Title: CBT - Time Value of Money.

a. Version: 1.0

b. Description: This a basic level, mini-lesson whose purpose is to familiarize the student with terms and techniques used in financial calculations. By the end of the lesson, the student will be able to calculate present and future values. Ultimately, this mini-lesson will be incorporated into the Economic Analysis module.

c. How to obtain: Download from the AFCSTC Bulletin Board or via DDN [Filename: TVM.ARC] or call OPR for copy.

d. OPR: Paula Spinner, SAF/FMCE, AV 227-1152, (202) 697-1152

3. Title: CBT - On Economic Analysis (EA)

a. Version:

b. Description: This CBT is an integration of a CBT lesson on an "Overview of the EA Process" with the already existing "How to Document an EA" CBT lesson. The purpose of this module is to familiarize the student with the step-wise process involved in developing an Economic Analysis. The lessons to be included within this module are:

1) Overview of an EA
2) How to Document an EA

By the end of the lesson, the student will be able to prepare an economic analysis.

c. Schedule: June 1990

d. OPR: Paula Spinner, SAF/FMCE, AV 227-1152, (202) 697-1152 (22)

Appendix B: Hypertext Applications

Note: This appendix is an excerpt from the documentation for HyperWriter (61:177-194). It is reprinted here with permission of the authors. Although the applications, pros, and cons listed here are specific for the HyperWriter software, many of the same applications could be created on any hypertext system. The specific applications outlined here can be downloaded from the NTERGAID Bulletin Board at 203-366-5698.

Chapter 16: Application Building in HyperWriter

Hypertext is based on an amazingly simple idea -- linking disparate information together by association as opposed to a rigid field/file organization. This idea can be extrapolated into a number of different applications including presentations, personal information managers, and much more. In total, nineteen different applications are detailed here.

Each application described in this chapter has only one or two pages to describe it. To really understand this chapter, it is necessary to load the HyperWriter application files that accompany it and read them concurrently with the chapter. All the applications described can be accessed from the document APPLGUID.HW -- an index to these applications.

An overwhelming advantage to each of these applications is simply economy. By stretching HyperWriter's functionality, many different applications can be constructed with a single software package. This avoids having to purchase different software packages that might go unused. A second advantage is that with this method, you only need to learn a single software package as opposed to several packages.

A useful approach to working with these files, should you choose to use them for your own purposes, is to treat each file as a 'kernel' document. That is, remove the sample data provided from the file and then use that file without data as a basis for future use.

16.1 Application #1: Expert System

An expert system is a type of computer program that guides a user through a decision tree and gives him 'expert' advice based on a predefined knowledge base. The basic action in an expert system is that of prompting the user for input and then branching to the next question in the knowledge base. This corresponds to the action of selecting a link and then following it to its destination. In fact, hypertext can be used to construct simple-to-moderate expert systems that guide the user to an answer and then generate a report of the result.

The demonstration file for this application is called APPLY.HW. Unlike many demonstration files, APPLY.HW is part of a commercially available hypertext application, the Reg-In-A-Box project. Reg-In-A-Box is a hypertext document written by the EPA to provide regulatory information in a hypertext environment. Reg-In-A-Box first became available in the Fall of 1989. More information on the Reg-In-A-Box can be found by clicking the 'About Reg-In-A-Box' button in the REGINBOX.HW file.

For more information on the Reg-In-A-Box project, contact:
United States Environment Protection Agency
Bill Foskett
644 G. Street N.E.
Washington, DC 20002

Generally, expert systems constructed within HyperWriter tend to be on the small side, with between 100 to 200 rules or different decision points. This reflects the very highly focused nature of a hypertext expert system. One of the best applications for a hypertext expert system is to help a reader find information inside a document. In any medium-to-large size document, an expert system can be created that queries the reader and then guides him or her to the correct information.

16.1.1 Pros

The overwhelming advantage of this approach to expert systems is the ease of construction. Traditionally, constructing an expert system had been a complicated task. In some cases, a special software expert called a Knowledge Engineer was needed. Expert systems constructed with HyperWriter can generally be created by the author of the document. A second advantage is that special, usually expensive, expert system software does not need to be purchased. In addition, HyperWriter's runtime can be used to distribute expert systems for free. This is an option that many expert systems do not come with. Finally, HyperWriter's multiple navigational tools can often be used to gain a better understanding of the knowledge base than the tools that accompany a traditional expert system.

16.1.2 Cons

Despite the above advantages to a hypertext expert system, there are a number of disadvantages. To start, true expert system programs are much more flexible for an expert system than is HyperWriter. A true expert system can not only accept user input but can also do calculations with a branching based on this input. Second, an expert system provides extensive facilities for 'chaining' or reasoning both backwards and forwards.

16.2 Application #2: DOS Shell

A DOS Shell is a program that provides DOS with an easier-to-use interface than DOS's traditional command line. Often a DOS Shell is pictorial, in nature replacing commands with icons. This approach is taken here. The demonstration file for this application is DOSSHELL.HW.

16.2.1 Pros

The advantages to this type of DOS shell are simply its ease of construction and its flexibility.

16.2.2 Cons

Some of the disadvantages with this DOS Shell are the disadvantages with any DOS Shell -- they isolate you from DOS. Although this is part of their goal, usually DOS Shells need to be exited at one time or another. And, if you need to exit a DOS Shell, then, really, what is its purpose? Our DOS Shell application relies on utility programs that ship with the Norton Utilities (TM) software package for part of its functionality. Finally, this DOS Shell does not have the sheer functionality that a program such as the Norton Commander (TM) has.

16.3 Application #3: Office Information System

One of the very first major hypertext applications created for internal use at NTERGAID is what we call MEMO - A hypertext memo system that manages our internal communications. The Memo system is chiefly composed of a series of hypertext files. The files used in this application are:

File Name	Use
MEMO.HW	Front end hypertext document to all files.
MEMO_BRI.HW	Memos from Scott to Brian.
MEMO_SCT.HW	Memos from Brian to Scott.
ARTICLE.TXT	Articles from magazines and networks.
ARTICLE.HW	Titles of articles and commentary on them.
DATES.HW	Upcoming dates and events. NTERGAID.
AMOUNTS.HW	Upcoming large cash expenditures that concern NTERGAID.

This collection of files and the cross links between them manage all of NTERGAID's internal communications. This is a fairly complex application; the best way to understand it is to load it and try viewing the different files.

As a backup strategy for this application, all files are simply copied to new file names and then a series of kernel documents are used to create new files.

16.3.1 Pros

The overwhelming advantage to this application is simply that it lets us run NTERGAID much more efficiently. By formalizing our internal communications, the MEMO application has reduced the human factor of memory.

16.3.2 Cons

One disadvantage to this application is simply that it requires human management of the individual files. Some degree of care must be taken that you are not working on an old file when the memo file is updated. A second disadvantage with this application is that HyperWriter isn't a multi-user system. However, in practice this is not really a problem for us because of our working habits. The people thac add to and update this application typically work on laptop computers in isolation from any type of network. By designing this application to use individual files as opposed to a network, this most closely mirrors our working habits.

16.4 Application #4: Educational Courseware

Educational courseware is the application of computer software to specific courses. This has resulted in the term courseware. Courseware generally tries to convey large quantities of course-specific information using techniques such as hypertext.

The demonstration file for this application is called DEMO.HW. This demonstration file is the demo disk for the HyperWriter application Culture 1.0 (TM). Culture is a four megabyte piece of educational courseware that covers Western culture. This demonstration disk gives a good flavor of the facilities that a well-designed piece of courseware might provide. These include both text and graphics, multiple navigational tools such as tours, and a high degree of user-interaction via HyperWriter's Readers Notes feature.

A second courseware demonstration file that shows how to use HyperWriter for multiple choice tests is called HTTEST.HW. This file constructs a hypertext multiple choice test that records the student's progress to an ASCII file called GRADE.TST.

A different approach to courseware, not illustrated here, is that of providing students with raw material in the form of information and then allowing them to link it together. This 'collage' approach to courseware is notable because it tends to involve the student more strongly than other approaches.

16.4.1 Pros

There are several advantages to constructing courseware with HyperWriter. First, HyperWriter's free runtime allows any courseware created to be re-distributed for either free or profit. Second, HyperWriter's ease of use makes creating course-ware very simple. Third, HyperWriter's ability to work with multiple forms of media such as text, graphics and video allows for dynamic, interesting applications. Finally, HyperWriter's many navigational features make navigating through HyperWriter constructed courseware easy.

16.4.2 Cons

The most obvious disadvantage to using HyperWriter for courseware applications is that it has no facilities for student evaluation. However, not all courseware requires this as the Culture Demo shows.

16.5 Application #5: Interactive Questionnaire

HyperWriter can be used to construct interactive multiple choice questionnaires. In this application, a series of links surrounding each answer to a multiple choice question are displayed onscreen. Whenever a user clicks on an answer, a comment window pops up to thank them for their response and then prompts them to go to the next question. As each question is answered, the response is written to an ASCII file. This file can then later be evaluated by the questioner.

The best way to use this type of application might be to create the hypertext questionnaire and then distribute it on floppy disk to the different people in the survey. At their leisure, they can run the application, answer the questions and then return the diskette to you. HyperWriter's ability to auto-detect graphics screens and adjust to them eliminates any worry that this application might not run on their systems.

The demonstration file for this application is HTQUES.HW.

16.5.1 Pros

One outstanding advantage with this application is that HyperWriter's runtime can be freely distributed. This means that any questionnaires created in this fashion can be distributed without royalty payment. A second advantage is that with HyperWriter's @REPORT command, ASCII files can be created that contain the results of the questionnaire. A third advantage is HyperWriter's ease of use. Creating attractive, interactive questionnaires using HyperWriter is quite easy.

16.5.2 Cons

There are two disadvantages to this application. First, questionnaires are technically limited to multiple choice answers. One way around this is to use the Readers Notes facility to accept free input. A second disadvantage is HyperWriter's relatively simple reporting facilities.

16.6 Application #6: Hypertext Technical Support Knowledge Base

After we, at NTERGAID, created the MEMO application, we began to look for other areas to apply hypertext. The second major application we constructed was called TECHSPRT or Tech Support. Tech Support is a hypertext technical support knowledge base. It consists of a series of observations and comments on the use of NTERGAID products. It is primarily built around our original Black Magic product but includes all of our other products. This file is broken down in two ways -- by product and then by possible problem area. This lets our technical support technicians solve problems quickly and easily.

Although this application is constructed as a single document file that covers multiple products, it can just as easily be constructed with multiple files. This will allow much larger knowledge bases to be constructed.

The demonstration file for this application is TECHSPRT.HW.

16.6.1 Pros

The overwhelming advantage to this application is simply the advantage of hypertext itself-the free linking of information. This allows all related information to be tied together. A second advantage is simply the ease of construction. This entire application was built in a little over one night of work. The time-consuming part was sorting through old paper letters for material to build the database around.

16.6.2 Cons

The major disadvantage to using HyperWriter for technical support is that it isn't multi-user in nature. This means that multiple support technicians can't update the database and make comments in real time. To get around this problem, we've adopted the following strategy. When a new technical support problem surfaces or a solution is found, that solution is added to the database with the Readers Notes function and then later merged and relinked into the master document. If you were constructing this application as multiple documents, you might want to use the @SUBFILE and @MASTERFILE commands to ensure that all Readers Notes were exported simultaneously.

16.7 Application #7: Paper Replacement

One of the problems with modern computing is the proliferation of different text files. Every activity you do with a computer seems to generate a text file of one sort or another. This is particularly acute if you use different online services and maintain log files of your sessions. Hypertext can be used to quickly and easily annotate these files with links. The files can then be passed on to co-workers and friends using HyperReader. By circumventing printing out the log files, not only is the step of printing eliminated but the step of making multiple copies for people is eliminated also.

The demonstration file for this application is ACTOR.HW. This file compromises a discussion about the ACTOR (TM) programming language that was downloaded from the WhiteWater Group bulletin board system. After downloading the discussion, a series of links were attached to the messages that clarified various points regarding Actor. Then, once the file was annotated, it was distributed without the need for printing it out.

A second approach to this concept of paper replacement is contained in the file LITERARY.HW. This file contains an English assignment a!cng with hypertext annotation containing the teacher's comments.

16.7.1 Pros

The overwhelming advantage to this application is the cost saving from not having to print out paper versions of documents. A second advantage is that with Readers Notes, the readers comments can easily be added to documents.

16.7.2 Cons

The disadvantage to this application is simply that it's only applicable to people with computers. Whereas paper is a universal medium, this is a medium limited only to those people with computers.

16.8 Application #8: HyperWriter as a PIM

A new type of software product that has surfaced in the past two years is called a Personal Information Manager or PIM. A PIM is a piece of software that is designed to mange miscellaneous scraps of personal data such as dates, times and contacts. Hypertext can be used to construct an application with many of the features of a PIM.

Other applications that can be constructed in the same vein as a PIM are: a personal database, phone call tracking, a personal time-tracking application and a sales tracking application.

The demonstration file for this application is HYPERPIM.HW.

16.8.1 Pros

There are several advantages of using a PIM based on a hypertext document. First, any PIM constructed with HyperWriter is less costly than specialized PIM software. Second, there is little to no learning curve with a HyperWriter-based PIM as there is with specialized PIM software. Third, any PIM constructed with HyperWriter is more flexible and adaptable than specialized PIM software.

16.8.2 Cons

Despite the above advantages of a PIM constructed with HyperWriter, there are a number of disadvantages as well. To start, most PIMs have cal^ulation abilities. They can sum up expenses, calculate dates and more. HyperWriter cannot handle these functions. Second, most PIMs have fairly flexible reporting and sorting capabilities. HyperWriter is limited to a basic text export and a printing function.

16.9 Application #9: Presentations

Unlike many presentation software packages on the market, HyperWriter has no built-in charting or graphics abilities. However, HyperWriter can be used to integrate graphics from multiple application with its screen grabber and .PCX support. In addition, HyperWriter can be used to construct very dynamic and interactive presentations.

Presentations have recently received much attention from the computing press with several excellent articles being the result. One outstanding article that covers creating presentations, although not hypertext presentations, appeared in February 1989 PC Computing magazine. The author was Robin Raskin and the article's title was "Desktop Presentations: On With The Show." Additional articles on creating presentations and the software involved appeared in the October 17, 1989 PC Magazine.

16.9.1 Pros

The greatest advantage to HyperWriter constructed presentations are the dynamic nature of these presentations. Not only can HyperWriter presentations use text and graphics but they can also use videodisc images and sound. HyperWriter is one of the only presentation packages that has these multimedia capabilities. A second advantage to HyperWriter presentations is that relevancy can be added to any part of a presentation with hypertext links. With hypertext links, any part of a presentation can have more information attached to it for clarification. HyperWriter's many navigational abilities can be used to find specific information within a presentation. As a presentation tool, tours can be used to construct self-running versions of a presentation. Readers Notes can be used by the viewer of the presentation to add their own comments to the presentation. Finally, HyperWriter's free runtime allows all presentations to be freely distributed.

16.9.2 Cons

A main disadvantage to using HyperWriter for presentations is HyperWriter's lack of charting capabilities -- a key feature in most presentation packages. This disadvantage is somewhat lessened due to HyperWriter's ability to import.PCX images. A second disadvantage is that HyperWriter lacks the ability to automatically generate handouts. To some extent, this can be addressed with HyperWriter's printing capabilities.

16.10 Application #10: Videodisc Front End

HyperWriter can be used to construct interactive, easy-to-use front ends to videodisc images. The example file, VIDEO.HW, is designed to work with National Gallery of Art Videodisc. It presents a card of information for selected works on the videodisc. Some of the cards allow you to "zoom in" on the images or take a guided tour around some of the images.

16.10.1 Pros

The advantage to this application is that it allows easy annotating and commenting on a videodisc. This addresses one of the traditional objections to videodiscs -- that they are read only mediums.

16.10.2 Cons

The disadvantage to this application is that for each different videodisc you have, a different front end needs to be created. This is not so much a limitation of HyperWriter as it is of the videodisc medium itself.

16.11 Application #11: Software Engineering

HyperWriter can be used to construct an environment for managing and documenting large programming projects. In this application, hypertext links are used to create a structure for viewing source code files. HyperWriter's ASCII file link is used for this.

16.11.1 Pros

There are two major advantages to this application. First, due to HyperWriter's ability to work with extremely large ASCII files, programming projects of any size can be managed. A second advantage is that this application addresses what critics have always accused software development of lacking -- poor documentation.

16.11.2 Cons

The disadvantage to this application is that it must be manually constructed for each programming project. There are no facilities to automatically build this hypertext structure. A second disadvantage is that HyperWriter is not multi-user. This prevents multiple programmers from documenting their work simultaneously.

16.2 Application #12: Technical Documentation

HyperWriter can also be used to construct outstanding technical documentation. With hypertext links, documentation can be annotated and explained at length while, at the same time, expert users can quickly move past material they're already familiar with.

The demonstration file for this application is HW1CHP11.HW. This is actually Chapter 11 of this manual converted to a hypertext document.

16.12.1 Pros

The main advantage to using HyperWriter for technical documentation is simply its ability to link information together. A second advantage is HyperWriter's ability to work with both text and graphics. Finally, any documentation created with HyperWriter can be freely dis𝑡ributed with HyperWriter's runtime.

16.12.2 Cons

The major disadvantage of this application isn't really related to HyperWriter at all. It's simply that people often prefer printed documentation over online documentation.

16.13 Application #13: Writing Environment

Despite HyperWriter's many hypertext-specific features, it can also be sue quite successfully as no more than a multi-window writing environment. Of course, it is one of the few such environments that boasts hypertext linking as well as editing. A hypertext writing environment features instant access to notes, sources, an outline and your writing itself.

This type of writing environment should be used for the initial stages of writing and organization. HyperWriter's linking abilities can be used to help organize a document. Due to HyperWriter's limited printer support, it is probably not the best application with which to generate the final essay or paper. Once most of the writing and organizing is done, HyperWriter's export function can be used to move the file into a wordprocessor or desktop publishing environment where the final version can be printed out.

16.13.1 Pros

HyperWriter is well suited to this type of application due to its linking abilities and its ability to handle multiple windows. In addition, HyperWriter's spell checker and full editor help in writing projects.

16.13.2 Cons

The primary disadvantage to this application is the need to move the information into another program to generate the final essay. A second disadvantage is that HyperWriter does not have a thesaurus.

16.14 Application #14: Hypertext Flash Cards

Hypertext can also be used to construct a modern equivalent to the old fashioned flash card. A hypertext flash card application displays something to remember on a card and then a link from that card to a comment window contains its definition or description.

The demonstration file for this application is called FCARD.HW. It contains a list of vocabulary words culled from <u>The Handbook of College Entrance Examinations</u> as well as their definitions. Two things are central to this application. First, a screen background used to create the card appearance shown above. Second, in addition to manually going through the cards, a constructed tour that displays each word, and then its definition was created. Using both the tour and the manual approach seems to duplicated the best of quizzing yourself with flashcards and having someone quiz you with flashcards.

A variant on this application [can] be constructed that displayed several alternative definitions for each word and then gives feedback as to the correct word when each definition is selected. In addition, with this approach, a report file can be generated containing the wrong answers.

16.14.1 Pros

The primary advantage to this application is its ease of use. Not only is it extremely easy to use, but it is also easy to create. In addition, this application can be very useful for students.

16.14.2 Cons

The drawback to this application is that HyperWriter has no evaluation facilities to give the user feedback as to how he or she does.

16.15 Application #15: Electronic Publishing

One area of hypertext that has been slow to take off is that of electronic publishing. That is the publishing of information in a hypertext format. To date, it seems that hypertext has been applied on an in-house basis but not for wholesale distribution of information. Happily, this is currently changing. The Reg-In-A-Box application described under Application #1 and the Culture Document described in Application #4 are both forerunners of the future of publishing. In an electronic publishing environment, documents are constructed designed to be read online. This allows them to take full advantage of technologies such as hypermedia.

The demonstration file for this application is called FCM.HW. It is a prototype of a fictional computer magazine constructed with hypertext links.

16.15.1 Pros

The single biggest advantage of this application is freedom from the restraints of conventional publishing. There are less costs involved and less worry about the physical realities of printing and distribution.

16.15.2 Cons

The disadvantage with this application is simply the lack of copyright protection that the author has over the reader. While it is rare to see someone photocopying an entire book, it is not rare in computer software. To date, this has been an impediment to electronic publishing.

16.16 Application #16: Hypertext Slide Show

Related to the presentation application described previously is this application, a hypertext slide show. Slide shows can be constructed in two fashions: manual or automatic. A manual slide show is sequenced with the rapid activation of links. An automatic slide show is run with a tour.

The file for this application is HTSLIDE.HW. It consists of both a graphic and a text version of a slide show.

16.16.1 Pros

The first advantage to this application is simply ease of use. A second advantage is that with HyperWriter's runtime, slide shows can be freely distributed to those who might benefit from them.

In addition, viewers of slide shows can annotate them to give you feedback -- something that no slideshow product on the market supports.

16.16.2 Cons

The one drawback to this type of computer slide show is HyperWriter's lack of specific transitional screen effects. Some slideshow products can change how the screen is displayed when each segment of the slide show changes.

16.17 Application #17: Image Database of .PCX Files

HyperWriter can be used to construct a database of .PCX images. An image database is very useful as it works to eliminate uncertainty about which graphic images are available on your hard disk.

The file for this application is IMAGEDB.HW. It contains an image database organized along the lines of a directory listing with links for keywords and a 'DISPLAY' link for each graphic image. A useful technique to create this application is using the DOS DIR command to create the file containing the information about each graphics file. To use this technique, go to the directory containing your .PCX files and type the following:

Keystroke	Meaning
DIR *.PCX > PCX.DAT	Do a directory for all files with an extension of .PCX and pipe that directory into a file called PCX.DAT.

When the above commands are given, no directory is displayed onscreen. Instead, that directory is written to an ASCII text file that can then be imported into HyperWriter. In addition, a macro or macros can be constructed that precisely format this text file and build the framework of links needed for this application.

Note: For reasons of size, only a few images, those used in the presentation application, were used in this application.

16.17.1 Pros

One of the advantages to this application is that a single HyperWriter document can hold references to several hundred .PCX files.

16.17.2 Cons

The real disadvantage to this application is the need to update it and build links by hand. Much of this effort can be eliminated, however, using the DIR technique described above. A second disadvantage is HyperWriter's ability to handle only .PCX images. Technically, an image database should handle multiple image formats. Finally, HyperWriter needs all graphics files to be stored in the same directory. One way around this is to construct macros that change graphics directories in HyperWriter.

16.18 Application #18: Text File Management

HyperWriter can be used to construct a framework of links to different text files. In this application, an outline structure of conferences on an online network is duplicated with links to each part of the discussion and the ASCII files that comprise the discussions.

The demonstration file for this application is called BIXTEXT.HW. BIXTEXT is a collection of different conferences downloaded form the Byte Information Exchange or BIX (TM) computer network. These files were downloaded and then linked into this framework of links.

Note: For legal reasons, only a very small sample of text, primarily that posted to BIX by NTERGAID, has been shown here.

16.18.1 Pros

There are two major advantages to this application. First, HyperWriter can browse ASCII files of any size. This means that the ASCII files in this application can grow indefinitely without worry of running out of memory in HyperWriter. A second advantage is HyperWriter's ability to duplicate structures such as BIX's hierarchy. This can be used to construct a very visual approach to text file management -- something that is normally quite obtuse.

16.18.2 Cons

The disadvantage of this application is that while hypertext links form the framework for this application, they can only be created to the text files - not from the text files. This limits some of the hypertext functionality of this document. Although this application does not demonstrate it, there is no reason why links to very specific items in each file could not be constructed. This would regain some of the power of hypertext linking for this application. A second disadvantage of this application is the lack of a powerful search command when viewing the ASCII files.

16.19 Application #19: Computer Based Training

Despite hypermedia's myriad uses, the application of hypermedia to Computer Based Training (CBT) stands out. A hypermedia system such as HyperWriter is an ideal platform for constructing CBT applications. One particular HyperWriter feature applicable to CBT is that of tours.

16.19.1 Pros

One of HyperWriter's chief advantages as a CBT tool is that any application created with HyperWriter can be freely distributed with HyperWriter's runtime. A second advantage is that with HyperWriter's Readers Notes feature, students can take notes on the CBT application if they desire. A final advantage is simply that HyperWriter makes it very easy to get applications up and running quickly.

16.19.2 Cons

As with the courseware example discussed in Section 16.4, the chief disadvantage to this is that HyperWriter has no evaluation facilities. This prevents HyperWriter from offering students evaluations of their progress.

16.20 Additional Applications

While the above application profiles are somewhat exhaustive, it merely scratches the surface of how HyperWriter can be applied. Some additional applications for HyperWriter are:

- Case Analysis

- Interactive Kiosks

- Recipe Management

- Knowledge Work

- Easy-To-Use Front Ends to Simulation Programs

- Executive Information Systems

16.21 Conclusion

Although some of the applications in this chapter may seem a bit extreme, they do demonstrate the versatility and power of hypertext.

These application bear out the point covered in Chapter 14: Nearly anything can be done with hypertext -- it only takes imagination.

16.22 Summary

HyperWriter can be applied to almost any problem involving information management. These applications can range from technical support to replacing paper. The key to applying HyperWriter to these diverse problems is imagination.

Appendix C: List of the Effects that Matter in Hypertext

Phenomenon studied	% Difference
1. Effect of age on moving from just looking at a hypertext system to actually using it: Young (≤20 years) vs. older users relative to on-lookers	115
2. Effect of users' motivation, activity level, etc. on number of new nodes created: Most active participant vs. modal participant on subnet A	100
3. Effect of users' motivation, activity level, etc. on number of new nodes created: Most active participant vs. modal participant on subnet B	63
4. Proportion of effort spent attending to the medium itself rather than on solving task: Revising text on computer vs. with pen and paper	51
5. Subjective preconceptions: Online fiction vs. printed fiction	42
6. Effect of level of expertise on approach to solving task: Use of table of contents on third day vs. first day	40
7. Touch screen activation strategies in Hyperties: Take-off vs. land-on activation, blank space errors	34
8. Effect of user's task on choice of navigation mechanism: Use of guided tour mechanism, exploratory vs. directed task	34
9. Hypertext vs. command-based database access: Spikes (user retracing route through data by backtracking)	32
10. Effect of user's task on choice of navigation mechanism: Use of index, exploratory vs. directed task	30
11. Issue-based argumentation networks built in gIBIS: Links "supporting" position vs. links "objecting to"	24
12. SuperBook vs. printed book: "Discriminating" facts included	23
13. Hypertext vs. command-based database access: Rings (user returning to a previously visited piece of data)	22
14. Effect of screen size: Time to answer questions: 120 lines vs. 22 lines, text editor with program text	21
15. Hypertext vs. command-based database access: Proportion of different nodes visited to total nodes visited	20
16. Effect of fast (3 sec.) vs. slow (11 sec.) response times: time to complete task	20
17. Hierarchical hypertext vs. linear text file: Users' subjective preference	20

(58:245)

Appendix D: Questions Asked MAJCOMS

1. Who is the individual at (MAJCOM) who handles economic analysis (EA)?

2. What is th' major type of EA performed? (i.e. MFH, MCP, MMHS)

3. What reference documents do you recommend using to complete an EA? (regs, guidebooks, base level cost hand book, AFSC cost estimating handbook, army pubs, policy letters, checklists, software programs)

4. Are different references used depending upon the EA type?

5. Who (at base level) performs EAs? (i.e. ACC, CE, LGS, etc)

6. Is preparer's knowledge base (on EAs) different? (basic vs advanced)

7. What are the most common problem areas found in EAs? (discounting, documentation, methodology, sources)

8. Are any of the above reference documents available in computerized form?

Appendix E: Hypertext Systems

System	Address	Phone
Hyperties	Cognetics Corp (55 Princeton-Hightstown Rd, Princeton Junction, NY 08550)	609-799-5005
FileVisionV	Marvelin Corp (3420 Ocean Park Blvd, Suite 3020, Santa Monica, CA 90405)	213-450-6813
Document Examiner	Symbolics, Inc (11 Cambridge Center, Cambridge, MA 02142)	617-621-7500
Text KRS (Knowledge Retrieval System)	Knowledgeset Corp (60 Garden Court, Building A, Monterey, CA 93940)	415-968-9888
Guide	Owl International, Inc (14218 Northeast 21st St, Bellevue, WA 98007)	800-344-9737 or 206-747-3203
HyperCard	Apple Computer, Inc (20525 Mariani Ave, Cupertino, CA 95014)	408-996-1010
KMS (Knowledge Management System)	Scribe Systems, Inc (Commerce Court, Suite 240, 4 Station Square, Pittsburgh, PA 15219)	412-281-5959
Knowledge Pro	Knowledge Garden, Inc (473A Malden Bridge Rd, Nassua, NY 12123)	518-766-3000
HyperPad	Brightbill-Roberts (120 East Washington St, Suite 421, Syracuse, NY 13202)	315-474-3400
Superbook	Bell Communications Research (Bellcore) (435 South St, Morristown, NJ 07962)	201-829-2000 201-740-6110

Appendix F: Questions for Testing and Validating Prototype

Please rate the questions according to the following scale:
1 - strongly agree
2 - agree
3 - neutral
4 - disagree
5 - strongly disagree
N - not applicable

1. I consider myself knowledgeable about general computer operation.
1 2 3 4 5 N

2. I ran the prototype from:
hard disk   floppy disk

3. I had no problems installing the software onto my computer.
1 2 3 4 5 N

4. The opening screen of the prototype was self explanatory.
1 2 3 4 5 N

5. The opening screen instructions allowed me to activate links.
1 2 3 4 5 N

6. I used the reference system help.   yes   no
(if no, skip to question 7)

a. After reviewing the general help I felt confident using the prototype.
1 2 3 4 5 N

b. I reviewed the advanced help screens   yes   no
(If no, skip to question 6d).

c. The advanced help screens were self explanatory.
1 2 3 4 5 N

d. With the online help a users manual is not required.
1 2 3 4 5 N

7. I found the following control panels useful:
a. map function   1 2 3 4 5 N
b. tours function   1 2 3 4 5 N
c. TOC   1 2 3 4 5 N
d. index function   1 2 3 4 5 N
e. note function   1 2 3 4 5 N
f. past history function   1 2 3 4 5 N
g. hypertext help function   1 2 3 4 5 N

8. This system was easy to learn.   1 2 3 4 5 N

9. The system was easy to use. 1 2 3 4 5 N

10. I got lost (i.e., was not sure where I was within the document).
1 2 3 4 5 N

11. The topics presented in the table of contents were appropriate.
1 2 3 4 5 N

12. This system would help me find information more easily.
1 2 3 4 5 N

13. This system would be useful as an office reference system.
1 2 3 4 5 N

14. This system would be useful in economic analysis for:
a. first time learning of concepts 1 2 3 4 5 N
b. browse for general information 1 2 3 4 5 N
c. looking up the answer to a specific question 1 2 3 4 5 N
d. research purposes 1 2 3 4 5 N
e. review of previously familiar material 1 2 3 4 5 N

15. In my normal job, it would have taken me more time to research an equivalent amount of information using any sources (i. e., paper documents, coworkers, etc).
1 2 3 4 5 N

16. For the amount of time spent, I could learn more with this system than using any of the sources above. (i. e., paper documents, coworkers, etc).
1 2 3 4 5 N

17. If I had this system on my desk (or a similar hypertext system in my functional area), I would expect to use it:
a. several times a day
b. once or twice a day
c. once or twice a week
d. once or twice a month
e. less than once a month

18. The features I liked about the system:
(You may select more than one response)
a. displays (use of color, screen organization, contextual info).
b. functions (what things you can do with the system).
c. data links (how you get from one place in the text to another).
d. speed of processing (how fast the computer does things).
e. instructions (on how to use the system).
f. data (the text used in the system).
g. depth and breadth of data available.
h. help (on-line instructions).

19. The following features of the system need to be improved:
(You may select more than one response)
a. displays (use of color, screen organization, contextual info).
b. functions (what things you can do with the system).
c. data links (how you get from one place in the text to another).
d. speed of processing (how fast the computer does things).
e. instructions (on how to use the system).
f. data (the text used in the system).
h. depth and breadth of data available.
g. help (on-line instructions).

20. I used the reference system for:
a. less than 30 minutes.
b. 30 minutes to one hour.
c. one to two hours.
d. more than two hours.

21. If you have any other general comments, please record them on a separate sheet of paper.

(54:56; 71:397-407; 80)

Appendix G: HyperWriter (TM) Technical Specifications

Typical Applications: Interactive, hypermedia and multimedia documents such as Presentations, Computer Based Training Guides, Technical Documentation, Educational Courseware and Electronic Publishing.

Hardware Required:

- Computer: IBM PC, XT, AT or PS/2 or Compatible. 2 Floppy drives minimum, Hard disk recommended, 384k memory, 640k recommended.
- Graphics Card: CGA, HGA, EGA, or VGA cards. Character graphics mode also supported.
- Printer: IBM Proprinter and Epson compatible dot matrix printers, Hewlett Packard Laser Jet. For additional printer support, software tools to create your own printer drivers are included.
- Input: Cursor Keys or Mouse (Microsoft, Logitech or compatible).
- Video Disc Players: Sony LDP-2000 series. Pioneer LDV-4200 series. Tools to create drivers for other Video Disc players are included.
- Digitized Sound: Digitized sound via Covox's Speech Thing and Sound Master is supported. This requires an optional speaker which can be purchased for $29.95. In addition, a 'PLAY' command is included in the script language for sound using the PC's speaker.

Price: $299.95 including documentation.

Abilities: HyperWriter is a general purpose hypermedia authoring tool that can be used to create multimedia documents fusing text, bitmapped graphics, digitized sound, and video disc images. Different media types are tied together using hypertext links.

Links: Hypertext links can be formed both to and from text and graphics. Links can also be created to sound and video disc images. HyperWriter supports three classes of links: Text, Graphic and Action. The subclasses of the text link are Jump (cross reference), SWAP (text replacement), Comment (pop up window) and ASCII (browse external files). The subclasses of the graphic link give control over how images are to be displayed. The subclasses of the Action link are DOS (run DOS programs), Menu (activate menu functions) and Script (write program scripts). The script subclass is used for controlling video disc images, digitized sound and generation of reports. Scripts can also be attached to any link type to allow for execution of an action when the link is activated.

Link Attributes: HyperWriter stores attributes of links such as author, password, date and more.

Document Length and Content: Documents can be as large as available RAM. Also, document files can be linked together on disk giving virtually unlimited storage.

Document Structure: Any document structure can be created in HyperWriter. It places no restrictions on the authoring process. Links can be formed to both new and previously existing materials.

Authoring: HyperWriter includes a full-featured text editor for document creation. Existing files cal also be imported. Link making in HyperWriter is freeform and similar to editing functions such as Cut, Copy and Paste. Links are created by first marking a link anchor, then specifying the type of link and finally entering the material for the destination of the link. A link mimicking feature allows previously created links to be duplicated with minimal overhead. To create documents using video disc images, a video disc scripting language is used.

User Interface: HyperWriter uses a pull down menu interface with dialog boxes. Control, Alt and Function keys are also supported for quicker use. The user interface, both menus and dialog boxes, can be controlled either with the mouse or keyboard. Scroll bars automatically appear if the contents of a window in a document stretch past the window boundary.

Scripting Language: A limited scripting language provides for directly controlling the video disc access, control of sound files and use of a play command (for digital sound effects). Additional scripting commands allow such things as a 'Select Box' which queries a user for the destination of a link.

Background Screens: To support card metaphor hypertext documents similar to HyperCard (TM), multiple backgrounds using both text and graphics are allowed per document.

File Format: HyperWriter uses an internally developed .HW file format. This format is extendable to different media types beyond those supported in the initial release. An integral part of this file format is data compression and encryption to ensure that documents are stored as compactly as possible. Specifications for the file format, as well as a toolkit for using it, will be released in early 1990.

Navigational Tools: Graphic map browser, Global text search, Goto window command, Bookmarks, Collapse document command, Temporary markers, and Dynamic tours of documents.

Compatibility: HyperWriter can load hypertext documents created with our Black Magic hypertext authoring system.

Formatting Tools: Text coloring, Justification, Bold Facing, Italicizing, Margins, Tabs, Text centering, Support for extended ASCII character set, Removal of carriage returns for imported files.

Additional Feature: Macro recorder; Screen Grabber provided for graphics integration; Reporting capabilities based on link activation; Import and Export of ASCII Files.

Document Distribution: Included Public Domain Runtime provides unlimited distribution rights for any documents that you create. The runtime has virtually all of the features of the authoring system except those of altering documents. In addition, the runtime supports annotation of documents.

Supported File Formats: ASCII for text files. .PCX files for bitmapped graphics. Also supported is the .MGR format of the included screen grabber.

(60:1-2)

Appendix H: Prototype Test Results

Questions were rated according to the following scale:

1 - strongly agree	4 - disagree
2 - agree	5 - strongly disagree
3 - neutral	N - not applicable

Testers were divided into two categories based on their answer to question one, inexperienced computer users (ratings 3, 4, 5, and N) and experienced computer users (ratings 1 and 2). The numbers show the number of individuals who gave the indicated rating. The numbers in parenthesis () are percentages of the of the total number per category. Ratings of "not applicable" were not considered (other than question one) in the counts or in the percentages.

1. I consider myself knowledgeable about general computer operation.

1	2	3	4	5	N
26(46)	24(42)	3(5)	2(4)		2(4)

2. I ran the prototype from:

hard disk	54(95)
floppy disk	3(5)

3. I had no problems installing the software onto my computer.

	1	2	3	4	5
Inexperienced User	3(43)	2(29)		1(14)	1(14)
Experienced User	33(66)	14(28)	1(2)	1(2)	1(2)
Total	36(63)	16(28)	1(2)	2(4)	2(4)

4. The opening screen of the prototype was self explanatory.

	1	2	3	4	5
Inexperienced User	2(33)	4(67)			
Experienced User	20(41)	17(35)	6(12)	4(8)	2(4)
Total	22(40)	21(38)	6(11)	4(7)	2(4)

5. The opening screen instructions allowed me to activate links.

	1	2	3	4	5
Inexperienced User	3(43)	4(57)			
Experienced User	19(40)	22(46)	3(6)	2(4)	2(4)
Total	22(40)	26(47)	3(5)	2(4)	2(4)

6. I used the reference system help.

	yes	no
Inexperienced User	7(100)	
Experienced User	41(82)	9(18)
Total	48(84)	9(16)

a. After reviewing the general help I felt confident using the prototype.

	1	2	3	4	5
Inexperienced User			2(29)	5(71)	
Experienced User	8(20)	18(44)	10(24)	4(10)	1(2)
Total	8(17)	18(38)	12(25)	9(19)	1(2)

b. I reviewed the advanced help screens

	yes	no
Inexperienced User	5(71)	2(29)
Experienced User	27(57)	20(43)
Total	32(59)	22(41)

c. The advanced help screens were self explanatory.

	1	2	3	4	5
Inexperienced User			2(40)	3(60)	
Experienced User	9(33)	14(52)	3(11)	1(4)	
Total	9(28)	14(44)	5(16)	4(13)	

d. With the online help a users manual is not required.

	1	2	3	4	5
Inexperienced User		1(14)		3(43)	3(43)
Experienced User	3(10)	9(29)	9(29)	10(32)	
Total	3(8)	10(26)	9(24)	13(34)	3(8)

7. I found the following control panels useful:

	1	2	3	4	5
a. map function					
Inexperienced User	1(14)	1(14)	4(57)	1(14)	
Experienced User	9(20)	18(40)	7(16)	8(18)	3(7)
Total	10(19)	19(37)	11(21)	9(17)	3(6)
b. tours function					
Inexperienced User	1(17)		5(83)		
Experienced User	7(15)	20(43)	10(21)	10(21)	
Total	8(15)	20(38)	15(28)	10(19)	
c. TOC					
Inexperienced User	1(20)	2(40)	2(40)		
Experienced User	14(30)	27(57)	6(13)		
Total	15(292	29(56)	8(15)		
d. index function					
Inexperienced User			4(100)		
Experienced User	6(18)	12(35)	10(29)	5(15)	1(3)
Total	6(16)	12(32)	14(37)	5(13)	1(3)
e. note function					
Inexperienced User	3(50)	3(50)			
Experienced User	10(22)	18(40)	11(24)	6(13)	
Total	13(25)	21(41)	11(22)	6(12)	
f. past history function					
Inexperienced User			5(83)		1(17)
Experienced User	3(8)	9(23)	9(23)	15(38)	3(8)
Total	3(7)	9(20)	14(31)	15(33)	4(9)

g. hypertext help function

Inexperienced User		2(33)	4(67)		
Experienced User	10(23)	22(51)	8(19)	3(7)	
Total	10(20)	24(49)	12(24)	3(6)	

8. This system was easy to learn.

	1	2	3	4	5
Inexperienced User		1(14)	1(14)	5(71)	
Experienced User	13(26)	24(48)	5(10)	7(14)	1(2)
Total	13(23)	25(44)	6(11)	12(21)	1(2)

9. The system was easy to use.

	1	2	3	4	5
Inexperienced User		2(29)	3(43)	2(29)	
Experienced User	12(24)	22(44)	9(18)	6(12)	1(2)
Total	12(21)	24(42)	12(21)	8(14)	1(2)

10. I got lost (i.e., was not sure where I was within the document).

	1	2	3	4	5
Inexperienced User	1(14)	3(43)		3(43)	
Experienced User	4(8)	11(22)	5(10)	20(40)	10(20)
Total	5(9)	14(25)	5(9)	23(40)	10(18)

11. The topics presented in the table of contents were appropriate.

	1	2	3	4	5
Inexperienced User		4(67)	1(17)	1(17)	
Experienced User	8(16)	33(67)	6(12)	2(4)	
Total	8(15)	37(67)	7(13)	3(5)	

12. This system would help me find information more easily.

	1	2	3	4	5
Inexperienced User	1(14)	4(57)	1(14)	1(14)	
Experienced User	16(32)	26(52)	2(4)	5(10)	1(2)
Total	17(30)	30(53)	3(5)	6(11)	1(2)

13. This system would be useful as an office reference system.

	1	2	3	4	5
Inexperienced User	1(17)	2(33)	2(33)		1(17)
Experienced User	18(36)	23(46)	4(8)	3(6)	2(4)
Total	19(34)	25(45)	6(11)	3(5)	3(5)

14. This system would be useful in economic analysis for:

	1	2	3	4	5
a. first time learning of concepts					
Inexperienced User	2(29)	3(43)	2(29)		
Experienced User	9(18)	25(50)	5(10)	8(16)	3(6)
Total	11(19)	28(49)	7(12)	8(14)	3(5)
b. browsing for general information					
Inexperienced User	2(29)	4(57)	1(14)		
Experienced User	14(28)	25(50)	8(16)	3(6)	
Total	16(28)	29(51)	9(16)	3(5)	

c. looking up the answer to a specific question

Inexperienced User	1(17)	3(50)	2(33)		
Experienced User	11(22)	15(31)	17(35)	5(10)	1(2)
Total	12(22)	18(33)	19(35)	5(9)	1(2)

d. research purposes

Inexperienced User		2(33)	3(50)		1(17)
Experienced User	9(18)	18(37)	14(29)	7(14)	1(2)
Total	9(16)	20(36)	17(31)	7(13)	2(4)

e. review of previously familiar material

Inexperienced User		5(83)	1(17)		
Experienced User	17(35)	25(52)	5(10)	1(2)	
Total	17(31)	30(56)	6(11)	1(2)	

15. In my normal job, it would have taken me more time to research an equivalent amount of information using any sources (i. e., paper documents, coworkers, etc).

	1	2	3	4	5
Inexperienced User	1(17)	3(50)		2(33)	
Experienced User	16(32)	22(44)	8(16)	2(4)	2(4)
Total	17(30)	25(45)	8(14)	4(7)	2(4)

16. For the amount of time spent, I could learn more with this system than using any of the sources above. (i. e., paper documents, coworkers, etc).

	1	2	3	4	5
Inexperienced User	1(17)	3(50)		2(33)	
Experienced User	14(29)	20(41)	9(18)	4(8)	2(4)
Total	15(27)	23(42)	9(16)	6(11)	2(4)

17. If I had this system on my desk (or a similar hypertext system in my functional area), I would expect to use it:

	Inexp	Exp	Total
a. several times a day	1(17)	11(22)	12(22)
b. once or twice a day	2(33)	10(20)	12(22)
c. once or twice a week	3(50)	17(35)	20(36)
d. once or twice a month	8(16)	8(15)	
e. less than once a month		3(6)	3(5)

18. The features I liked about the system:

(Note: Percentages for questions 18 and 19 are calculated based upon the fact that each individual could select any number of items. Inexperienced percentages are based on a total possible of 7 and experienced are based on a total of 50)

	Inexp	Exp	Total
a. displays (use of color, screen organization, contextual info).	2(29)	19(38)	21(37)
b. functions (what things you can do with the system).	4(57)	26(52)	30(53)

c. data links (how you get from one place in the text to another).	2(29)	38(76)	40(70)
d. speed of processing (how fast the computer does things).	1(14)	29(58)	30(53)
e. instructions (on how to use the system).		6(12)	6(11)
f. data (the text used in the system).	3(43)	12(24)	15(26)
g. depth and breadth of data available.	2(29)	12(24)	14(25)
h. help (on-line instructions).	1(14)	15(30)	16(28)

19. The following features of the system need to be improved:

	Inexp	Exp	Total
a. displays (use of color, screen organization, contextual info).	3(43)	12(24)	15(26)
b. functions (what things you can do with the system).		8(16)	8(14)
c. data links (how you get from one place in the text to another).	4(57)	6(12)	10(18)
d. speed of processing (how fast the computer does things).	1(14)	3(6)	4(7)
e. instructions (on how to use the system).	5(71)	28(56)	33(58)
f. data (the text used in the system).		7(14)	7(12)
g. depth and breadth of data available.	3(43)	14(28)	17(30)
h. help (on-line instructions).	2(29)	14(28)	16(28)

20. I used the reference system for:

	Inexp	Exp	Total
a. less than 30 minutes.	2(29)	19(39)	21(38)
b. 30 minutes to one hour.	4(57)	28(57)	32(57)
c. one to two hours.	1(14)	2(4)	3(5)
d. more than two hours.			

Appendix I: Instructions to Run Prototype

Use the instructions below to run the prototype.

a. If you are running the system from a hard disk drive:You will be given one disk with a file called INSTALL.EXE. To install the program, make a subdirectory called HR on the C drive and copy the file into it. Install the reference system by typing INSTALL. You need approximately 380K of free disk space. Begin the prototype by typing START.

(Note: If you are using another drive or subdirectory name, the first four lines of the HR.INI file must be modified (after installing) to reflect the correct path. To do this, edit the file with WordPerfect, Ted, EDLIN, etc and save as ASCII format.)

b. If you need to run the system from a dual floppy drive let me know, I'll provide you with two disks to use.

Appendix J: Comments from Prototype Test

1. Installation/Evaluation

a. I'm not sure if I installed my software incorrectly, but the first screen that come up for me was a blank one except for a couple of numbers in the corner. After pressing a few keys I got a menu. I played around in there for a while and read most of the help section. I wasn't sure exactly what was meant by the term link in this software and I couldn't find where it was explained. Finally I chose the right key to bring up your prototype. This helped me to understand what the system was supposed to do (up to that point I had no idea). More instructions telling us what menus to use to bring up the prototype and a basic explanation of the software package would have saved a lot of time.

b. I really didn't understand what it was I was evaluating. Maybe some more in-depth instructions as to what it is this thing is supposed to do and who is supposed to use it, I would be able to evaluate it better.

c. Your letter notwithstanding, it's very difficult to separate the bugs in the development software from your program. Frequent lockups were very irritating. Telling users to ignore these problems does not eliminate their subconscious impact.

2. Instructions on how to use the Prototype

a. Explain the F10 key (menu option) on the cover screen. Need a key template for the menu function keys.

b. A users note would be helpful in some instances since some of the help notes were cryptic at times. Look at other software screens to determine help notes that should be on the screen (e.g., on the opening screen instructions say "Press the Function Key F1 to activate links" and "To return from link press ESC". It would be better to say "F1 Activate Link, ESC Return from Link".)

c. Didn't mention mouse capability. It was not hard to figure out how to run the program with a mouse, but some instructions would help. The mouse implementation is probably the biggest benefit of hypertext. I figured out that on a 3 button mouse the left key = F1 and the right key = ESC. I found the system much simpler to use with a trackball. Through some experimentation, I discovered that if I clicked on the top of the screen the F10 menu would appear. Movement was much quicker with my trackball. The only time I was unable to use it was when typing was required.

d. A manual would be a much better way to present this material even if problems with tutorial help are fixed. Definitely need manual. Need to define what hypertext is. The terms "link" and "icon" as used were unfamiliar. The meaning becomes obvious after going through a few windows, however.

e. Liked instructions on how to use system -- system easy enough to use so that much instruction isn't necessary, but it is nice that it is available.

f. Control panel at top of each window uses nonstandard applications of standard commands (exit, control panel help).

g. No way to exit the system from cover screen.

3. Display

a. Monochrome monitor: A monochrome monitor was distracting because the background is amber with black blocks surrounding the letters. Does it need to be used with color? I have a mono-color monitor and I found the darkened spaces between words distracting. Is there anyway to make this background one color vice the blank spaces? I had difficulty with the appearance of the text on my screen. The text was difficult to read because each individual word was reversed highlighted from the background. I worked the system on CGA and changed the colors to get a black background with white text to make it easier to read.

b. Color monitor: Could have been just me, but even though the background color (blue) and the white text looked nice, I found that when there was a fairly full page of material the text seemed crowded and harder to read.

c. The tours screens changed too quickly for me to read them. Tours function was too slow.

d. Displays should have more fonts.

4. Depth/Breadth of System

a. The reference system will not, I don't think, answer questions about what's inside the regulations.

b. A very useful extension would be to have the actual documents accessible with a means to capture pertinent parts of them.

c. Data and depth and breadth of data in system is ok for cost analysis refresher, but not for an instructor (i.e., does not provide formulas to be used in cost analysis, etc). As an introduction to cost analysis, your program as it stands is totally acceptable, but could be better if it provided more detail. More links also need to exist. Links needs to exist between terms and their definitions. This is a good tool as is for people with experience who want to quickly brush up on some concepts of cost analysis; for people new to cost analysis there needs to be more definitions; and for people to perform an analysis using only this tool there needs to be a little more detail.

d. Overall, I thought the system worked very well and could be of benefit. It needs to be expanded, which does not seem to be a small task. Also, it may be difficult to keep the regulations up to date if you intend to include them.

e. I think the system has merit. You might want to break the EAs up by majcom. Running a search through the entire USAF database could take a while (provided the prototype had only a small sample in it. I know I would make good use of such a system if I were a base level analyst.

5. Applications/Use

a. This is my first exposure to the concept of hypertext with its range of capabilities. The concept is sound and as you've applied it, it obviously has strong applicability to the military.

b. I think this system could be used effective for many reference applications. Putting the AFR 0-1 is a very likely application, along with adapting the system to a specific office that requires a great deal of reg references like an orderly room or CBPO.

c. This could be an excellent teaching aid as well.

d. It's good for tasks one doesn't perform on a regular basis, or those tasks that are complex and require detailed steps. I could also see a use for this as a training aide.

e. Would use the system several times a day when new to the program but only once or twice a month after becoming familiar with my job.

f. Overall, the system would be very useful! Can you read ASCII text directly in and create the links after the text is there? How many levels can you imbed the text to? Can you search for topics? How much does it cost? Have the "glitches" been fixed yet? Who makes the software?

6. HyperWriter system unique comments

a. Remembering to press the F1 instead of enter took some getting used to.

b. Past history function could be expanded.

c. Tabbing beyond last selection should loop back to first selection.

d. Data links would be great if you could activate them with the first letter of the word. F1, arrow, tab, and esc movement were good but enabling the link with key letters would be nice too.

e. Couldn't get out of link map. Pushing F1 to activate link caused the computer to lock up and I had to reboot. There were several times when the system locked up on me.

f. Map is hard to read but is the heart of the system. . . Come up with a better map! Why is the map (flowchart) sideways anyhow? What about scrolling [back and forth as opposed to up and down]? Map function does not serve much use for the end user.

g. I found it annoying that the screen had to be completely redrawn every time you selected an item.

Bibliography

1. Akscyn, Robert M. and others. "KMS: A Distributed Hypermedia System for Managing Knowledge in Organizations," Communications of the ACM, 31: 820-835 (July 1988).

2. ----- and others. "The Data Model is the Heart of Interface Design," Proceedings of CHI '88, Human Factors in Computing Systems Conference. 115-120. New York: Association for Computing Machinery, 1988.

3. Ambron, Sueann and Kristina Hooper. Interactive Multimedia - Visions of Multimedia for Developers, Educators, and Information Providers. Foreword by John Sculley. Redmond WA: Microsoft Press, 1988.

4. Beck, Peter M., Department Staff Analyst. Telephone interviews. The Analytic Sciences Corporation, Arlington VA, April through May 1990.

5. Beeman, William O. and others. "Hypertext and Pluralism: From Lineal to Non-lineal Thinking," Proceedings of Hypertext '87. 67-88. New York: Association for Computing Machinery, 1987.

6. Begeman, Michael L. and Jeff Conklin. "The Right Tool for the Job," Byte, 13: 255-266 (October 1988).

7. Bernstein, Mark. "The Bookmark and the Compass: Orientation Tools for Hypertext Users," ACM SIGOIS Bulletin, 9: 34-45 (October 1988).

8. Bielawski, Larry and Robert Lewand. Expert Systems Development, Building PC-Based Applications. Wellesley MA: QED Information Sciences, Inc., 1988.

9. Bixby, Randy L. "Hypertext in a Resource Sharing Greater Machine Level Environment: Hypertext on the VAX," Hypermedia Laboratory, Defense Applied Information Technology Center; Review for 1988, December 1988. Defense Applied Information Technology Center, Hypermedia Laboratory (AD-A209318).

10. -----, Technical Information Specialist, Office of Information Systems and Technology. Telephone interview. Defense Technical Information Center, Alexandria VA, 23 May 1990.

11. Boar, Bernard H. Application Prototyping. New York: John Wiley & Sons, Inc., 1984.

12. Boehm, Barry W. Software Engineering Economics. Englewood Cliffs NJ: Prentice-Hall, Inc., 1981.

13. Boone, MSgt David A., Economic Analysis Program Manager, Directorate of Cost Analysis. Telephone interview. HQ MAC, Scott AFB IL, 27 March 1990.

14. Bush, Vannevar. "As We May Think," Atlantic Monthly, 1: 101-108 (July 1945).

15. Carlson, Patricia Ann. "Hypertext and Intelligent Interfaces for Text Retrieval," The Society of Text: Hypertext, Hypermedia, and the Social Construction of Information, edited by Edward Barrett, Cambridge MA: The MIT Press, 1989.

16. Carney, John, Cost Analyst. Telephone interview; Air Force Cost Center, Crystal City DC, 30 May 1990.

17. Charney, Davida. "Comprehending Non-Linear Text: The Role of Discourse Cues and Reading Strategies," Proceedings of Hypertext '87. 109-120. New York: Association for Computing Machinery, 1987.

18. Conklin, Jeff. A Survey of Hypertext. Microelectronics and Computer Corporation (MCC) Technical Report, Number STP-356-86, Rev. 2. Austin TX, December 3, 1987.

19. Crane, Gregory, "From the Old to the New: Integrating Hypertext into Traditional Scholarship," Proceedings of Hypertext '87. 51-55. New York: Association for Computing Machinery, 1987.

20. Croft, W. Bruce and Howard Turtle. "A Retrieval Model for Incorporating Hypertext Links," Proceedings of Hypertext '89. 213-224. New York: Association of Computing Machinery, Inc., 1989.

21. Delisle, Norman M. and Mayer D. Schwartz. "Contexts--A Partitioning Concept for Hypertext," ACM Transactions on Office Information Systems, 5: 168-186 (April 1987).

22. Department of the Air Force. "Air Force Cost Center: Cost Bulletin Board." Electronic download, Bulletin 2, AF Cost Center Projects, numbers 53, 54, and 84, 4 April 1990.

23. -----. Cost Analysis: Economic Analysis and Program Evaluation for Resource Management. AFR 173-15. Washington: HQ USAF, 4 March 1988.

24. -----. "Cost Training and Professional Development," unpublished report, prepared by HQ AFCC/ACC, 14 August 1987.

25. -----. The AFSC Cost Estimating Handbook Series, Volume I "AFSC Cost Estimating Handbook". Prepared for AFSC by The Analytic Sciences Corporation, undated.

26. -----. The Base Level Cost Analysis Handbook. Prepared for Deputy Comptroller, Cost and Economics (SAF/ACCE) by The Analytic Sciences Corporation, 1988.

27. Duffy, Thomas M. and others. "The Evaluation of Online Help Systems: A Conceptual Model," The Society of Text: Hypertext, Hypermedia, and the Social Construction of Information, edited by Edward Barrett, Cambridge MA: The MIT Press, 1989.

28. Economic Analysis Checklist. Prepared by HQ MAC/ACCC, Scott AFB IL, undated.

29. Fairley, Richard E. Software Engineering Concepts. New York: McGraw-Hill Book Company, 1985.

30. Feltovich, Paul J. amd Richard L. Coulson. Acquisition, Understanding, and Application of Biomedical Science Knowledge: Final Report, 1 May 87 - 30 April 88. Grant No. N00014-87-G-0165. Southern Illinois University School of Medicine, December 1988 (AD-A204712).

31. Fiderio, Janet. "A Grand Vision," Byte, 13: 237-244 (October 1988).

32. Fischer, Gerhard. "JANUS: Integrating Hypertext with a Knowledge-based Design Environment," Proceedings of Hypertext '89. 105-117. New York: Association of Computing Machinery, Inc., 1989.

33. Florian, Daniel J. Use of Hypermedia as a User Interface for an Artificial Intelligence-Based Problem Solver, MS thesis, AFIT/GCS/ENG/89D-3. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1989 (AD-A215551).

34. Frisse, Mark E. "Searching for Information in a Hypertext Medical Handbook," Communications of the ACM, 31: 880-886 (July 1988).

35. Gaulding, Jill and Boris Katz. "Using "Word-Knowledge" Reasoning for Question Answering," The Society of Text: Hypertext, Hypermedia, and the Social Construction of Information, edited by Edward Barrett, Cambridge MA: The MIT Press, 1989.

36. Glushko, Robert J. "Design Issues for Multi-Document Hypertexts," Proceedings of Hypertext '89. 51-60. New York: Association of Computing Machinery, Inc., 1989.

37. Granda, Richard E. and others. "The Perceived Usefulness of Computer Information Sources: A Field Study," SIGCHI Bulletin, 21: 35-43 (April 1990).

38. Grice, Roger A. "Online Information: What Do People Want? What Do People Need?" The Society of Text: Hypertext, Hypermedia, and the Social Construction of Information, edited by Edward Barrett, Cambridge MA: The MIT Press, 1989.

39. Halasz, Frank G. "Reflections on Notecards: Seven Issues for the Next Generation of Hypermedia Systems," Communications of the ACM, 31: 836-852 (July 1988).

40. Hayden, MSgt Michael T, NCOIC, Field Support, Directorate of Cost Analysis. Telephone interview. HQ TAC, Langley AFB VA, 29 March 1990.

41. Herrstrom, David S. and David G. Massey. "Hypertext in Context," The Society of Text: Hypertext, Hypermedia, and the Social Construction of Information, edited by Edward Barrett, Cambridge MA: The MIT Press, 1989.

42. "Hypermedia and Hypertext - Linking Up," Hypermedia Lab, Program Note 5, October 1988. Hypermedia Laboratory, Defense Applied Information Technology Center; Review for 1988, December 1988. Defense Applied Information Technology Center, Hypermedia Laboratory (AD-A209318).

43. HyperWriter Software. Version 2.0. Fairfield CT: NTERGAID, 1990.

44. Jones, III, Henry W. "Developing and Distributing Hypertext Tools: Legal Inputs and Parameters," Proceedings of Hypertext '87. 367-374. New York: Association for Computing Machinery, 1987.

45. Kern, Richard R, Instructor. Personal interviews. AFIT, Dayton OH, 16 May through 20 July 1990.

46. Kinnell, Susan K. "Information Retrieval in the Humanities Using Hypertext," Online, 12: 32-40 (March 1988).

47. Koved, L. and B. Shneiderman. "Embedded Menus: Selecting Items on Context," Communications of The ACM, 29: 312-318, (Apr 1986).

48. Kuhn, Allan D. Hypermedia Laboratory, Defense Applied Information Technology Center; Review for 1988, December 1988. Defense Applied Information Technology Center, Hypermedia Laboratory (AD-A209318).

49. -----. Hypermedia: Our Entry into the Interwingling Zone, August 1989. Defense Applied Information Technology Center (AD-A212604).

50. -----. The DOD Gateway Information System (DGIS): The Development Toward Artifical Intelligence and Hypermedia in Common Command Language, December 1988. Defense Applied Information Technology Center, Hypermedia Laboratory (AD-A203674).

51. -----. "Toward an Artificial Intelligence Environment for Defense Technical Information Center (DTIC): Proposed Tasks - 1987," Hypermedia Laboratory, Defense Applied Information Technology Center; Review for 1988, December 1988. Defense Applied Information Technology Center, Hypermedia Laboratory (AD-A209318).

52. Landow, George P. "Relationally Encoded Links and the Rhetoric of Hypertext," Proceedings of Hypertext '87. 331-343. New York: Association for Computing Machinery.

53. Ledgard, Henry. Software Engineering Concepts. Reading, Massachusetts: Addison-Wesley Publishing Company, 1987.

54. Liddle, Jeffrey M. An Expert System for Learning the National Electric Code, MS thesis, AFIT/GEM/LSM/89S-12. School of Systems and Logistics, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, September 1989 (AD-A216131).

55. McClelland, Bruce. "Hypertext and Online . . . A Lot That's Familiar," Online, 13: 20-25 (January 1989).

56. McCracken, Donald L. and Robert M. Akscyn. "Experiences with the ZOG human-computer interface system," International Journal of Man-Machine Studies, 21: 293-310 (October 1984).

57. Nielsen, Jakob. "The Art of Navigating through Hypertext," Communications of the ACM, 33: 296-310, (March 1990).

58. -----. "The Matters that Really Matter for Hypertext Usability," Proceedings of Hypertext '89. 239-248. New York: Association of Computing Machinery, Inc., 1989.

59. -----. "Trip Report: Hypertext '87 Chapel Hill, North Carolina, 13-15 November 1987," ACM SIGCHI Bulletin, 19: 27-35 (April 1988).

60. NTERGAID Inc. HyperWriter Advertisement. Fairfield CT: NTERGAID, Inc., 1990

61. -----. HyperWriter!, HyperWriter software documentation manual, Fairfield CT: NTERGAID Inc., 1990.

62. Oborne, David J. and Doreen Holton. "Reading from Screen versus Paper: There is no Difference," International Journal of Man-Machine Studies, 28: 1-9, (January 1988).

63. OED on CD-ROM, Oxford University Press, Byte, 15: 292 (January 1990).

64. QMT 345 Economic Anlysis for Cost and Pricing. Written by Lcdr Gary R. Kaufman, AFIT, March 1990.

65. Raymond, Darrell R. and Frank Wm. Tompa. "Hypertext and the Oxford English Dictionary," Communications of the ACM, 31: 871-879 (July 1988).

66. ----- and others. "Measuring the Effectiveness of Personal Database Structures," International Journal of Man-Machine Studies, 31: 237-256, (1989).

67. Rhoads, R.W. "Management Implications for Hypertext Projects," Proceedings of the Eight Annual Conference on Technology and Innovations in Training and Education (TITE '90). 8-19. 1990.

68. Robertson, G. and others. "The ZOG approach to man-machine communication," International Journal of Man-Machine Interface, 14: 461-488 (May 1981).

69. Saffo, Paul. "What You Need to Know About Hypertext," Personal Computing, 11: 166-173 (December 1987).

70. Sculley, John. "The Relationship Between Business and Higher Education: a Perspective on the 21st Century," Communications of the ACM, 32: 1056-1061 (September 1989).

71. Shneiderman, Ben. Designing the User Interface: Strategies for Effective Human-Computer Interaction. Reading MA: Addison-Wesley Publishing Company, 1987.

72. -----. "Reflections on Authoring, Editing, and Managing Hypertext," The Society of Text: Hypertext, Hypermedia, and the Social Construction of Information, edited by Edward Barrett, Cambridge MA: The MIT Press, 1989.

73. -----. "User Interface Design for the Hyperties Electronic Encyclopedia," Proceedings of Hypertext '87. 189-194. New York: Association for Computing Machinery, 1987.

74. -----. Hypertext Hands-On!: An Introduction to a New Way of Organizing and Accessing Information. Reading MA: Addison-Wesley Publishing Company, 1989.

75. Smith John B. and others. "A Hypertext Writing Environment and its Cognitive Basis," Proceedings of Hypertext '87. 195-214. New York: Association for Computing Machinery, 1987.

76. Smith, Karen E. "Hypertext--Linking to the Future," Online, 12: 32-40 (March 1988).

77. Stamps, Capt Mark, Cost Officer Course Instructor. Telephone interview. Sheppard AFB TX, 30 March 1990.

78. Taylor, Dianne D, Cost Analyst, Directorate of Cost Systems Division. Telephone interview. HQ SAC, Offutt AFB NE, 28 March 1990.

79. Thronesbery, C.G., and R.W. Rhoads. "Application of Hypertext to Army Field Manuals," Proceedings of the Eight Annual Conference on Technology and Innovations in Training and Education (TITE '90). 568-589. 1990.

80. -----. Personal correspondence. LB&M Associates, Lawton OK, 21 June 1990.

81. Trigg, Randall H. "Guided Tours and Tabletops: Tools for Communicating in a Hypertext Environment," ACM Transactions on Office Information Systems, 6: 398-414 (October 1988).

82. Van Dam, Andries. "Hypertext '87 Keynote Address," Communications of the ACM, 30: 887-895 (July 1988).

83. Weyer, Stephen A. and Alan H Borning. "A Prototype Electronic Encyclopedia," ACM Transactions on Office Information Systems, 3: 63-88 (January 1985).

84. Yankelovich, Nicole and others. "Creating Hypermedia Materials for English Literature Students," SIGCUE Outlook, 19: 12-25 (Spring/Summer 1987).

Vita

DeAnna L. McMurry (Weelborg) [redacted] was an honor graduate from Saint Mary's High School in Dell Rapids, South Dakota, in 1977 and then attended South Dakota State University. She earned a Bachelor of Science in Agricultural and Biological Science in December 1981 and received her commission as a Distinguished Graduate of the Air Force Reserve Officer Training Corps. She graduated as Distinguished Graduate from the Cost Analysis Officer Course and was assigned as a cost analyst to the 1st Space Wing at Peterson AFB, Colorado. In 1986 she was assigned to Scott AFB, Illinois where she worked as Chief Cost Branch, 375 Comptroller Squadron and as Cost Analyst, Headquarters Air Force Communications Command until entering the School of Systems and Logistics, Air Force Institute of Technology, in May 1989.

[redacted]

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE September 1990	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE

USE OF HYPERTEXT FOR THE DEVELOPMENT OF AN OFFICE REFERENCE SYSTEM ON ECONOMIC ANALYSIS

5. FUNDING NUMBERS

6. AUTHOR(S)

DeAnna L. McMurry, Captain, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology, WPAFB OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GCA/LSQ/90S-9

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The objective of this research was to evaluate the use of hypertext technology for an automated office reference system. A secondary purpose, because of the relative newness of hypertext technology, was to provide a single reference document which describes hypertext and the steps needed in managing or developing a hypertext product. Individuals unfamiliar with hypertext technology can learn how to apply it in an office environment. The evaluation consisted of two phases, a literature review to discover the hypertext supports for a reference system and a prototype development and test of an Economic Analysis reference system. The literature review includes a review of hypertext features, advantages and disadvantages, applications (including government use), factors affecting usability, and legal concerns. The prototype development answered questions in three areas: economic analysis information required, hypertext development concerns, and applicability of hypertext to the office environment. The developed prototype, tested using a questionnaire answered by 60 graduate students, showed that hypertext has strong applicability for an office reference system. The results showed the reference system was easy to learn and use, would provide quicker and easier access to information, and users would learn more than through the current paper reference system.

14. SUBJECT TERMS

automation, computer applications, economic analysis, hypertext

15. NUMBER OF PAGES
124

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL