SESSION XII

HYPERCARD AND HYPERTEXT: APPLICATIONS AND USE

Thomas T. Hewett, *Presider*
Drexel University

The potential of HyperCard for psychological research and instruction: A general discussion and description of two research applications

ELKE GEISLER-BRENSTEIN and ROBERT J. BRENSTEIN
Southern Illinois University, Carbondale, Illinois

In this paper, we discuss the potential of HyperCard for research and instruction in psychology. First we give a general overview of the HyperCard program; after that, we present two HyperCard stacks as sample solutions for two specific research applications. Surveyor, a self-contained survey tool, is a HyperCard-based vehicle for developing, administering, and processing tests and surveys. Queston demonstrates how HyperCard can be used as a data-management and data-analysis tool during the stages of questionnaire development. Both stacks illustrate how flexible HyperCard is and how easy it is to use it to manage, analyze, and process data, to transfer data to other programs, and to print reports. HyperCard, unlike traditional applications, gives the user a great degree of control over the way information is stored, manipulated, and presented. Although both stacks are custom-made for specific purposes, the concepts underlying the design can be generally applied and adapted for other purposes.

Brief History and General Concept of Hypertext

Computers have become indispensable tools for the management and exchange of information. However, to become truly useful as "extensions" of the human mind, they need to be able to function similarly to the way the human mind works. As early as 1945, Vannevar Bush, the president's science advisor, recognized the need for accessing data by means of free association instead of storing information in categories and retrieving it in systematic, linear fashion. Bush envisioned a machine called Memex, which would allow rapid access through association rather than through conventional alphabetical, numerical, and hierarchical methods of indexing information (Wolman, 1988). Later, in 1965, Ted Nelson's vision of "a corpus of universally, electronically accessible writings, infinitely cross-referenceable and freed from their traditional physical limitations and conceptual boundaries" (Wolman, 1988), gave rise to the concept of hypertext, in which special linking capabilities allow nonlinear organization of text. With the advent of new technolo-

gies, the concept of hypertext has evolved into the notion of hypermedia, which interlinks different modalities such as text, graphics, digitized speech, audio recordings, pictures, animation, film clips, and possibly tastes, odors, and tactile sensations (Conklin, 1987). This new technology, which is opening up exciting new possibilities for the computer as a tool of communication and thought, has taken many different forms, ranging from multimedia databases to "idea processors." In this paper, we will discuss HyperCard, a successful commercial implementation of the hypermedia concept, and examine its potential usefulness for psychological research and instruction.

HyperCard, an Innovative and Powerful Computing Environment

HyperCard, which was released by Apple Computer in August, 1987, is included with every new Macintosh computer (it is formally an extension of the Macintosh operating system). HyperCard is a conceptually innovative program that spans traditional classifications of computer programs into databases, applications, word-processors, spreadsheets, graphics, authoring programs, etc. Because of its many features, HyperCard's potential,

Correspondence may be addressed to Elke Geisler-Brenstein, Department of Psychology, Southern Illinois University, Carbondale, IL 62901.

effectiveness, and usefulness are not easy to discuss (cf. Custer, 1987).

In brief, HyperCard offers fast and efficient information-managing capabilities, a sophisticated, easy-to-use programming environment, and extraordinary flexibility for designing or modifying the user interface. It also includes simple word-processing and graphics capabilities. The hyperexcitement and controversy its introduction has created within the Macintosh community, as well as among Apple's competitors, stems from the fact that HyperCard puts what once was the exclusive domain of professional programmers into the hands of the user. Unlike other software, which requires the user to follow prescribed patterns of interaction and rarely permits customization of a program's interface, HyperCard provides tools for tailoring interface and functionality to particular needs. As a hypertext-like information manager, HyperCard allows the user to make associative as well as hierarchical links in a database-like compendium of text, graphics, and sound. HyperCard can also be used as a front-end to other data-storage devices such as video disks (control of a videodisc player requires only an additional cable, thanks to external functions that have already been written to communicate with various players) or CD ROMs. As a data-management tool, HyperCard supports different modes of navigation through stored information, including (1) the systematic examination of contents or by following links; (2) searches through all or part of the database for a string, keyword, or attribute value; and (3) browsing, by means of a graphical overview of the network that provides contextual and spatial cues to supplement the user's model of the nodes viewed and how they are related to each other.

HyperCard gets its name from the *card* metaphor that underlies the visual representation of stored information. In HyperCard, sets of cards are grouped as *stacks*. HyperCard stacks can be thought of as virtual stacks of rolodex or index cards. Physically, each stack is just a computer file, or a HyperCard document in Macintosh vocabulary. The information on the cards can be linked with other cards or stacks by means of *buttons*. Data is stored in editable *fields*. The functionality of the buttons, the cards, the fields, or the stack as a whole are determined by *scripts*, pieces of code in *HyperTalk*, the HyperCard programming language. HyperTalk scripts can be modified, copied, and pasted anywhere in any stack, just as can any other HyperCard object.

Individual elements such as the buttons, fields, or cards can easily be tailored to suit the user's needs, tastes, and personal preferences by means of additions, deletions, or changes in their functions or appearance. For example, the user can choose among several visual effects that mark the transition from one card to the next; fields or buttons can be moved around and appear in different sizes and shapes; text can be printed in different fonts, sizes, or styles; buttons can have different icons; the user can choose among the modalities of text, graphics, or sound. The extent of the changes that can be made will, of course,

depend on the user's familiarity with HyperCard, and the degree to which the stack programmer has enabled certain functions (see Goodman, 1987). In addition, so-called *external functions* (XFCNs) and *external commands* (XCMDs)—short programs written in a high-level programming language—can be added to supplement or expand functions not supported directly by the HyperCard environment.

HyperCard allows several levels of complexity. The use of HyperCard can be restricted to any of these levels when one develops or modifies a stack. On the simplest level, the *browsing* level, a user can browse through information sequentially, looking at one card after another, or associatively, navigating among cards and stacks according to specific interests or themes. At the *typing* level, the user can enter and modify data in editable text fields. The *painting* level allows one to alter graphics with a MacPaint-like set of tools for the creation of bit-mapped graphics and text. At the *authoring* level, the user can modify and create new cards, fields, and buttons and establish links among them. Finally, at the *scripting* level, those who want to delve into programming with HyperTalk can create and change scripts of any object, and thus have full control over the HyperCard environment (see HyperCard User's Guide; Lasky, 1988; Tessler, 1988).

The simple syntax of HyperTalk allows even non-programmers to create stacks that look and feel like "real" Macintosh programs. HyperTalk is an object-oriented language, designed to recognize and manipulate the basic elements of HyperCard. It lends itself well to the development of subroutines, procedures, and functions, and it includes user-friendly error checking. Additional functions can be written in Pascal, C, or Fortran (see Goodman, 1987, 1988; Kaehler, 1988; Shafer, 1988).

As of now, HyperCard has several limitations. First, it is available only on Macintosh computers, which are more expensive than IBM PCs and compatibles (although in the same price range as PS/2 systems). It requires 1 MB of memory, or more, if another program is to be executed simultaneously (under MultiFinder). Running HyperCard from a hard disk is preferable to running it from a diskette because of the size of the program and speed of disk I/O. If digitized sounds are included in a stack, disk space requirements increase rapidly. HyperCard uses a window of fixed size, which matches the standard screen of a Macintosh Plus and SE; it cannot take advantage of larger monitors. Unlike other Macintosh programs, HyperCard does not support multiple windows or user menus. It also does not support colors or object-oriented graphics. Its text-processing capabilities are limited to simple operations. Mixing different fonts, sizes, or styles within a field is not possible. Many of its applications could benefit from more advanced database features. Although HyperTalk is quite powerful, it lacks more advanced computational capabilities, such as support of arrays. Finally, HyperCard's built-in printing and reporting capabilities are limited. Fortunately, none of these limitations should be critical for most research and instructional applications,

and many of them can be overcome with available stacks, external functions, and add-on products.

Potential for Psychological Applications

Computers are no longer just a delivery medium. They are becoming more and more popular as tools for supporting instructors in the preparation and presentation of course and training materials. Because of its low cost and flexibility, HyperCard is an attractive alternative to traditional programs for a variety of instructional and research purposes. As an authoring tool, HyperCard is unique because it is easy to learn and use at the browsing, typing, and painting levels, while at the same time, it allows access to the high-level programming features. Thus, it is making the courseware development process accessible to people without a formal background in computer programming or instructional design (Rice, 1988; Wolman, 1988). HyperCard is well-suited for courseware applications (drills, tutorials, and simulations) because it offers user-friendly hypertext-like linkage with external storage devices, in addition to its own storage capabilities. Through its ability to edit its own scripts and create cards, buttons, or fields on its own, HyperCard also has potential for anyone interested in artificial intelligence.

HyperCard has a high scanning rate (up to 40 MB of data in 2 sec or up to 16 frames per second). It uses efficient compression algorithms for bit-mapped images, so graphics come up fast and use little space. These features, in addition to the Mac's graphics capabilities, make the HyperCard environment also ideal for simulations and animations. While other programs provide similar functionality, HyperCard is preferable when time or money are important (e.g., for experimentation with a new technique), since it allows one to test, revise, update, and customize designs faster and more cheaply than traditional methods do. Moreover, to design by association, which is the way the mind works, allows the instructor who uses HyperCard for presentations or the learner who studies a subject to explore a topic without a "straight jacket"; it encourages creative exploration and problem solving (see Wolman, 1988).

In a recent review of microcomputer programs for adaptive testing, automated test generation, and computer-assisted surveying programs, Gray (1988) discusses the usefulness of such programs for facilitating the creation and implementation of instruments (surveys or tests), data collection, analysis, and reporting. Below, we discuss how HyperCard can be used as a low-budget, yet flexible and effective alternative to existing software for designing and implementing applications of this sort. Different aspects are considered: development of the instrument items, management of an item pool, design of the screen/print format for display of the items, creation of a procedure for printing the instrument or administering it interactively, analysis of the results, and generation of reports. We have found that HyperCard-based solutions can be functionally equivalent to those of commercial programs in these areas. Unlike standard applications, in which

the functionality and the user interface are largely predetermined, representing a particular developer's view of what is necessary and important, HyperCard gives the researcher control over crucial aspects of a program.

Surveyor, a Survey and Testing Tool

Surveyor is a multipurpose stack for testing and survey applications. It can be used for test or survey preparation and interactive administration, as well as for processing the response data. It allows the researcher to custom-design a particular testing or survey environment. Surveyor consists of a control stack and a data stack. The control stack contains the item pool, the actual test or survey representation, data templates, and functions for the analysis and export of data. The researcher can choose item and response formats and design the screen layouts, as well as select different procedures for small-scale statistical analysis of item- and process-related data. The data can be copied into a text file as input to other Macintosh or mainframe programs, or they can be printed with Surveyor's report-printing facility. Surveyor can be executed on more than one Macintosh computer; the data gathered on different computers can be merged into a single data set.

Computer-based surveying and adaptive testing programs are unique in that they permit interaction with respondents, which greatly facilitates the collection of information. This is accomplished through branching and randomization, which control, for example, the user's entry point (depending on demographic or other background information), item selection, pattern of response presentation (randomization or alternation of item selection and response-choice format), response criteria (acceptance of predetermined ranges), branch criteria, or feedback through visual or sound effects.

Computerized survey administration with HyperCard has several advantages over paper-and-pencil administration, and it often surpasses the capabilities of other computerized survey tools. Generally, computer-based surveys eliminate hand-coding of the data as well as avoid potential OCR errors during machine coding. Like a paper-and-pencil survey, Surveyor can support different response formats (bipolar, multiple-choice, open-ended, or matching). Unlike paper-and-pencil instruments, Surveyor incorporates program-like interactive functionality. Since HyperCard supports floating-point arithmetic, simple analyses can easily be built in and performed at any moment (more complex analysis would require writing external functions in a high-level programming language). For example, Surveyor is currently being used to survey students' opinions regarding different aspects of a particular on-campus service. In this case, a small-scale decision analysis program that requires students to choose among different options according to several weighted attributes is part of the survey.

Surveyor includes a selection of design elements and preprogrammed functions. For example, the researcher can choose from several screen templates for background

design. Different designs are also provided for various response formats (e.g., text fields for open-ended responses, text and numerical input fields, check boxes for multiple-choice formats, bar graphs for Likert scale responses, etc.). Preprogrammed HyperTalk scripts are included for handling user input (e.g., management of a group of checkboxes for multiple-choice responses, so that when one box is checked all the others remain unchecked to avoid multiple responses where they are not expected). In addition, special-purpose field and button scripts (e.g., for checking whether a given response is valid) are provided.

When a test or survey is constructed, the items can be either typed in directly or selected randomly or according to certain criteria from a given item pool. Items as well as response choices can easily be edited, changed, deleted, or added by means of standard word-processing features. Items or other elements of the instrument may also include graphics, in the form of graphs and charts, hand-drawn illustrations, or scanned-in images. The researcher has control over fonts, font sizes, or styles (underlined, bold, italics) in the fields that contain item or response information. This is important, since an attractive, easy-to-follow form is critical for obtaining cooperation. In certain instances, sound can also be utilized to enhance or supplement textual information.

To protect the data from unauthorized access, this stack is designed so that the respondent has no control of the HyperCard environment. Unlike the typical stacks that allow the user to freely explore and modify the HyperCard environment, Surveyor prohibits any user action that has not explicitly been provided by the programmer or owner of the stack. All regular HyperCard functions, command keys, etc., are disabled. Only the person who knows the proper access code can exit the survey loop and access the stack's control card, patterned after the HyperCard home card. The control card allows one to set certain parameters (for example, toggling the security feature on and off), transfer data to analysis or other special-purpose cards, or simply view the data.

Surveyor is a startup application, which means that it automatically enters the survey loop after the operating system is booted up and the HyperCard program is loaded. It then runs independently and does not require any further assistance. Additional software will lock the drive, so an unauthorized person cannot eject the disk or reboot the computer from another disk. The respondent moves through the survey at his or her own pace. Upon completion, the survey starts over. When an authorized user exits the program, the system shuts down automatically.

The data entered by the respondent is stored in a separate data stack; it can be transferred into a microcomputer statistics package or uploaded to a mainframe for further analyses. The data may include both alphanumeric and numeric item responses from forced-choice, open-ended, or scaled formats. In addition, demographics as well as process data, such as number of tries, time elapsed, date, number of items responded to, or other item statis-

tics, may be included. The analysis functions allow one to perform simple statistical analyses such as frequency counts and averaging. The use of external functions permits more advanced analyses as well. The results of analyses can be printed out in reports.

Question, a Management Tool for Item-Level Questionnaire Data

The Question stack is designed as a data-management and analysis tool for instrument development and validation. Taking full advantage of the HyperCard environment, it allows one to store large amounts of information that can be related, sorted, and grouped according to user-selected criteria. HyperCard's search and retrieval capabilities make it easy to find and extract specific information. The computational capabilities of HyperTalk are utilized for mathematical transformations of the data. Export and print capabilities, which have been enhanced with special scripts, allow data to be easily extracted and stored for external use.

Despite its relative complexity, Question's interface is easy to use. All the required functions are made available through "smart" buttons. For example, lock/unlock text buttons and show/hide comment field buttons visually indicate the field's status, showing whether the text is locked or whether comments have been entered. The limitations imposed by the current card size have been partially overcome by the creation of special, small button icons. On-line help is provided for all the features.

Question serves two important functions. As a database, it allows one to keep different kinds of item data, such as the item's wording, the version of questionnaire used, or the dimension to which the item is assigned according to different keys, as well as selected item statistics and data from item analyses, factor analyses, etc., *on a single card*. We have found it very useful during the many phases of questionnaire development and validation to have all the relevant item-related information available at a glance. All the cards can be sorted and regrouped by data fields according to a variety of aspects. Find functions make it easy to find a particular word, text, or item number. Provisions for assigning values "remotely" facilitate the input task greatly. For example, items can either be coded individually or be assigned globally to a particular dimension on a given key. Each item may be coded on several dimension/key combinations (12×5 possibilities, in this particular case). All assignments can also be changed or deleted locally, card by card, or globally, for all cards or for a specified range of cards.

In addition to serving as a data-management tool, this stack provides for the further calculation of item-level data and the display of the results. A special print function (an improvement upon the report-printing capabilities provided by HyperCard) allows one to print selected cards and card information or to write information to an external text file, which can then be transferred to a word-processing program (e.g., for making tables) or to a data file for input to a statistical program for further analysis.

While a few special functions (such as checking for valid ranges, calculation of averages, etc.) are provided, other functions must be coded by the user by means of HyperTalk script editing. HyperTalk programming knowledge is required for one to take full advantage of Queston's potential.

Conclusions

HyperCard's toolbox approach allows experts as well as novices to mix and match their own ideas with what is given in existing HyperCard applications. This environment should have particular advantages and appeal for researchers who know a programming language, yet people without a formal background in computer programming can easily learn how to program with HyperTalk. Although the creation of complex stacks from scratch does require advanced knowledge of HyperCard and HyperTalk, less programming-proficient researchers can use existing stacks or elements of stacks and combine them with their own creations to achieve the desired functionality or form. The great variety of stacks available on public BBSs (e.g., GEnie or Compuserve) and file depositories on computer networks (e.g., MacServe on Bitnet, Sumex on Arpanet) can serve as a starting point for stackware developers. Alternatively, several software companies offer customized stackware design.

In this paper, we have tried to show that HyperCard offers the researcher or the instructor a unique degree of freedom and control. As one designs or uses HyperCard stacks through associative links, the HyperCard environment can mimic the way the mind works, thus encouraging creative problem-solving approaches.

The Surveyor stack demonstrates that the flexibility and openness of the HyperCard environment does not necessarily constitute a disadvantage for those who need to restrict its functionality, which in this case involves using

it with subjects as a survey or testing tool. Surveyor enables researchers to tailor a survey or testing application to their particular needs by taking advantage of HyperCard's user interface with its graphics and its sound capabilities, which can turn the interaction into a multimedia event. Voice, music, and graphics can be used, separately or in combination, to capture the respondent's attention and provide feedback.

The Queston stack is an example of a specialized data-management and analysis application that has been created to meet particular task requirements in the development of a psychological instrument. Although Queston can be used for other similar tasks, knowledge of programming in HyperTalk is a prerequisite for making it meet the individual needs of the particular researcher.

REFERENCES

- CONKLIN, J. (1987, September). Hypertext: An introduction and survey. *Computer Magazine*, pp. 17-41.
- CUSTER, L. (1987, December). Stacking the deck. *MacUser*, pp. 112-127.
- GOODMAN, D. (1987). *The complete HyperCard handbook*. New York: Bantam.
- GOODMAN, D. (1988). *Danny Goodman's HyperCard developer's guide*. New York: Bantam.
- GRAY, P. J. (1988). Microcomputers in evaluation. *Evaluation Practice*, 9(3), 47-53.
- HYPERCARD USER'S GUIDE (1987). Cupertino, CA: Apple Computer.
- KAHLER, C. (1988). *HyperCard power*. Reading, MA: Addison-Wesley.
- LASKY, R. D. (1988, June). Getting started with HyperCard. *MACazine*, pp. 36-40.
- RICE, M. (1988, September). Dr. HyperCard, PhD. *MACazine*, pp. 60-62.
- SCHAFER, D. (1988). *HyperTalk programming*. Indianapolis: Hayden.
- TESSLER, F. (1988, May). Build your first HyperCard application. *MacWorld*, pp. 140-143.
- WOLMAN, R. (1988, May/June). Is there a HyperCard in your future? *CBT Directions*, pp. 20-25.