AUTOAUTHORING PROCEDURAL TRAINING HYPERMEDIA

Lee W. Lacy, Dr. Richard Braby, and Captain Kevin Glass
Logicon Eagle Technology
Winter Park, Florida
U.S.A.F. Armstrong Laboratory
San Antonio, Texas

ABSTRACT

Developing interactive technical training materials currently consumes a large amount of resources to script, storyboard, and program lessons. Information is repeatedly used in presentations, demonstrations, exercises, tests, and job aids. A prototype system is being developed, named PATHS (Prototype Animated Training Hypermedia System), to demonstrate the possibility of substantially reducing the costs associated with developing and maintaining interactive courseware (ICW). PATHS transforms ICW development from instructional design and programming tasks into a data management task. Text, still graphics, audio, and video data describing a procedure are stored within PATHS according to a formal structure of knowledge. PATHS then uses these data elements to automatically generate (autoauthor) training hypermedia for procedure-oriented applications. Standard formats are used which incorporate proven learning strategies. Therefore, the developer does not need extensive expertise in training design or programming.

A subject matter expert enters the procedural information, including text, still graphics, and video clips. This knowledge base is stored by PATHS in a standard interservice format. PATHS then uses algorithms to automatically generate training hyperdocuments. These hyperdocuments include introduction material, demonstrations, practice exercises, tests, and job aids. Instructional developers can then refine and personalize the hyperdocument using the PATHS hypermedia authoring tool. The resulting instructional material, including digital video, can be distributed on CD-ROM for presentation on low-cost PC-based platforms equipped with Intel's Digital Video Interactive (DVI®) technology.

Trainees access the hyperdocuments using the PATHS browser. The PATHS browser includes icons which are used by trainees to navigate through the hypermedia. In a formal training program they are encouraged to follow the author's recommended "path".

ABOUT THE AUTHORS

Lee Lacy is a Software Engineer at Logicon Eagle Technology. He holds M.S. and B.S. degrees in Computer Science from the University of Central Florida. Dr. Richard Braby is a Senior Scientist and consultant to Logicon Eagle Technology. He holds M.A. and Ed.D. degrees in instructional design from Columbia University. Captain Kevin Glass is a Systems Computer Analyst at the U.S.A.F. Armstrong Laboratory's Human Resources Laboratory. He holds an M.S. degree in Aeronautical Sciences from Emory University and a B.S. degree in Criminology from Florida State University.

AUTOAUTHORING PROCEDURAL TRAINING HYPERMEDIA

Lee W. Lacy, Dr. Richard Braby, and Captain Kevin Glass

INTRODUCTION

Developing interactive technical training materials currently consumes a large amount of resources to script, storyboard, and program lessons. Specific information is used repeatedly, such as in presentations, *demonstrations, exercises, tests, and job* aids. The reuse of data suggests that the process can be automated to some degree, especially in regard to training highly procedural tasks.

To demonstrate the possibility of substantially reducing the costs associated with developing and maintaining interactive courseware (ICW), a prototype system is being developed, named PATHS (Prototype Animated Training Hypermedia System). PATHS transforms ICW development from instructional design and programming tasks into a data management task.

BACKGROUND

ICW development efficiency is heavily dependent on authoring tools, media, and data interchange. Courseware is typically developed according to the five phases of the Systems Approach to Training: Analysis, Design, Development, Implementation, and Evaluation. The most expensive phases are design and development. During the design stage, instructional strategies are selected, and instructional material is flowcharted. This flowcharted information is programmed during the development stage.

ICW Design

Selecting instructional strategies is a key element of ICW design. The instructional strategy is the plan for how the student will interact with the instructional material and learn to recall and apply what is presented. It includes the basic design for the types of displays and interaction that will accomplish this type of learning. Sound instructional strategy supports efficient learning.

Learning strategies to perform procedures from memory or with a job aid can take various forms. For example, prior to learning the steps of a procedure, an effective strategy is to have the trainee locate and identify the components used in the procedure.

ICW Development

Whenever a task involves repetition, it is a candidate for automation. This includes creating ICW to teach procedural tasks. For procedural tasks, the design and development process is repeated for a variety of teaching operations[1]. Steps are presented, demonstrated, used in student exercises and tests, and presented in job performance aids. All steps in a procedure can be presented using a consistent format. Many different equipment operation and maintenance procedures can be presented using the same formats. Preparing ICW to teach procedural tasks is an area where the author's operations are highly repetitive, and can therefore be automated.

Hypermedia

An understanding of the terms *hypertext* and *interactive multimedia* is helpful in understanding the characteristics of hypermedia. Hypermedia is the combination of hypertext and interactive multimedia. A hypertext is an interactive text and graphics document that contains extensive branching and cross-referencing. Interactive multimedia is the presentation of audio/visual information by a computer with extensive control exercised by the user.

Therefore, hypermedia refers to interactive documents which contain a variety of media including video and sound, and contain extensive branching and cross-referencing.[2] Because of the cross-referencing ability, hypermedia is often used for interactively accessing archived information. Instructional hypermedia is ICW presented using the hypermedia paradigm.

ICW is defined by MIL-STD-1379D[3] as "a computer program controlled instruction that relies on trainee input to determine the order and pace of instruction delivery. The trainee advances through the sequence of instructional events by making decisions and selections. The instruction branches according to the trainee's responses." Traditional ICW contains limited branching.

Hypermedia can be used to implement ICW and add new flexibility to the learning process. Hypermedia supports users with varying goals by allowing students to choose both content and depth of data. Users may include new students who must perform procedures, experienced operators that need refresher training, skilled operators of similar equipment that need cross training, supervisors that will never perform the procedure but need to know about it, and operators that need to maintain skills on rarely used procedures. With hypermedia, the material can be adapted to the special needs of any of the students.

In formal courses, the instructor can define a core *path* or hypertrail that all students are responsible for navigating. Hypermedia presentations are one solution to the problem of addressing diverse audiences because the mix of learning events accessed by different types of students to master the instructor's requirements will vary from student to student.

Hypermedia does not normally restrict a student to a single *path* through a document. The student's ability to freely navigate through a hyperdocument is called *learner control*. Learner control is thought to be more motivating for certain populations of students because it supports exploratory learning and supports students at a variety of levels.

However, the *web* nature of hypermedia can also cause problems for the student. Users of hyperdocuments get lost in "hyperspace"[4]. The solution to the disorientation problem includes providing support features such as *back on track* and hypermaps, and in properly structuring the hyperdocument. Extensive webs can also lead to problems with distraction. Instructor-specified recommended *paths* are one solution to keeping the student *on track* and focused.

DVI Technology

ICW has traditionally provided multimedia support through still graphics and motion video from an interactive videodisc (IVD) system. Technology is now available which provides ICW developers with video from a hard disk, CD-ROM, or network. This video support technology eliminates the need for the IVD system, reducing the delivery platform footprint, and increasing maintainability of the digital video files. Reduced video production costs should also result from using this technology, because digital files are easier to manipulate. A variety of systems and standards are becoming available to support digital video. However, the only commercial system currently capable of providing full-screen full-motion full-color (impression of 24-bit) video as a PC peripheral is Intel's Digital Video Interactive (DVI) technology[5]. DVI is used in PATHS to provide the hypermedia screen graphics, digital audio, and digital video clips.

Data Interchange Standards

One key solution to reducing the labor involved in creating ICW is the reuse of data. Technical manual data already exists for many systems for which ICW is being created. If the technical manual data can be interchanged from a digital source, it can be reused in the ICW development effort. These standards should be followed to achieve maximum efficiency in ICW development.

To interchange data, standards are required. The most inclusive set of standards regarding the interchange of digital technical information is the Department of Defense Computer-aided Acquisition and Logistics Support (CALS) initiative.[6] CALS is intended to eliminate duplication of data and improve processes.

A variety of specifications and standards are considered part of CALS. These items give specific guidance on interchanging various media. For example, MIL-M-28001 (SGML)[7], specifies markup instructions for text. SGML documents are described using Data Type Definitions (DTDs).

A key use of CALS involves the interservice Interactive Electronic Technical Manuals (IETM) effort. An example of an IETM is a portable PC-based job aiding system which is used on the flight line by aircraft engine mechanics. IETM information will be stored according to a database standard, (MIL-D-IETMDB)[8].

The IETM data base, often referred to as the *Content Data Model*, is used to store weapon system information using SGML so that it can be accessed by multiple applications. The IETM data base structure is described using the Hypermedia/Time-based Structuring Language (HyTime)[9], which is a draft international standard (ISO/IEC DIS 10744.1.1)[10] for specifying hypermedia DTDs.

The presentation of IETM database information is standardized by another IETM standard, MIL-M-GCSFUI[11]. The purpose of this document is to standardize the way in which users interact with information stored in an ITEM data base. ICW systems can benefit from these guidelines to ensure the interchange of data and the use of consistent user interface principles.

PATHS

The Prototype Animated Training Hypermedia System is currently being developed by Logicon Eagle Technology for the U.S. Air Force to experiment with solutions for the described problems. PATHS is currently being tested and debugged, with final delivery scheduled for February, 1993.

The major software components of PATHS (see Figure 1) are:

- Application Knowledge Base Authoring
- Instructional Hypermedia Autoauthoring
- Hypermedia Authoring
- Trainee Browsing

The major PATHS databases are:

- Application Knowledge Base
- Instructional Hypermedia
- Student Records

Application Knowledge Base Authoring

Subject-matter expertise concerning the procedures is stored in a database which we refer to as the Application Knowledge Base (AKB). The term *knowledge base* is used to suggest that knowledge concerning the performance of the procedures is stored. However, the term is not meant to suggest that any artificial intelligence or intelligent tutoring is taking place.

The AKB is a tree structure (see Figure 2) with a *system* forming the root node. A system has jobs; jobs have tasks; tasks have steps. Steps include an action, and responses which follow the performance of the action. Additionally,

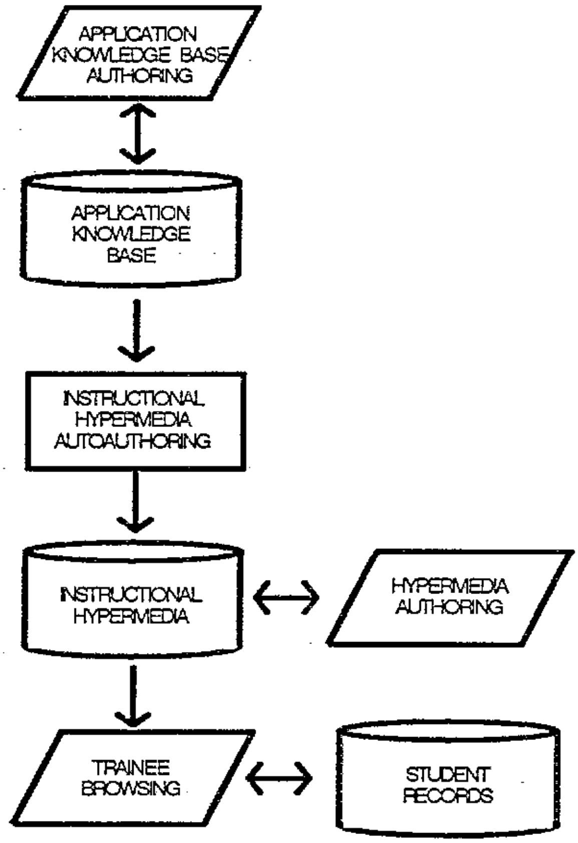

Figure 1. PATHS Software Components

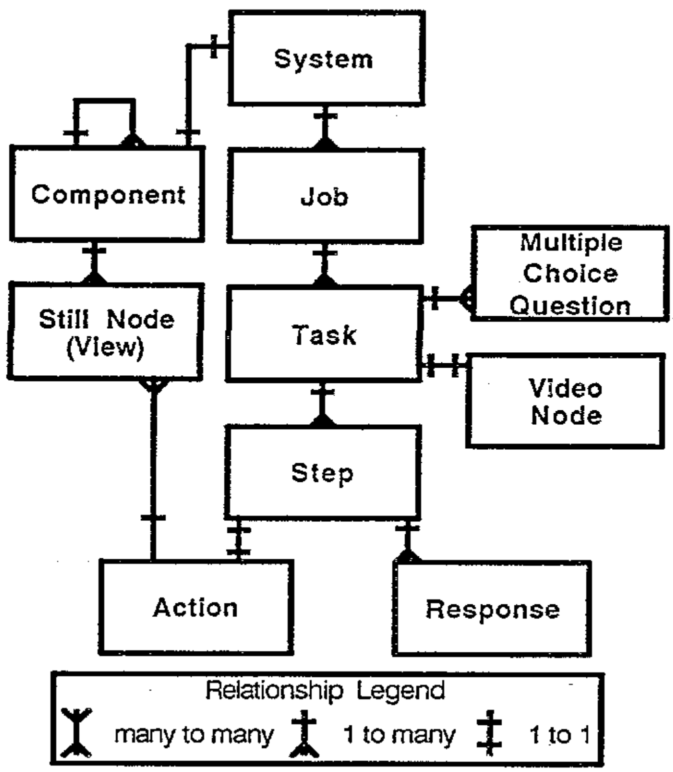

Figure 2. Application Knowledge Base Structure

a hierarchy of components (and their various graphical views) is associated with the system.

The AKB is specifically designed to support procedural training. Procedural training is more structured than cognitive task training.

Instructors or subject matter experts enter the AKB information into the computer using a series of menu, browser, and form screens. The browsers combine selection with presentation of data, thereby reducing the number of keystrokes required to navigate to a form screen. Form entries are made to enter text and file references. In the future, this process will be primarily limited to edits required to adjust the data automatically input from IETM databases.

IETM database information will be input in the future by translating IETM database files into PATHS AKB files. This translation process will be relatively simple because of the Data Type Declarations (DTDs) which describe the format of both files, and the use of translation software such as HyMinder™.

The majority of procedural information should be available from the IETM database files. However, there will be a need to manually add instructional information to the database.

The hyperdocument automatically generated by PATHS is also described by a DTD, which conforms to the HyTime standard. As hypermedia browser vendors begin to accept HyTime hyperdocuments, it should be possible for their systems to import PATHS hyperdocuments. By following this *open systems* approach, the end-user is not tied to any one particular hypermedia browser, and can select the one that best meets their requirements.

Instructional Hypermedia Autoauthoring

Once the AKB has been defined, ICW is automatically generated using instructional strategy pedagogy algorithms. The ICW generated is hierarchical hypermedia[12]. A variety of instructional treatments are supported in PATHS including: presentation, demonstration, practice, nomenclature drills, multiple choice test questions, and tested practice.

Each treatment is implemented with an algorithm which automatically formats and presents information selected from the AKB. Selected treatments for procedures are described below.

Presentation - This treatment provides an overview of the principles that underlie the system's operation.

Demonstration - Graphic details of each step in a procedure are accessed as either motion sequences or still images, selectable from a menu of steps.

Nomenclature Drill - The student is presented with directions for a nomenclature drill. This drill helps the student locate all the components referenced in the procedure. The student locates a named component by clicking on its location on an overview graphic.

Practice - The screen displays the text for the step and an overview graphic containing the points to be inspected or acted upon. The student clicks on the graphic to act on it, i.e., to set a switch to a new position. In this way the student practices performing all the steps in the procedure. A variant of practice is available which scores the student's ability to perform the procedure.

Multiple Choice Test Questions - Questions are used for academic practice and for testing the theory portion of instruction. The student is presented with a question, including a choice of possible answers. The question is generated by PATHS from question data components.

Treatments are autoauthored by placing the AKB into a predefined hypermedia *web*. The flow created by the links of the web is similar to the ICW design flowcharts. By repeatedly placing pieces of AKB into the web, the lesson is constructed. Because the resulting ICW was generated algorithmically, the screens are consistent. Random simulated system failures can be introduced to help keep the student motivated.

Hypermedia Authoring

The hyperdocument created from the autoauthoring can be manipulated by the instructional developer. The hypermedia authoring facility can be used to adjust the computer generated hyperdocument to make it more useful for the student. Computer generated screens can also be modified, to make the information more personable, or specific.

Basic editing capabilities are provided to the instructional developer to manipulate the hypermedia elements. For example, the text associated with a button link can be modified, or a new button can be added. However, only basic editing operations are provided.

Trainee Browsing

Trainees access the resulting hyperdocument through a browser. The browser provides traditional hypermedia navigation capabilities as well as instructional support features.

The top of the student's screen is used to present the title/name of the current mode of data. The majority of the student's screen is taken up by a data pane. The hypermedia material is presented in this pane. A row of navigation icons is arranged along the bottom of the screen.

The navigation icons are used to move through the hyperdocument. To move *right*, *left*, or *up* one screen, arrow icons are used. To *retrace* the progress through the screens, a backup icon is available. A hypermap icon can be used to display a graphical representation of the hyperdocument (table of contents outline). A bookmark icon allows the student to mark their position in the hyperdocument, and return to that spot later. An exit icon allows the student to exit the lesson.

Instructional support features are also provided through icons and through student record maintenance. The instructor can *blaze a trail* through the hyperdocument to specify a desired route for students. A student navigating through the hyperdocument can choose to follow the blazes, or explore on their own. If they deviate from the *path*, they can return by clicking on the *back on track* icon. A scrapbook facility allows the student to mark key screens for later reference. Student records are maintained which archive the student's progress through lessons, and their performance on tested activities.

The PATHS browser screen layout conforms to the MIL-U-GCSFUI standard for IETMs. Although instructional hypermedia differs from maintenance information, the systems share a common need to present information on demand to the military user in a consistent, user friendly manner.

One of the more powerful aspects of the PATHS browser is the use of *formatting-on-the-fly* to present the hyperdocument to the student. Rather than storing screen *frames*, the hyperdocument is stored as a database, and information from the database is viewed as needed. This reduces file size while increasing maintainability, by allowing the developer to modify data elements rather than *repaint* screens.

Applications

Two sample application lessons will be developed as part of the current contract. A medical evacuation litter carry lesson is being constructed. This lesson will highlight the advantage of motion video for training (hand-intensive) skills. Also, a calibration procedure is being constructed for the MiniOx III oxygen analyzer, which is used to monitor oxygen quantities at altitude. This lesson is an example of interaction with a control panel which has graphical and auditory responses.

The ideal applications for development on PATHS are applications which are highly procedural, can be photographed, and which have discrete tasks associated with them.

The economy of autoauthoring is highest when all instructional treatments are required to support different types of operators and maintainers of the system.

Equipment Configurations

There are two hardware/software configurations used to support the PATHS software. The developer's configuration (see figure 3) is used by subject matter experts and instructional developers to specify the AKB and prepare media. The student's configuration (see figure 4) is used by trainees to access the resulting hypermedia.

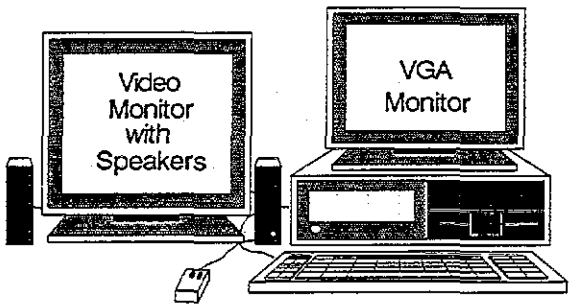

Figure 3. Developer Hardware Configuration

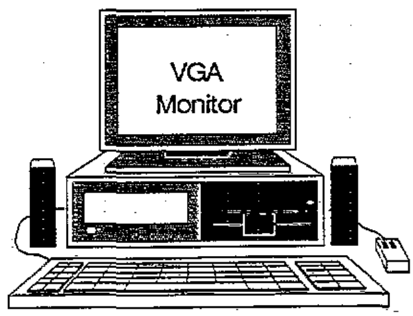

Figure 4. Trainee Hardware Configuration

The ICW developer's station consists of a 486 PC equipped with a DVI presentation board, a DVI capture board, a VGA monitor, and a video monitor. The VGA monitor is used for the author's user interface, while the student's view of the hypermedia is presented on the video monitor. The student station consists of a 486 PC equipped with a DVI presentation board and a VGA monitor.

Although a large hard disk was used in this effort, DVI was developed with CD-ROM in mind. DVI is timed to support the relatively slow access time of CD-ROMs. CD-ROMs can be created by a pressing facility for large quantities, or using a CD-ROM recorder for small quantities.[13] The number of lessons stored on a CD-ROM depends almost entirely on the extent to which video clips are used. A CD-ROM could not hold more than a single one-hour PATHS lesson if the entire hour was used to present motion video. However, a dozen lessons could be stored on a single CD-ROM disc if motion video was limited to 5 minutes or less per hour of instruction.

Additional equipment is required for media production. Although the DVI capture board allows for on-site digitizing of video, the real-time video (RTV) produced is of unacceptable quality for most applications. Instead, video should be professionally compressed by an Intel-licensed facility. However, high quality still video images can be digitized using the capture board or scanned using color scanners.

The current PATHS system utilizes the Series I DVI boards from Intel. Although, the Series II boards have been available for some time, the software which takes advantage of these boards has not been readily available. Media Control Interface (MCI) drivers, available for use with DVI under Microsoft® Windows™ 3.1, provide an application programming interface (API) for developing Windows 3.1 applications.[14]

SUMMARY

PATHS provides instructional developers with a new tool to efficiently generate instructional hypermedia for procedural training. The database used to autoauthor the ICW can be entered directly, or translated from an IETM DB. The ICW is presented to a trainee using a hypermedia browser, that can hopefully be replaced soon by a COTS hypermedia browser.

REFERENCES

[1]Mears, John E., Richard Braby, Joseph Hebert, A.J.G. Babu, J. Peter Kincaid, & Larry Maliza, (1989, October). *Organization of Technical Information for Electronic Delivery* (ARI Research Report 1543). Orlando, FL: U.S. Army Research Institute for the Behavioral and Social Sciences.

[2]Horn, Robert E., (1989) *Mapping Hypertext*, Lexington, MA: The Lexington Institute.

[3]MIL-STD-1379, (1990, December). Military Standard - Military Training Programs.

[4]Marchionini, Gary, (1988). "Hypermedia and Learning: Freedom and Chaos", *Educational Technology*, November.

[5]Fox, Edward A., (1989, July). "DVI - A Digital Multimedia Technology", *Communications of the ACM*, 32, 7.

[6]MIL-HDBK-59A, "Military Handbook - Department of Defense Computer-Aided Acquisition and Logistics Support (CALS) Program Implementation Guide".

[7]MIL-M-28001 "Markup Requirements and Generic Style Specifications for Electronic Printed Output and Exchange of Text (SGML)".

[8]MIL-D-IETMDB, (1991, April). "Data Base, Revisable: Interactive Electronic Technical Manuals, for the support of". (Draft).

[9]Newcomb, Steven R., Neill A. Kipp, & Victoria T. Newcomb, (1991, November). "The HyTime Hypermedia/Time-based Document Structuring Language", *Communications of the ACM*.

[10]"Information technology - Hypermedia/Time-based Structuring Language (HyTime)", ISO/IEC DIS 10744.1.1

[11]MIL-M-GCSFUI, (1991, April). "Manuals, Interactive Electronic Technical: General Content, Style, Format, and User-Interaction Requirements. (Draft)

[12]Jonassen, David H., (1986). "Hypertext Principles for Text and Courseware Design", *Educational Psychologist* 21(4).

[13]Bunzel, Mark J. & Sandra K. Morris, (1992) *Multimedia Applications Development Using DVI Technology*. New York: McGraw-Hill.

[14]Donovan, John W., (1991, December). "Intel/IBM's Audio Video Kernal", *Byte*.
