HYPERTEXT – Moving Towards Large Volumes

I. RITCHIE
Office Workstations Limited, Rosebank House, 144 Broughton Road, Edinburgh EH7 4LE and *OWL International Inc., 2800 156th Avenue SE, Bellevue, Washington 98007, USA*

Hypertext is an approach to information management in which documentation is displayed as a network of nodes connected by links. Such nodes can contain text, graphics, audio, video or can link to other software or data.

The result is a very powerful publishing medium. It is now possible to provide large volumes of information in the form of interactive documents and to make it accessible to a wide population of users. As the volume of data grows, however, the task of authoring and reading such documents becomes much more complex.

The paper reviews hypertext technology, both in the research laboratory and in commercial application. It then examines the types of software tools which are required to manage, maintain and diagnose faults in large-volume Hypertext systems, based on experience with GUIDE hypertext technology at Office Workstations Limited, which has led to a volume document management system called IDEX.

Received June 1989

1. ORIGINS OF HYPERTEXT

The term hypertext is now widely used to describe systems which allow highly interactive electronic documents to be created and published. The distinguishing feature of hypertext documents is that they are non-linear; they allow 'links' between parts of the documentation for purposes such as explanation and expansion, comment or criticism.[20,7] Such document structures are not new, established examples include the following:

Non-linear reference books such as encyclopaedias, dictionaries and thesauri have been common for centuries.

Medical textbooks such as *Gray's Anatomy* (first published in 1858) rely heavily on cross-references including links between text and diagrams, text and text, and diagrams and text.

Refereed papers in technical journals depend on citations of evidence, normally provided by cross-references to other published technical papers.

Instead of being required by the physical constraints of paper to present such information only in a sequential fashion, leaving the reader with the task of physically finding and obtaining separate sections or volumes, the hypertext system allows authors to create electronic paths through related material, to cross-reference other documents, to annotate text and to create nodes. The user of such a hypertext system can quickly follow references and footnotes, see figures and charts, without losing their original context (Fig. 1).

1.1 Vannevar Bush and the Memex

The mechanisation of this type of information was a specific interest of Vannevar Bush, who in 1941 was appointed the first Director of the United States Office of Scientific Research and Development. In this role he was responsible for some 6000 scientists engaged on defence research during the Second World War. In 1945 Bush wrote an article, called *As we may Think*, which was published in the magazine *Atlantic Monthly*[6] in which he argued that: 'Our methods of transmitting and reviewing the results of research are generations old and by now are totally inadequate in their purposes'. Bush proposed a machine called 'the Memex', which he described as 'a device in which an individual stores his books, records and communications, and which is mechanised so that it may be consulted with exceeding speed and flexibility'.

Bush referred to Leibnitz and to Babbage, both of whom designed sophisticated calculating machines which could not be constructed during their lifetimes, and he went on to envisage his Memex as a combination of two (at that time) emerging technologies – microphotography and electrical logic. Bush then turned his attention to the methods that a memex would automate.

> Our ineptitude in getting at the record is largely caused by the artificiality of systems. When data of any sort are placed in storage, they are filed alphabetically or numerically, and information is found (when it is) by tracing it down from subclass to subclass. It can only be in one place, unless duplicates are used; one has to have rules as to which path will locate it, and the rules are cumbersome. Having found one item, moreover, one has to emerge from the system and re-enter on a new path.
>
> The Human mind does not work that way. It 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.

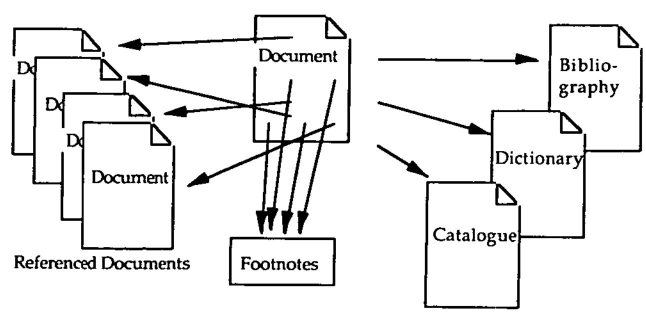

Figure 1. Structure of a hypertext system. Links from a document to its footnotes, to other referenced documents, and to backup documentation are automated for rapid access by the reader.

He then envisaged the mechanisation of this process where the user will add his own 'trails' to the material on his Memex.

When the user is building a trail, he names it ... Before him are the two items to be joined, projected onto adjacent viewing positions ... The user taps a single key, and the items are permanently joined....

Thereafter, at any time, when one of these items is in view, the other can be instantly recalled merely by tapping a button.

Bush's foresight from the 1940s makes dramatic reading in the 1980s when so much of his prediction is being delivered. It is ironic therefore that, like Leibnitz and Babbage before him, Bush did not live to see the machine he had defined.

1.2 Doug Engelbart and Augment

The most effective work in developing and implementing the concepts described by Bush during the 1950s and 1960s was led at the Stanford Research Institute (SRI) by Doug Engelbart. The NLS (oN Line System), developed by the Augmented Human Intellect Research Centre at SRI, allowed users to create electronic documents based on connected concepts, to build hierarchies of information and to collaborate with others on the joint development of documentation. This work was also responsible for the development of many of the features which we now recognise as standard in modern personal computer systems: notably text processing and electronic mail, multi-document screen displays and interactive control by the user of a computer system. In a paper in 1963,[10] Engelbart introduced, among other things, the pointing device that later became the first computer 'mouse':

The hypothetical writing machine thus permits you to use a new process of composing text. For instance, trial drafts can rapidly be composed from rearranged excerpts or old drafts, together with new words or passages which you insert by hand typing.

At the Fall Joint Computer Conference in San Francisco in December 1968 Engelbart demonstrated NLS for the first time in public. The system was primarily hierarchical, reflecting the structure of most technical documentation, but it also allowed reference links to be established between levels and between files. NLS could also be used by more than one user working jointly on the same document; each user operated a separate mouse pointing device and could pass the pointer, or 'electronic chalk' from one to the other to indicate points of discussion.[11]

NLS was commercialised by McDonnell-Douglas as the *Augment* system but the hardware required to operate the system was too expensive for normal commercial use and it did not succeed as a product.

1.3 Ted Nelson, Hypertext and Xanadu

The originator of the term Hypertext was Ted Nelson who coined it in 1963 and published it in 1965. This definition is from his book: *Dream Machines*:[15]

By 'hypertext' I mean non-sequential writing.

Ordinary writing is sequential for two reasons. First, it grew out of speech ... which (has) to be sequential, and second, because books are not convenient to read except in a sequence.

But the structures of ideas are not sequential. They tie together every which way. And when we write we are always trying to tie things together in non-sequential ways.

Nelson's hypertext project is called Xanadu; and is surely the most ambitious hypertext project of all. Xanadu is concerned with the construction of a hypertext server which will allow all the world's literature to be linked up and made available to the user of a computer terminal.[16] The access routes, display mechanisms, and protocols are to be provided by other vendors (the 'front-ends') with Xanadu providing the 'back-end' functions.[17] After more than 20 years as a loosely-organised cooperative the Xanadu project is now a commercial project, being developed by Autodesk Inc.

2. HYPERTEXT RESEARCH PROJECTS

A number of research projects have significantly advanced the technology of hypertext during the last two decades. Notable examples have included the following:

Brown University's Intermedia, which is the direct descendant of an early hypertext project called FRESS by Ted Nelson and Andy Van Dam at Brown University during the late 1960s and early 1970s.[20] The Institute for Research in Information and Scholarship (IRIS) at Brown University has developed hypertext systems for a variety of courses, including for the department of English Literature.[21]

Intermedia documents are rich in display features, showing both text and graphics together in scrollable document windows.

A notable feature of Intermedia is the 'web' which displays a set of links of a hypertext document and can be used in order to provide a model of the information paths, similar to Engelbart's trails, through the documents. Typically a tutor will set up recommended paths in the hypertext course material and the student will add his own links and annotation to this.[22] Although Intermedia runs on Apple Macintosh environments it is not a commercial product and has so far only been used outside of Brown University in limited circumstances.

Carnegie-Mellon's ZOG was a research project for most of the 1970s and in the early 1980s was applied to an information-management project on the USS *Carl Vinson*, the largest aircraft carrier in the world. The system is frame based, filling most of the screen of a graphics workstation with combinations of text and graphics; links to other documents are indicated by a small circle icon at the start of a section of text; dynamic feedback is provided by a changing cursor pattern which indicates the functions available at each link in the system. A commercial product, KMS, was derived from this work and is marketed by Knowledge Systems Inc. to run on SUN and other workstations.[1]

Xerox PARC's Notecards is a system developed on powerful Xerox Lisp machines at the Palo Alto Research Centre (PARC) of the Xerox Corporation.[12] NoteCards provides an environment in which the electronic equivalent of 3″ × 5″ note cards can be created to contain both text and graphics and hypertext links can be created between the individual cards. A typical NoteCards display will show several cards on the screen at once. NoteCards also provides two generic navigation card types: the Browser Card which is generated by the system and shows a map of the links between a set of cards, and the Filebox Card which can be used to categorise and collect related cards. NoteCards is not a commercially available product but it has been quite widely used for research and evaluation projects outside Xerox PARC.

University of Kent's GUIDE was begun in 1982 by Peter Brown at the Computing Laboratory of the University of Kent at Canterbury.[3,4] The goals of the project were to develop electronic documents which users would prefer to paper ones.

GUIDE set out not to imitate pieces of paper or card, but instead to offer the user an active electronic model of a document, allowing the opportunity to interact directly with it in order to navigate and browse through the information. The basic structure of GUIDE is the 'button', a section which is replaced when selected, normally with an expansion to provide additional levels of detail. This mechanism allows the user (whether author or reader) to expand and contract a document, viewing the desired level of detail at any time.

GUIDE was further developed by Office Workstations Limited with the addition of complex graphics, multiple font selection and, most significantly, reference link buttons. Reference buttons allow the reader to quickly navigate around sections of a document and to automatically call up other, related, documents.

The resulting product for the Apple Macintosh and the IBM Personal Computer was first marketed by its US subsidiary, OWL International Inc. in August 1986.[13,19] This product, which can display multiple scrollable documents containing high-quality text and graphics has now become very widely used for hypertext application development. The University of Kent has also continued development of its own UNIX-based program.[5] GUIDE was awarded the BCS Award for Technology, 1988.

3. COMMERCIAL HYPERTEXT

Adoption of hypertext technology outside of the research laboratory has been quite limited until relatively recently. In the last two years, however, there has been an explosive growth in commercial hypertext projects. Some of the reasons for this have been as follows:

Graphics-oriented office workstations. During the mid-1980s the display on a typical personal computer has evolved from character-array screens to pixel-array screens. These displays can present information much more flexibly than character screens, with a variety of typestyles and high quality graphics capabilities. The result of this is that documents can be displayed on a relatively low cost machine with a high quality of appearance. At the same time the interactivity of the personal computer has been improved with the widespread acceptance of the 'mouse' pointing device (almost exactly as first developed by Engelbart in the 1960s) and which is now utilised by most newly released software for the selection of commands and for manipulation of text and graphics images.

While not strictly essential, both of these features at a low cost are critical to the acceptance of hypertext document management systems. Unlike Computer-Aided Design, where the expense of a powerful workstation can be justified for a relatively few highly qualified users, the successful hypertext system must be made generally available to a wide population of users and so the cost and features of a station is a significant factor.

High-capacity storage and transmission. The development of data networks to interconnect computers both in one location and between different locations has created the capability of rapidly publishing information to a wide population of users.

The development of higher capacity storage devices, both magnetic and optical, has reduced the cost of digital storage dramatically. A CD-ROM optical disk, which can be manufactured in volume for about $2, can be used to carry the contents of some 300,000 pages of documentation. Both of these factors provide a fertile environment for the application of hypertext document delivery systems.

Powerful software products. Products with hypertext capabilities have been becoming commercially available. The GUIDE product, for both the Apple Macintosh and the Personal Computer (running *Microsoft WINDOWS*), has been very widely used in corporate and education hypertext projects since its launch in 1986. Other commercial hypertext products include KRS from Knowledgeset in Monterey, California, Hyperdoc from a French-based company, GECI International and HyperTies from Cognetics Corporation in Maryland.

The most dramatic product arrival in this field however is undoubtedly HyperCard from Apple Computer Inc., which was launched in August 1987. Although not specifically a hypertext product (it is primarily a powerful application-generation environment), it can be used to create very attractive dynamic documentation with hypertext links based around 'stacks' of 3″ × 5″ electronic 'cards'. Apple Computer have effectively placed this product in the public domain by including a free copy of HyperCard with every new Macintosh sold and by selling it to existing customers at a very low price. The result has been to create a wide population of users, in both education and in corporate applications, who are developing hypertext documents.

It is clear that these are only the first generation of hypertext products and that many others will be launched in the coming few years.

One of the most active areas of development of hypertext will be to build in the capabilities to manage other information sources including Video and Audio material; the resulting technology has already been called Hypermedia.

It is interesting to note that Bush's Memex included provision for voice storage and recovery and that Engelbart's NLS project experimented with integrated video images.

Both the GUIDE and HyperCard products are now being used actively to create applications which will run Video and Audio sequences and can call on other software packages for functions such as animation, database access and expert system diagnosis.

4. HUMAN FACTORS IN HYPERTEXT

It seems obvious that if documentation is difficult to read then there is likely to be very few readers; unfortunately this is not always the case. In many situations there is simply no choice in the access route to information and so the user has to tolerate quite appalling user interfaces as a necessary evil. Often the importance of high-quality displays are ignored, factors such as the refresh rate of a screen can critically affect even simple tasks such as proof reading.[2]

Modern dynamic document environments such as GUIDE and HyperCard allow a great deal of scope to the author to present information in an attractive and well designed way, but in many cases this great flexibility, if poorly used, can lead to confusing and badly structured documents, It is very easy for a thoughtless author to create labyrinthine documents which lead the user through to a succession of poorly labelled 'dead-end' trails. Hypertext documents are very different from paper documents and it is not realistic to expect authors to quickly master the art of hypertext authoring. Ironically it is often the desire to imitate paper that results in poor hypertext.

Some of the factors which distinguish the user acceptability of hypertext are as follows.

Appropriate use of hypertext attributes. It is important to distinguish between documentation factors which are attributes of the structure of a document as opposed to factors which are attributes of the way paper documentation is constructed. The first category includes section headings, footnotes, cross references, etc., which are often directly translatable into their hypertext analogies; the second category includes multiple column layout, arbitrary page splits etc. which are only have function on the page. The ability to interact with the computer is what distinguishes hypertext, and intelligent use of this characteristic can generate documentation which is more functional than the paper equivalent.

Quality of image and design. Issues of image quality and design are not purely aesthetic: they are fundamental to the user acceptance and comprehension of the information being imparted.

Document based hypertext systems such as Intermedia and GUIDE place as few restrictions as possible on the appearance and structure of documents. The HyperCard product contains its own graphics editing environment; ensuring that the appearance of documents created by it can be as visually attractive as the author is capable of.

Navigation. One of the most pressing issues in hypertext is that of 'getting lost' in the system.[9] Several of the hypertext products have browsing aids to assist in this area. GUIDE and HyperCard have 'backtrack' capability, HyperCard via 'thumbnail' sketches of recently visited cards. NoteCards and Intermedia both can generate automatic maps of the links in a system.

Opinion is divided on the value of these mechanisms. Although Intermedia will generate such maps, called 'webs', they are not found to be particularly helpful when the volume of documentation rises to a reasonable large working set. The IRIS team are now evaluating enhancements of the web concept, which they call filters, which allow the user to display hypertext links by characteristics such as author, date, authorisation level etc.

'Getting lost' in hypertext systems can be due to many different factors, not all of which are the province of the software product. Brown[5] has argued that getting lost can mean either:

(*a*) the reader does not understand the navigation features of the hypertext system.

(*b*) the reader does not understand the way in which the author has structured his information.

(*c*) the reader has taken a wrong turning and does not know how to recover.

The first of these is a factor of the user interface of the hypertext package; the other two are largely the responsibility of the author and are comparable with similar problems in other print and visual media. There is growing evidence that if the author spends creative effort on making the model of the documentation structure clear and consistent then the problem can be largely eliminated.

5. LARGE-VOLUME HYPERTEXT

Although the GUIDE hypertext system was initially developed as a retail product for single users, its applicability to corporate documentation was recognised at an early stage. A number of systems based on it have been designed for application to large technical documentation systems in the engineering industry. These systems are quite diverse: they include CD-ROM based manuals for automotive repair and catalogues of spare parts; standards documents for design and production engineering in the aviation industry; the production of user documentation by computer manufacturers; and operation and maintenance manuals in power generation.

The outstanding feature of all these applications is that the number of documents is very large, ranging into the hundreds or even thousands of individual documents. The size of such collections can be substantial, such that, for example, a single CD-ROM with a capacity of 550MB is insufficient to hold a working set.

Some of the issues which are significant for small hypertext document sets become more important when a larger collection is under development, often by a team of authors.

5.1. Consistent cognitive model

The very size of these collections is not of itself important in assessing the problems of management and navigation. The most important factor is the complexity of the 'cognitive model'; the logical structure of the information which the user must know in order to become comfortable with their use of the system. It is clear that the users ability to efficiently access information is directly related to their knowledge of the document structure that they are using.[9] Whether the user is trained in the cognitive model or is required to abstract it for themselves from use of the hypertext system will vary from application to application. The 'learning curve' involved in acquiring this cognitive model, however, will be directly related to the simplicity, familiarity and consistency of the structures employed. Thus it is vitally important to ensure the consistency of the cognitive model, if possible, by the provision of software tools which 'impose' the agreed model upon the authors.

5.2 Complex structured searching

In an engineering documentation department or other structured work environment, information is a complex web of classifications and connections which transcend document boundaries. The logical connections – within a document and between documents – become inaccessible. It requires expensive training to know what documents to look in and inefficient sequential reading to find information within each document. With large, complex collections of hundreds or thousands of documents of different types, a document collection becomes inherently unmanageable. There is a requirement to construct structured searching which can be applied to the document set and which can efficiently organize and access huge collections of information on a network.

5.3 User transparency

Although the document collection may be very large the user will often only be interested in a small subset of it. Some users will never venture outside their alloted area, and some will not be permitted to! Each user should see a complete and consistent document set but only have visibility of documents which they are authorised to view.

5.4 Automatic generation of documents

Where documents are already available in SGML (Standard Generalised Markup form) or in another structured mark-up it is desirable to import these while maintaining the structure, suitably converted to hypertext form.

SGML is an international standard format for structured technical publications used by governments and corporations. It uses 'tags' or standard formats, that act as shortcuts for specifying repetitive text and graphics elements, such as page numbers, headings, footnotes, and body text.

It is thus possible to create very large volumes of hypertext documentation – not as a separate authoring process, but as part of the normal one. These issues have led to the design of a hypertext document management environment known as IDEX, which provides the management of a library of documents to one or more users, often on a network of workstations.[8] It is for these reasons that IDEX allows the development of systems which provide standardised navigation structures for its users. The resultant hypertext system has a consistency which is not dependent on the personal discipline of its authors.

5.5 Features of the IDEX system

IDEX is a document database system designed to manage large collections of technical and office documentation in hypertext format. It is designed to complement or replace paper-based systems in a professional user environment.

IDEX is usually implemented as a distributed document database on a PC network, often connected to a mini or mainframe host, supporting both authors and readers. IDEX runs locally on the PC under *Microsoft Windows*, and the documents reside on the network document server. A network-based authoring system can also export documents to compact disk (CD-ROM) format.

The core of IDEX is a hypertext browsing environment (Fig. 2). IDEX additionally provides for the indexing and retrieval of documents based on catalog cards, document display and printing controlled by style templates, and conversion to hypertext from markup in SGML-compatible format. The main factors which have been added to the GUIDE hypertext model in the construction of this product include:

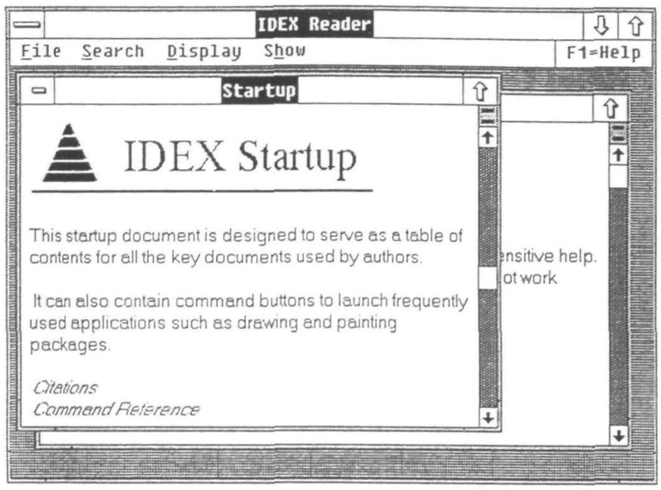

Figure 2. IDEX Startup 'Shell' Document. The two lines of text in italics are reference buttons. If selected, further documents are opened in separate windows, such as the one which is now shown 'behind'.

5.6 Classification, indexing and retrieval

5.6.1 Document type

All documents in IDEX have a document type. Document type is user defined and can be any convenient system of classification such as: 'design standard', 'specification', 'production tool drawing', 'operation flow chart', 'parts list', etc.

Document typing determines a variety of attributes, including how a document is indexed, and how it is styled in the screen display or how it appears if it is printed (e.g., font, style and size of type used for each level of heading, caption etc.). The document manager automatically creates a directory for each type and its associated style files.

When documents are being authored, or are being converted from SGML form, the document type will impose a consistent appearance and structure.

5.6.2 Document status

Document status (draft, approved, issued etc), describes the stage reached in the document 'life cycle', and is used to control the publication process, and (with other controls) to limit user access privileges.

The Document Manager application provides a formalised mechanism for changes in Document status allowing full control of the progress of a Document towards publication. Draft documents can be edited by authors, and read by editors; only approved documents can be issued; and only issued documents can be viewed by readers.

5.6.3 Catalog card

Documents can be represented to users of the system in terms of their attributes, including Document status described above. These attributes are shown through a

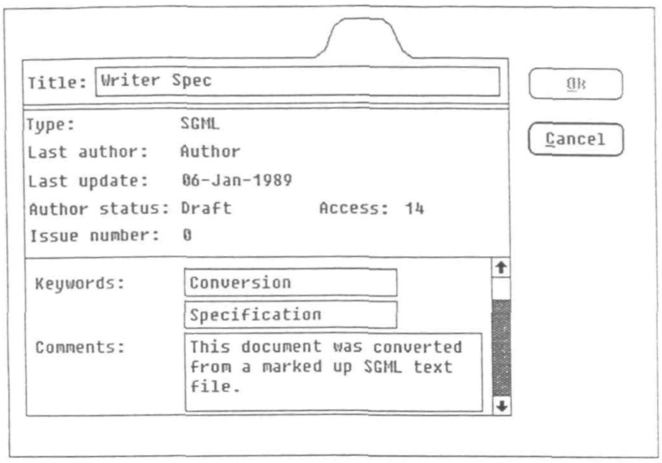

Figure 3. Example of an IDEX Catalog Card. The upper fields are mandatory; the lower fields, in this case keywords and comments, are system specific.

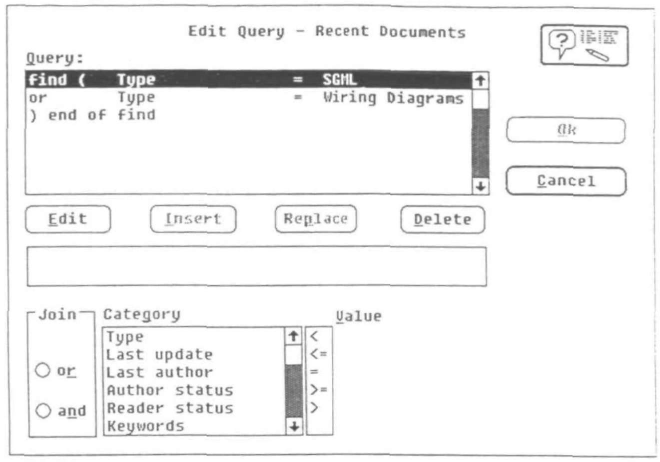

Figure 4. IDEX Query Search Criteria Editing. The user can build up a complex search criteria, shown in the upper dialog box, by combining searches from the lower dialog.

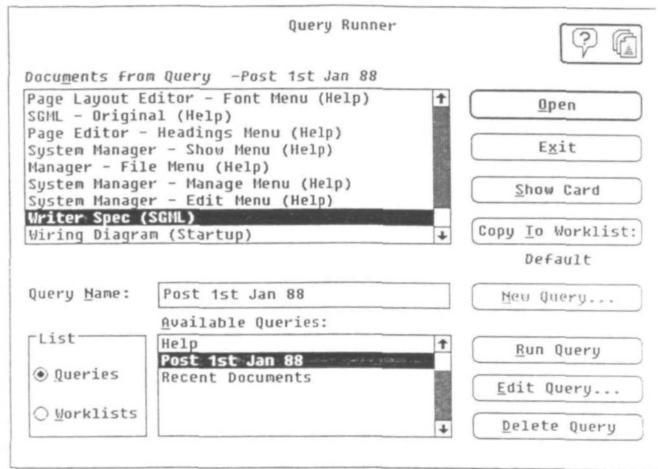

Figure 5. Example of an IDEX Query. The result of activating a Query is a list of documents which match the search criteria. Documents may then be transferred to an incremental worklist in combination with those found by other queries.

Catalog Card interface, familiar to many users. Each Catalog Card can be brought into view when the appropriate document is being browsed (Fig. 3).

5.7 Finding documents

5.7.1 Catalog card search

The computer files and directories are transparent to (or hidden from) the user; who accesses the document collection via an electronic model of a familiar Catalog Card with each document indexed by any of, or combinations of, Document Type, Date, Author, Keywords, etc. Cards for different document types can contain appropriate field combinations, thus allowing (for example) a design standard to be indexed by part number, and a drawing by scale, while sharing common fields such as author and issue number.

This also adds security and access controls. For example, purchasing departments may have read-only access to 'parts list' and 'specifications' document types, while engineering departments might have read/write access to these but no access to 'purchasing contracts'.

The user can search the indexes by defining a search criteria known as a 'Query'; complex queries can be built up using logical operators (Fig. 4). The output of the search process is either a filtered list of document titles which can be displayed in a list box in an 'open' dialogue (Fig. 5) or a view of the catalog cards. The results of such searches can be combined into Worklists which can be named and saved for future reference. Thus useful sets of documents can be presented by a descriptive name to the more naive users of the system – typically the readers – removing them entirely from the complexities of the search process. Queries can also be named and saved for convenience.

5.7.2 Authored navigation

Users of technical documents often know their system well, and may prefer to use conventional access paths such as tables of contents and lists of illustrations. The system may be configured to load and display a 'shell' document automatically (defined uniquely if required for each user). This 'shell' document acts as the top-level 'contents page' for the user and may consist of a number of contents tables which consist largely of reference buttons linked to documents concerned. Clicking on the button opens the required document (Fig. 2).

5.8 Document style management

For many authors or consumers of documents, a consistent house style is very valuable. For efficient use of a large hypertext set it is especially so. Layout and typography are powerful signposts to readers which indicate, without confusion, the function of each level of detail provided. Screen-based documentation is no exception to this rule.

5.8.1 On screen

Hypertext buttons in IDEX carry additional qualifications, such as type or level number of heading, or distinctions between references, captions, and citations. Information required for screen display: typographical style, indentation, etc. is stored for each of these in style files. Consistency across sets of documents is maintained by editing style files for each Document type. As a document is displayed, the application refers to the style file for the necessary display parameters. This approach allows both the distinctive styling of documents which the authors wish to appear different, and the easy modification of style to accessing a single display template. (Fig. 6)

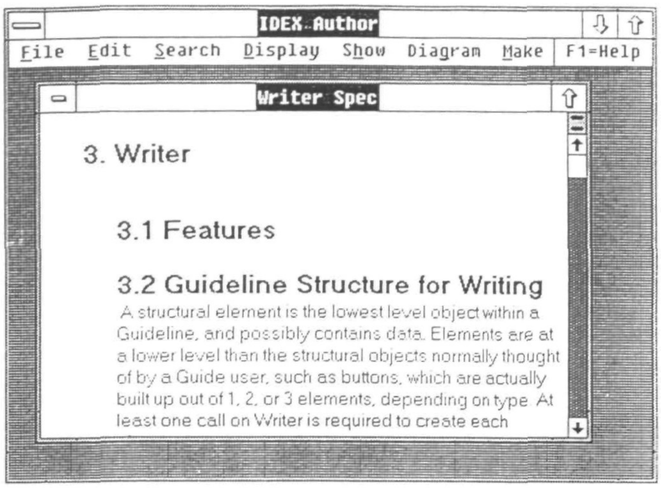

Figure 6. IDEX Document Display. The appearance of structural elements of the hypertext document, headings, subheadings, section content etc., is set in the Style definition for each Document Type.

5.9 In print

Because there will always be internal and external users who do not have access to terminals, printed versions of the hypertext documents are required.

A Page Template Editor accessed through the Document Manager allows the user to lay out the format of the printed page: not only the position of the area to receive document text and graphics, but also static information such as the Document Title, page number, and any Catalog Card attribute. These areas are formatted on demand during the printing process.

5.10 Document conversion

A major task facing the authors of a hypertext system is often the authoring of the structured document. In many cases the source material will already exist in printed or machine-held form and therefore tools are provided that simplify conversion of such source material.

5.10.1 Text

In most cases existing non-fiction documents contain some accessible text structure. If the documents exist in machine-readable form they usually contain embedded markup used by the word processor on which they were prepared. Complex documents designed for application across a range of users and uses, or to be subject to frequent updating are increasingly being encoded in a general markup language like SGML.[14,18]

If no machine-readable version is available, there are other options. With typeset texts, scanning devices using intelligent character recognition techniques can be used to substitute the markup implied by the typographic content.

A document containing suitable markup can be used to create structure for a hypertext document containing at least a hierarchy of headings. Cross references (including forward references) and notes can also be constructed where the text is well-coded.

5.10.2 Graphics

Data in vector format poses relatively few problems, and usually some simple conversion tools are all that is required to ensure the transfer. Dealing with bit maps is less straightforward. First, with bitmaps that are already digitised, conversion from one resolution to another is usually very unsatisfactory.

There is a significant difference between the resolution required for screen display of bitmaps (70–100 dpi) and that required if documents in the system are to be printed on laser printers (240–300 dpi).

In some cases the solution has been to rescan the images, and hold the data at both resolutions. Large bitmaps can also prove too large for satisfactory display on smaller screens.

5.11 Shell documents and functions

Shell (top level) or documents have Document types like any other. They have special purposes, however, such as holding contents information, lists of illustrations, a bibliography and a glossary. The bibliography contains descriptions of documents outside the hypertext collection. Citations in other documents point to individual entries, and are linked like any other cross reference. Each entry can contain structure allowing the reader to obtain more detail about a title by expanding the document.

Glossaries contain the definitions of note buttons in hypertext documents. However, the definitions are global to the collection as distinct from those that are local to an individual document. Like local definitions, glossaries can contain both text and graphics.

An important category are the Help files. It is appropriate for a hypertext system to be self-documenting. Help documents may contain internal indexes which allow context-sensitivity to be provided to any application in the IDEX suite. Context messages are passed by any relevant application to the Indexer module, and are used to find a specific point in the Help file when help is invoked by the user.

6. CONCLUSIONS

After a gestation of over 40 years hypertext is 'coming of age'. Users are now beginning to apply hypertext techniques to the publishing of economically valuable information systems. With this brings the desire to improve the quality, consistently and maintainability of such electronic documents.

Much of the virtue of the hypertext system is that, freed from the physical limitations and bulk of paper, it can be used to distribute extremely large volumes of documentation quite inexpensively.

The technology required to achieve this, however, is not trivial, since there are few useful high-volume precedents from the world of paper-based documentation. The best basis is the experience gained by applying hypertext systems to large applications such as engineering documentation.

Authors are engaged with the problems of how best to present information clearly, concisely and attractively. They need powerful tools that ease the task of converting existing documents to new formats, and which enable the management of the entire system. They should be allowed to concentrate their main efforts into the content, and not in struggling with the delivery mechanism, of their information system.

It is now becoming possible to offer generic tools which can be used with confidence by groups of authors wishing to make the best use of this exciting new technology to create complex, voluminous and easy-to-read documents.

REFERENCES

1. R. M. Akscyn, D. L. McCracken and E. A. Yoder, KMS: a distributed hypermedia system for managing knowledge in organisations. *Comm. ACM* **31** (7), 820–835 (1988).
2. S. Askwell, Computer supported reading vs. reading text on paper: a comparison of two reading situations. *International Journal of Man–Machine Studies* **22**, 425–439 (1985).
3. P. J. Brown, Viewing documents on a screen. In *CD-ROM The New Papyrus*, edited Lambert and Ropiequet, pp. 175–184. Microsoft Press, Redmond WA (1986).
4. P. J. Brown, A simple mechanism for authorship of dynamic documents. In *Text processing and document manipulation, Proceedings of the International Conference*, edited J. C. van Vliet. Cambridge University Press (1986).
5. P. J. Brown, 'Hypertext the way forward'. Paper presented at EP88, Nice, France (1987).
6. V. Bush, As we may think. *Atlantic Monthly* **176** (1), 101–108 (1945). (Reprinted in *CD-ROM The New Papyrus*, edited Lambert and Ropiequet, pp. 3–30. Microsoft Press, Redmond WA (1986).
7. J. Conklin, Hypertext: an introduction and survey. *IEEE Computer*, 17–41 (1987).
8. P. Cook and I. Williams, Design issues in large hypertext systems for technical documentation. In *Hypertext* vol. 1, edited R. McAleese. Ablex, Norwood, NJ (1989).
9. D. M. Edwards and H. L. Hardman, Lost in hyperspace: cognitive mapping and navigation in a hypertext environment. In *Hypertext* vol. 1, edited R. McAleese. Ablex (1989).
10. D. Englebart, A conceptual framework for the augmentation of man's intellect. In *Vistas in Information Handling*, 1, edited Howerton and Weeks. Spartan Books, Washington DC (1963).
11. D. Engelbart and W. English, A research centre for augmenting human intellect. In *Proceedings of the 1968 Fall Joint Computer Conference. (Montvale, NJ)*, pp. 395–410. AFIPS Press (1968).
12. F. G. Halasz, Reflections on noteCards: seven issues for the next generation of hypermedia systems. *Comm. ACM* **31** (7), 836–852 (1988).
13. W. Hershey, Guide. *Byte* (1987).
14. *ISO 8879 Standard Generalized Markup Language (SGML)*. ISO, Geneva (1986).
15. T. H. Nelson, *Computer Lib/Dream Machines* (self-published by Mindful Press, Ted Nelson. 2320 Marinship Way, Sausalito, CA 94965. (1974). (Second edition: Tempus Books (Microsoft Press), Redmond, WA (1987).)
16. T. H. Nelson, *Literary Machines* (1981; self-published by Mindful Press), (Hypertext Edition 87/1, OWL International 1987.)
17. Managing immense storage. *Byte*, 225–237 (1988).
18. J. Smith, Starting to use the standard generalised markup language (SGML). In *Proceedings of 'Electronic Publishing' conference London, September 1986*, pp. 65–74. Online (1986).
19. D. Tebbut, Guide. *Personal Computer World* (1988).
20. A. van Dam, Hypertext '87 keynote address. *Comm. ACM* **31** (7), 887–895 (1988).
21. N. Yankelovich, N. Meyrowitz and A. van Dam, Reading and writing the electronic book. *IEEE Computer* **18** (10), 15–30 (1985).
22. N. Yankelovich, B. Haan, N. Meyrowitz and S. Drucker, Intermedia: the concept and construction of a seamless information environment. *IEEE Computer* (1988).