Hypertext and Hypermedia

Carl Grafton, *Auburn University at Montgomery*
Anne Permaloff, *Auburn University at Montgomery*

Hypertext is a term borrowed from science fiction. It refers to heavily cross indexed written material organized so that the reader can move easily from one category or subcategory to another. An encyclopedia is perhaps the most widely used example of hypertext.

Microcomputer hypertext authoring software allows convenient creation of hypertext materials. Hypertext viewing software permits hypertext to be read on the monitor and in some instances printed or saved to disk. The term hypermedia expands on the hypertext idea by adding pictures and sound to text.

Hypermedia software became popular when the program HyperCard was bundled with Macintosh computers. Except for a Windows program called ToolBook, HyperCard 2.0 has more features than IBM and IBM compatible hypermedia programs, and with the same exception, those features shared by HyperCard 2.0 and IBM and IBM compatible programs are usually more powerfully rendered in the Macintosh program.

There is little point in comparing HyperCard with the IBM programs discussed here. A Macintosh user has no reason to move to an IBM machine for hypermedia applications.[1] An IBM user has available adequate hypermedia choices. Since most IBM users select their machines to perform specific tasks, there is little reason for a switch from IBM to Macintosh just because of hypermedia applications.[2]

A Hypermedia Example

The creation of tutorials is a major use of microcomputer hypermedia. For example, a hypermedia program could be used to write an interactive tutorial on Congress that begins with an introductory screen describing Congress' powers and basic organizational structure. The reader would notice that some words or phrases are highlighted. On-screen instructions would inform the reader that by placing the cursor over highlighted text and pressing a particular key, a more detailed description of that text will appear.

One of the highlighted terms on the first introductory screen might be "committees." By highlighting "committees" and clicking a mouse button, the Enter key, or another designated key, the reader would see more detailed information on that subject. The committees material might contain highlighted terms such as "seniority" and "subcommittee." Jumps could be made to detailed treatments of these subjects. "Seniority" and "subcommittee" text might also be accessed from other parts of the hypermedia treatment of Congress.

The Congress tutorial might include a map of the United States and allow the user to point to a state with the cursor and call up information on the state's senators and representatives, district maps, and economic and political characteristics.

Structure and Hypermedia

Hypertext and hypermedia consist of discrete information packets that can often be accessed in virtually any order. The advantage of such flexible access is that readers may pursue their interests wherever they lead and may provide much of the structure there is to the reading process. This central feature of hypermedia is often over-promoted by its advocates. For example, one author writes:

> By definition, linear writing is two dimensional. Words go across the line and the lines flow down the page. Hypertext, on the other hand, is three dimensional; it gives writing *depth*. . . . Without hypertext, the people reading your document might have to look to a dictionary or encyclopedia to find an explanation of a difficult or obscure topic. Hypertext links allow any word or phrase to be linked to additional information. . . .[3]

Many disagree with one-sided characterizations of this sort. Traditional linear writing permits and even encourages the use of a variety of devices such as footnotes, appendices, and bibliographies that provide whatever detailed definitions and explanations are needed.

Hypertext advocates also argue that the structure of standard text inhibits and limits readers. This thesis misses the point that structure assists readers by guiding them from one point to the next. Careless use of hypertext can encourage writing and thinking in discrete, disconnected fragments.

Not everyone who sees value in hypermedia is guilty of promoting structureless writing. Many are careful to stress the importance of structure and how structure can be built into hypermedia by carefully limiting the reader's flexibility of movement from one part to another.[4]

Ease of Use

Many inexpensive hypermedia programs including those represented here and the Macintosh program HyperCard require no programming. Two of the programs discussed below (HyperPAD and ToolBook) and HyperCard contain powerful and complex programming languages that can be accessed if sophisticated applications are being developed.

Other Hypermedia Applications

The Congress tutorial is an example of basic hypertext or hypermedia usage, but hypermedia and hypertext also lend themselves to the

https://doi.org/10.2307/419416 Published online by Cambridge University Press

creation of: custom made terminate and stay resident (TSR) help programs that accompany a host program such as a word processor or statistics package; TSR or non-TSR flat file database programs (containing, for example, mailing lists); specially designed classroom exercises; simulations; DOS Shells; presentations; and cross-indexed research files.

Hypermedia programs can be especially useful for the generation of presentations because such presentations need not follow a rigid predetermined path. Hypermedia presentations may be dynamic. They can allow for questions from the audience or class that change the direction of the presentation just as questions can change the direction of an ordinary verbal presentation.

Hypertext Creators and Users

It is important to distinguish between the creation of a hypertext application and its use. Sometimes the creator and the user will be one person as would commonly be the case with a presentation. In other instances, they will be different people, and the user may have few computer skills.

Major Categories

Table 1 displays major features of the programs discussed below. Table 2 shows which programs can be used for particular applications. Among the unobvious features considered in Table 1 are:

- *Operating mode.* Some of the programs can operate in terminate and stay resident (TSR) mode, popping into and out of the foreground over another program. Others are standard DOS software, and still others require Microsoft Windows 3.0.
- *Paste from hyper to application.* The user can move (paste) text from a TSR hypertext application to a host program such as a word processor.
- *Application context-sensitive.* The TSR hypertext application (especially a help program) can be designed so that information popped up over a host program is automatically appropriate to the user's needs.
- *DOS batch command in application.* An application can include the execution of a DOS batch file.
- *Link to external program.* An application can run an external program without leaving the application.
- *Drawing tools.* This feature creates illustrations as part of the hypermedia content or gives screens a professional appearance.
- *Map of links.* A link map looks much like a road map. It allows the author or user to see the structure of connections among hypertext components.
- *Tour.* This feature puts a hypermedia application on autopilot, giving a first time user a guided tour through a hypermedia application.

TABLE 1.
Hypertext Features

Feature	Program							
	HRez	PC-B	HWrite	BlackM	HPad	SmarT	HTies	ToolB
Operating Mode	TSR	TSR	DOS	DOS	DOS[a]	WIN	WIN	WIN
Paste from hypertext to application	N	Y	N	N	N	Y	N	Y
Application context-sensitive	N	Y	N	N	N	N	N	N
Jump to specific line or word	Y	Y	Y	Y	Y	Y	N	N
Simple search in application	N	Y	Y	Y	Y	Y	Y	Y
Boolean query in application	N	N	Y	Y	Y	Y	N	N
Pop up windows in application	N	Y	Y	Y	Y	N	N	Y
Sort in application	N	N	N	N	N	N	N	Y
DOS access in application	N	N	Y	Y	Y	Y	Y	N
DOS batch command in application	N	N	Y	Y	N	Y	Y	N
Link to external program	N	N	N	N	N	Y	Y	Y
Auto index and/or table of contents	N	N	N	N	N	Y	Y	N
Auto link creation	N	N	N	N	N	Y	Y	N
Drawing tools	N	N	N	N	Y	N[e]	N	Y
Bit-mapped (pixel) graphics	N	N	Y	Y	N	Y[e]	Y	Y
Print to disk in application	N	N	N	N	N	Y	Y	Y
Print to printer in application	N	Y	Y	Y	Y	Y	Y	Y
Mouse support	N	N	N	N	Y	Y	Y	Y
Touch screen support	N	N	Y	N	N	N	Y	N
Sound in application	N	N	Y	N	N	N	N	Y
Compact disc control in application	N	N	Y	N	N	N	Y	Y
Programming/scripting	N	N	Y[b]	N	Y[c]	N	N	Y[c]
Import ASCII	Y	Y	Y	Y	Y	Y[d]	Y	Y
Import dBASE	N	N	N	N	Y	N	N	Y
Tour	N	N	Y	N	N	Y	Y	Y
Map of links	N	N	Y	Y	N	N	N	N

[a] Limited. Cannot be used to build full TSR applications.

[b] Limited. Not a full programming language.

[c] Programming is not necessary for creation of useful applications; however, this is a relatively full featured language that can add significantly to an application's usefulness.

[d] Also imports Ami, Ami Pro, WordPerfect 5.0, and Microsoft Word files.

[e] Can use files created by Windows drawing program and many others.

https://doi.org/10.2307/419416 Published online by Cambridge University Press

TABLE 2.
HyperMedia Applications

	Programs							
Application	HRez	PC-B	HWrite	BlackM	HPad	SmarT	HTies	ToolB
Tutorial	Y	Y	Y	Y	Y	Y	Y	Y
TSR help	Y	Y	N	N	N	Y[a]	Y	Y[a]
TSR database	N	Y	N	N	N	Y[a]	Y	Y[a]
Database	N	Y	Y	Y	Y	Y	Y	Y
Simulation	Y	Y	Y	Y	Y	N	Y	Y
Cross-indexed research	N	Y	Y	Y	N	Y	N	N
DOS Shell	N	N	Y	Y	Y	na	Y	na
Presentations	Y	Y	Y	Y	Y	Y	Y	Y

[a] Running in one window while application in another.

na = not applicable.

Hardware

Except for ToolBook and SmarText that require the relatively hardware intensive Windows 3.0, the programs examined operate on minimal hardware configurations of an 8088 based computer with 640K RAM and a hard disk. Several do not require a graphics card.

TSR Programs

HyperRez. MaxThink, 44 Rincon Rd., Kensington, CA 94707. (415) 540-5508. 256K and floppy. Although the author asks for no registration fee, HyperRez is not in the public domain, i.e., the author retains control of it. Obtained for $3.00 plus $4.00 S&H from Gemini Marketing, Inc., P.O. Box 640, Duvall, WA 98019-0640. (800) 346-0139 and (206) 788-4295. Fax: (206) 788-0717. Also available via other shareware distributors.

PC-Browse, Version 1.0. Quicksoft, 219 First Ave. N., #224, Seattle, WA 98109. (206) 282-0452. $49 for full registration, newest version, printed manual, and one year telephone support. Manual alone $24. Registration only $10. Obtained from Gemini Marketing (see above).

Hyperties, Version 3.0. Cognetics Corp., 55 Princeton-Hightstown Road, Princeton Jct., NJ 08550. (800) 229-TIES. $379.

HyperRez

HyperRez is a TSR hypertext reading program. Text is created using any text editor or word processor that generates an ASCII file. The hypertext author indicates jumps (usually called links) with the file name to be jumped to shown in greater than and less than symbols such as <filename>. If the link is to a specific page, the syntax might be <filename 2> for page 2. If the link is to be made to a page and line, the syntax is <filename 3 5> for page 3, line 5. A link can also be made to a specific word in a file with <filename-word>. Unfortunately, the HyperRez end user sees the greater than and less than symbols and file names in the text, but their presence is not bothersome if they are confined to the beginning or end of lines or paragraphs.

Each file in a HyperRez collection of files may be no longer than 20K. This is the equivalent of approximately 10 full screens. The only practical limit to the number of files that can be accessed by a given application is disk capacity.

When a set of files has been created containing appropriate hypertext links, HyperRez is invoked. Screen colors (background, foreground, etc.) are adjustable as is the hot-key combination that brings it up. The default hot-key combination is Ctrl-H.

Once HyperRez is invoked, it operates entirely with the arrow keys and − and + keys except when an exit is made from the program. On a hard disk, jumps are virtually instantaneous.

Documentation for HyperRez is in a format readable by HyperRez. HyperRez documentation is an excellent example of how confusing hypertext can be to use when it lacks structure. In seeking the program's syntax rules, the hypertext documentation offers no easily followed guidelines. Growing tired of wandering through the information in an almost random fashion, we finally resorted to loading the hypertext documentation files one at a time into a word processor.

HyperRez operates flawlessly. It even works when it and the files that constitute a hypertext application are on a different disk than the word processing software that is being used. HyperRez pops up over the software, runs, disappears on command, and returns repeatedly without difficulty. The HyperRez syntax is simple as is the operation of the program itself.

HyperRez would work for on-line documentation for other programs such as statistics packages, but such documentation would not be context sensitive. PC-Browse (described immediately below) is a better choice for this application. HyperRez would also work for the construction of a tutorial, simulation, or dynamic presentation as long as the display of bit mapped (pixel) graphics is not a requirement. HyperRez can display text graphics.

PC-Browse

Like HyperRez, PC-Browse is a TSR program and shares many of HyperRez's basic hypertext characteristics. PC-Browse application files are created with any word processor capable of saving in ASCII format. Control codes are entered surrounding words or phrases from which jumps to other parts of a file or to other files are made. Movement backward or forward between these links is easy for the user to initiate. Jumps can also be made to a particular place in a given file such as a page number or specific word.

Unlike HyperRez, PC-Browse has search capabilities comparable to most word processors. Searches may be made among files or within a single file. Searches among files may include DOS wild cards.

Information can be pasted from the hypertext screen onto an application program as long as it is text based. Thus PC-Browse lends itself well to having a database created using any database management program capable of generating ASCII text sitting over an application

https://doi.org/10.2307/419416 Published online by Cambridge University Press

program such as a word processor or spreadsheet. Data can be located and then pasted into the application.

PC-Browse users can place the cursor in an application program over a word that can cause PC-Browse to jump to the topic indicated by that word. This feature allows the creation of a context-sensitive help system. For example, with a statistics program, the user could highlight the word regression and a definition of regression analysis might appear.

Most PC-Browse commands are shown on screen as prompts, making it very easy to use.

Hyperties

Hyperties is the only full featured hypertext program represented in the TSR category. It is far more expensive than the others, but, aside from the TSR feature, it does not compare favorably with several programs discussed below.

Hyperties consists of two central programs, one for writing applications and the other for reading them. Since the Hyperties reading program may not be copied freely, widespread distribution of a Hyperties application would represent additional cost.

The Hyperties writing program is essentially a text editor designed for hypermedia creation. Its on-screen prompts are so helpful that the process of writing an application requires little reference to the manual.

Unfortunately, Hyperties text formating techniques are primitive. For example, text that is to be single spaced must be preceded with the symbols @s1, and a left margin of five spaces requires the symbol @L5.

Hyperties' links only allow movement from one file to another. Jumps within a file cannot be made nor is it possible to jump to a specific segment of another file.

Hyperties supports all major graphic modes including text graphics. Graphic images may be pictures or background for text. An example of the latter is a graphic background consisting of lines and boxes to emulate index cards.

Because Hyperties does not allow movement to a particular location in a file, an author using this program will be inclined to divide hypertext into many small files. Fortunately, Hyperties automatically creates an index for its files, a feature that helps author and end user alike. Hyperties also facilitates creation of a table of contents. When words or phrases are selected to trigger links to other files (by surrounding them with tildes), Hyperties can run through an automatic procedure that insures that all links have an end point.

Hyperties imports graphics in TGF, PCX, and text formats as well as ASCII text. It supports touch-screen monitors (the only program discussed here that does) and videodisks.

A Hyperties application can be quite professional in appearance and operation; however, other programs discussed below give the application author access to a greater variety of tools for a lower cost.

Non TSR Programs

HyperWriter, Version 2.5. NTERGAID, Inc., 2490 Black Rock Turnpike, Suite 337, Fairfield, CT 06430. (203) 368-2174. Fax: (203) 334-5658. $395. 640K and graphics board.

Black Magic, Version 1.51. NTERGAID (see above). Black Magic is a shareware predecessor of HyperWriter. $99.95 full registration, Version 2.0, and printed manual. 640K and graphics board. Obtained from Gemini Marketing (see above).

HyperPAD, Version 2.2. Brightbill-Roberts, Ltd., 120 E. Washington St., Suite 421, Syracuse, NY 13202. (315) 474-3400. $49.99 mail order discount. 512K.

HyperWriter

HyperWriter consists of a writing and a reading program. The reading program has been placed in the public domain so that hypertext applications created with HyperWriter may be freely distributed with the software necessary to read them. Although HyperWriter is not a TSR program, an extra module can be purchased from the manufacturer that will allow the creation of a TSR hypermedia application.[5]

HyperWriter consists of a rather full featured text editor that includes formating tools and spell checker. Text may be typed and various hypermedia links added entirely within a single environment. ASCII files may also be imported.

Hypermedia applications generated with this program can be quite sophisticated since HyperWriter uses three types of links: text; graphics; and actions. Each of these link categories has a number of subcategories.

One kind of text link is the standard one that allows the reader to jump from a highlighted word to another part of a current file or to any part of another file. A second link generates a pop up window that provides additional information without leaving a current location. A third text link operates somewhat like a word processing block insert. When it is invoked, a block of text is inserted directly into the area where the cursor is located. The fourth text link allows the reader to jump to an ASCII file.

HyperWriter also permits jumps to bit mapped graphics that can either be popped up as a window or may cover the entire screen. Graphics, such as a bar chart or a photograph brought into the computer via scanning, may be imported using a TSR utility program included with HyperWriter.

Action links are of two types. One executes a series of DOS batch commands; the other starts a number of HyperWriter scripting (programming) commands. The HyperWriter language is highly specialized and not a complete programming language.

A HyperWriter script can play a tune (either through the computer's speaker or full digital sound played through Video Associates or Covox equipment), write a report to disk in ASCII format, pop up a menu of links, or play part of a videodisc (Sony or Pioneer).

A large hypermedia application can become so complex that the writer loses track of its content and how components are connected. HyperWriter assists the writer by generating a map of the entire hypertext structure showing these connections.

HyperWriter can perform Boolean searches within a file. Searches across files are also possible if the author adds an index file.

Another feature called "tour"

https://doi.org/10.2307/419416 Published online by Cambridge University Press

takes the reader through a predefined sequence of linkages to give a structured overview of a hypermedia application. HyperWriter also allows for the creation of simple macros.

Black Magic

Black Magic is distributed through shareware channels. As Hyper-Writer's predecessor, it contains many of its best characteristics, and is a highly capable product in its own right.

Black Magic contains a text editor that permits block moves, different styles of text, underlining, and italics, as well as adjustable window sizes, margins, and color. It allows text links to a different part of the current file or to another file, links to an ASCII file, windowed notes that pop up over current text, and graphic links. The TSR program required to pull graphic images from other programs is included as are other features such as a DOS link, macros, a simple word processor-like search capability, and a mapping capability.

The HyperWriter features Black Magic lacks are primarily those associated with HyperWriter's scripting capabilities. Both programs are accompanied by superb tutorials that combine the structure needed to guide someone new to the program and the freedom that allows the more experienced to strike out on their own.

HyperPAD

HyperPAD is considerably different from the other programs discussed here. Despite its extremely modest system requirements and low price, HyperPAD is an object oriented application development environment operated via menu choices and sometimes by programming. It is similar to HyperCard and ToolBook.

HyperPAD can be used to create a variety of applications that do not resemble traditional tutorial-like hypertext. Provided on disk are a Rolodex-like address book, a daily planner, a hand calculator, and an animated puzzle. Viewing these applications, the user could easily conclude that they had been developed using a traditional programming language. Only the excellent tutorial also included on disk suggests HyperPAD's hypertext origins.

While the programs discussed above view application development largely in terms of linking screens of information, HyperPAD applications are built up in layers, like a stack of transparencies. The idea of linkages between various screens is present as well.

The foundation of a HyperPAD application is called a background. A background consists of items that appear on most or all screens. For example, the address book that comes with HyperPAD shows a "card" complete with Rolodex "slots" in the bottom and lines on which to write names, telephone numbers, and addresses.

A background may consist of pictures and/or objects. A picture (that can be imported or created with HyperPAD's modest drawing tools) may be decorative or informative, but it does not "do" anything. There are two kinds of HyperPAD objects, fields and buttons. A field is a location where text may be entered or displayed. Field characteristics such as size, the seemingly obligatory shadow effects, screen location, and internal properties such as scrolling are determined by menu choices. The other kind of object is a button. Selection of a button triggers an event such as movement to a new screen or new file. As with fields, button location, size, and other attributes are determined by menu choices. Once a field or button is created, it may be copied to other screens with all of its characteristics part of the copy. This is one of the central features of object oriented programming.

Pictures and objects attached to a background appear on all Hyper-PAD application screens. Layered on top of a background the user may find a "page." Like a background, a page may contain pictures and/or objects with the same kinds of qualities as the pictures and objects described above; however, pictures or objects associated with a page may be made unique to that page.

A particular screen might consist of pictures and objects present in all screens (as parts of the background) plus additional pictures and objects unique to that screen. All the user would see would be various prompts, data input areas, buttons, and various decorative touches.

HyperPAD can import ASCII and dBASE III files. It can launch other applications and remain resident so that it can pop up again when those applications are exited. Thus, it can be used as a DOS shell. HyperPAD applications may include Boolean searches and sorts.

Even more powerful applications can be developed using the HyperPAD programming or "scripting" language containing well over 200 commands. This language contains a full complement of control devices such as "if-then-else" and "repeat-until" as well as ways to affect the appearance of the screen.

HyperPAD could be used for all the applications listed in Table 2 except those requiring TSR capabilities and those in which the display of bit mapped graphics is necessary.

Windows Hypermedia

Both Windows programs represented here have the usual hardware and software requirements associated with Windows. They are irritatingly slow on 386 based machines of average speed using 1 MG disk caches. Both display the fuzzy text common to all Windows software and suffer from the plague of cute but obscure icons that Windows developers seem to delight in creating. For example, an icon showing a person looking through binoculars represents a Boolean search.

SmarText, Lotus Development Corp., 5600 Glenridge Dr., Atlanta, GA 30342. (800) 831-9679. Fax: (404) 255-9460. $495.

ToolBook, Version 1.5, Asymetrix Corp., 110 110th Ave., NE, Suite 717, Bellevue, WA 98004. (206) 637-1500. $289.

SmarText

The SmarText manufacturer recommends that 10MG of disk space be available. Approximately 3 MG are required for the program itself. The creation of hypertext with no graphics would not require a full 10MG, but the addition of VGA pictures consumes disk space quickly.

https://doi.org/10.2307/419416 Published online by Cambridge University Press

From an application developer's perspective, SmarText is in some ways the easiest to use program represented here. Text and graphics are created using other programs including the Windows text editor and painting program. Text and graphics are then imported into SmarText. SmarText can read ASCII, Ami, Ami Pro, Microsoft Word 5.0, and WordPerfect 5.0 text files as well as Bitmap, PC Paintbrush, and TIFF graphic files.

SmarText itself contains no significant formating or editing tools. Editing and formating must be done before text is imported by SmarText.

SmarText's greatest power comes from its ability to almost automatically create hypermedia from the text and/or graphics brought into it. Specifically, it generates an outline, index, and links. The reader can call forth the outline at any point and then jump to any point in a document from any point in the outline. The same procedure can be accomplished from the index except that a single index entry often leads to several possible jumps as is the case with an ordinary book index. An outline and index could also be created with the other programs, but this would require much more effort.

SmarText's automatic creation of outline, index, and links is done via a content analysis by which the program attempts to determine what words are most important. This process can produce bizarre results. For example, it created links to several unimportant words such as "little" used in the sentence "There seems little point in. . . ." SmarText offers several tools to control this sort of thing, and if they fail, which is likely, changes can be made manually.

SmarText also permits the creation of "views" which allow readers to see only specified portions of material in a particular order. Multiple views of a single document can be created for different users. A view can be created manually, be based on a previously created view, or be based on the selection of index words.

A note feature allows hypermedia readers to leave the equivalent of Post-Its throughout a SmarText document. Readers may also use powerful Boolean search capabilities and print selected portions of text.

A SmarText end user would have to be somewhat familiar with Windows to be able to load and view hypermedia created with this program. Given the fact that some hypermedia resources are intended for those with little computer expertise, this is a serious drawback. Loading a file is far too difficult with SmarText. It is much too easy to pile windows on top of windows, each showing a different part of a file, in a completely confusing jumble. A person untrained in Windows usage could only escape by trial and error.

Accidental exits from SmarText can occur with no warning. Nothing is lost by such exits, but someone unused to Windows might have difficulty reentering the program and the file being read.

SmarText's most serious weakness from the perspective of the hypermedia author is its dearth of formating tools. SmarText does not allow the hypermedia designer to limit user access to information in a strictly controlled fashion, one specific screen at a time. It is oriented to allowing free user movement throughout the text. It is, therefore, a poor tool for the creation of simulations. SmarText's high price will serve as another barrier to its use especially since SmarText reading software costs an additional $99 per copy.

ToolBook

ToolBook can produce the most attractive applications of any of the programs discussed here even though their long term use irritates the eyes because of the fuzzy text generated by its Windows host. ToolBook contains a vast array of graphic tools including a competent drawing and painting module to create virtually any imaginable graphic and text combination. ToolBook can also import scanned images. ToolBook's text formatting tools are complete. They even include table formatting capabilities.

ToolBook is an object oriented programming environment. It builds screens in layers via menu choices. Object characteristics defined in menu choices may be supplemented with ToolBook's highly sophisticated scripting language.

Although ToolBook produces the most professional looking screens of the programs discussed here, creating them and the links between them requires a great deal of effort. The creation of links, an index, and a table of contents are entirely manual processes, making ToolBook one of the least efficient hypermedia development tools represented here. And, ToolBook does not automatically create a map of links.

While ToolBook is not an efficient device for creating hypertext and hypermedia applications, it is more than a hypermedia authoring program. It is a large, powerful, and highly complex object oriented language that can be used for most programming needs. However, there is considerable doubt whether it would be the best choice for this purpose since it is competing with the less expensive and faster Visual BASIC, a package that will undoubtedly spawn a much larger reference literature and be supported by far more third party programmers.

Conclusions and Recommendations

Those working with the major microcomputer platforms—the Macintosh, all manner of IBM and IBM compatible machines, and computers using Windows 3.0—have an opportunity to use excellent hypermedia software. All of the programs described here operate well, come with clear documentation (except HyperRez), and represent good value. One's selection among them would depend on the application being built together with the features offered by each program. For example, TSR requirements leave only three programs in contention, and the need for high resolution bit mapped graphics narrows the field. If Windows and the TSR feature are not requirements, the superb shareware version of Black Magic is the best place to start. If more power is needed, Black Magic's big brother, HyperWriter is an excellent value.

Windows represents the direction toward which the IBM and IBM compatible world is moving at least

https://doi.org/10.2307/419416 Published online by Cambridge University Press

in the near to medium term. If Windows is a requirement and if rapid, partly automated creation of relatively unformatted hypertext is all that is needed, SmarText is a good if expensive choice. The far more difficult and time consuming ToolBook provides a strong collection of techniques for controlling every aspect of screen appearance and movement.

Notes

1. Hypercard dominates the Macintosh hypermedia world because it is bundled with the Macintosh and is an excellent program. Consequently, Hypercard has spawned a growing industry of add-on software and reference books. Other manufacturers of Macintosh hypermedia programs refused to provide copies of their products for comparison with Hypercard.

2. Those interested in Hypercard features should consult Danny Goodman, *The Complete HyperCard 2.0 Handbook* (Toronto: Bantam Books, 1990).

3. J. Scott Johnson, *Hyperwriter! User's Guide* (Fairfield: Ntergaid, 1990), p. 2.

4. Robert E. Horn, *Mapping Hypertext* (Lexington: The Lexington Institute, 1989).

5. Help System II, $199.95.

Murray Edelman in Appalachia

Michael Lipsky, *Massachusetts Institute of Technology*

Author's Note: These recollections were prepared for a dinner at a conference honoring Murray Edelman's work upon his retirement from the University of Wisconsin.

In the spring of 1970, our friend Murray Edelman, having decided that being a staff member of the Institute for Research on Poverty did not provide as much exposure to the world of poverty as might be desirable for someone in his position, determined to tour the hills of Appalachia to see for himself. He took along as guides two trusted companions, James Scott, a specialist on peasant politics, and Michael Lipsky, a specialist in the political behavior of the urban poor. (You never know what expertise you may require when exploring new regions.) If you follow this memoir closely you will learn, perhaps, to the best of our knowledge, why Murray never engaged in direct field work again.

We assembled by separate planes at the airport in Charleston, West Virginia, rented a car, and proceeded to drive south and west into the Appalachian mountains, through the defile of the infamous and unfortunate Hazard, Kentucky, and on to Harlan, the largest town in the southern Kentucky mountains. Here we stopped at a motel for the night.

In the morning, over breakfast, we read in the daily paper that a local youth had been charged with murdering his brother-in-law. Mayhem and intrafamilial violence! We began to feel we had come to the right place. After breakfast we got in the car to drive into the countryside, and we picked up a young male hitchhiker just outside of town.

"Where are you from?" one of us asked, to be polite but also to begin to interview the first research subject we had encountered. He named the town where the murder had occurred.

"Really?" one of us replied. "That's the place where some kid just murdered his brother-in-law." This was said with the pride social scientists muster when they discover they actually know something about a subject under discussion.

"Yup," the boy said. "That was me."

The interview was not going in a predictable direction.

The remainder of the encounter with our interviewee was, you might say, strained. We learned that he had shot his brother-in-law because the deceased had been beating the fellow's sister and had not responded to previous and less drastic efforts to persuade him to change his ways. Our rider had been arrested the previous day and released on bail that morning. We supposed that he, as a local boy, expected to beat the charge by claiming justifiable homicide. Each of us no doubt filed the episode away for recall when next required to lecture on comparative legal systems.

During the day we hiked up a stream valley that had been taken for a hydro-electric project but had not yet been flooded by the waters rising behind a new dam. This was an exercise in contemporary archaeology, as all that remained of the settlements in the valley were the daffodils and occasional lilacs that bloomed in erstwhile front yards.

Later that day we came upon a gentleman who urged us not to proceed up a particular "holler" because a friend of his had a private coal mine up there for household use and was known to shoot people who threatened his supplies. We noticed that this family displayed its poverty in what seemed to be peculiar ways. The family owned every conceivable large household appliance: refrigerator, freezer, washing machine, and so on. But all the appliances were on the collapsing front porch rather than inside. Was this evidence that the family was so large that there was no room inside for the appliances? Did it represent a form of conspicuous consumption? We never did trace the origins of this practice or determine whether it was widespread.

We went back to Harlan that night but Professor Scott insisted that we move our quarters. We had been too comfortable the night before, he complained. We had not experienced the full depravity of the region. We needed, he insisted, to locate and take up temporary residence in the worst hotel in town.

Who could stand up to so persuasive an argument, particularly when it was dressed up in the veils of research? Off we went to find the worst hotel in Harlan, Kentucky.

https://doi.org/10.2307/419416 Published online by Cambridge University Press