AD_____

REPORT NO 95-3

HYPERTEXT AND MULTIMEDIA FOR FUNCTIONAL ENHANCEMENT OF USARIEM MEDICAL HANDBOOKS AND BIOMEDICAL SIMULATION

U S ARMY RESEARCH INSTITUTE OF ENVIRONMENTAL MEDICINE

Natick, Massachusetts

DTIC
ELECTE
DEC 2 8 1994
F

DECEMBER 1994

Approved for public release: distribution unlimited.

UNITED STATES ARMY
MEDICAL RESEARCH & DEVELOPMENT COMMAND

19941223 023

DTIC QUALITY INSPECTED 1

TECHNICAL REPORT

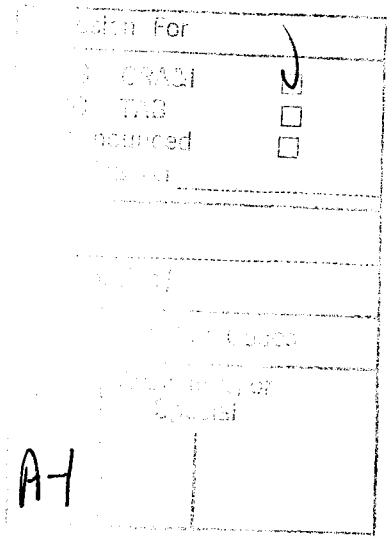
NO. T 95-#

HYPERTEXT AND MULTIMEDIA FOR FUNCTIONAL ENHANCEMENT OF USARIEM MEDICAL HANDBOOKS AND BIOMEDICAL SIMULATION SOFTWARE

by

Matthew J. Reardon, MAJ-MC

December 1994



U.S. Army Research Institute of Environmental Medicine
Natick, Massachusetts 01760-5007

December, 1994 Technical Report

Hypertext and Multimedia for Functional Enhancement of USARIEM Medical Handbooks and Biomedical Simulation Software

Reardon, Matthew, J., MAJ, MC

US Army Research Institute of Environmental Medicine
Natick, MA 01760-5007

US Army Medical Research and Materiel Command
Fort Detrick
Frederick, MD 21702-5007

Approved for Public Release; Distribution Unlimited

This technical report illustrates the feasibility of converting USARIEM medical handbooks, deployment manuals, and other technical publications into hypertext electronic documents. Techniques are elaborated for creating complex, but easily traversed, networks of hypertext or multimedia jumps, links, and pop-up windows. Five USARIEM documents are restructured as individual networks of hypertext nodes and links. It is also shown how such hypertext handbooks or publications can themselves be represented as hypertext nodes within a higher-level hypertext application. The higher-level hypertext application functions as the entry point for accessing the component hypertext documents. Additionally, a method and example are provided for integrating a hypertext document into a biomedical simulation software program. The example consists of incorporating the USARIEM medical handbook for preventing injury and illness during cold weather military operations, as an on-line interactive hypertext reference, into a dynamic lumped parameter cold digit simulation program.

Hypertext, Multimedia, Modeling, Biomedical simulation, Software development, Medical training, Distributed information systems, Frostbite, Heat stress, Altitude illness

UNCLASSIFIED UNCLASSIFIED UNCLASSIFIED NONE

DISCLAIMER

The view, opinions and/or findings in this report are those of the author and should not be construed as an official Department of the Army position, policy, or decision unless so designated by other official documentation.

Citations of commercial organizations or trade names in this report do not constitute an official Department of the Army endorsement or approval of the products or services of these organizations.

DTIC AVAILABILITY NOTICE

Qualified requestors may obtain copies of this report from Commander, Defense Technical Information Center (DTIC), Cameron Station, Alexandria, Virginia 22314.

Approved for public release. Distribution is unlimited.

CONTENTS

Page

LIST OF FIGURES iv
ACKNOWLEDGMENTS v
EXECUTIVE SUMMARY 1
INTRODUCTION 2
BACKGROUND 2
METHODS 9
RESULTS 12
USER'S GUIDE: USARIEM MEDICAL HYPERTEXT HANDBOOKS 13
A HYPERTEXT MEDICAL HANDBOOK AS A PROGRAM MODULE 25
DISCUSSION 29
CONCLUSIONS 36
REFERENCES 38
APPENDIX A 41
APPENDIX B 42
DISTRIBUTION 60

LIST OF FIGURES AND TABLES

Figures		Page
1	A conceptual framework for hypertext/multimedia product development.	10
2	Flowchart for the hypertext conversion process.	11
3	The USARIEM Medical Handbooks selection window.	14
4	Jumping to the heat illness medical handbook.	14
5	Pop-up table for work-rest cycles.	16
6	Topic or keyword search.	18
7	Appending notes to a handbook.	18
8	Examples of pop-ups in the altitude medical handbook.	20
9	Embedded pop-up windows in the nutrition handbook.	21
10	The USARIEM Type Protocol: some hypertext features.	23
11	Features of the lumped parameter cold digit software program.	26
12	Hypertext document management diagram.	37

Tables

1	A framework for concepts related to hypertext and multimedia development.	5
2	Characteristics of the USARIEM Medical Handbooks hypertext files.	13

ACKNOWLEDGMENTS

The author would like to thank Dr. Kent B. Pandolf (Director, Environmental Physiology and Medicine Directorate, USARIEM) and Dr. Richard R. Gonzalez (Chief, Biophysics and Biomedical Modeling Division, USARIEM) for their advice and suggestions. Additional editorial guidance and recommendations were obtained from Shari Hallas, USARIEM's technical editor. Mrs. Karen Pederson and Mrs. Edna Safran provided valuable secretarial and administrative support.

EXECUTIVE SUMMARY

This technical report describes how hypertext and multimedia authoring processes can be exploited to add greater functionality and new avenues of distribution for USARIEM technical and deployment publications. To demonstrate these additional functionalities several USARIEM environmental medicine handbooks were converted to hypertext. The most recent published USARIEM medical manuals for heat, cold, altitude and military nutrition, as well as the USARIEM Type Protocol (a compendium of safety limits and guidelines governing USARIEM researchers' use of volunteer soldier test subjects), were converted to computer-based interactive hypertext applications. These five electronic documents were also collated to form a comprehensive, stand-alone, medical handbooks hypertext product. The functionality of this compendium of hypertext USARIEM references was then compared to the 1 Oct. 93 version of the Navy's CD-ROM-based SEAL "Computer Assisted Medical Reference System" (SCAMRS). It was determined that the USARIEM medical handbooks hypertext application utilizes six times less computer memory per document, yet includes a wider range of dynamic interactive features, better visual resolution, superior navigational functionality, capability for keyword or topic searches, insertion of user notes, and a more efficient printing mechanism. Additionally, this report describes how the USARIEM cold weather medical hypertext handbook was incorporated into a computer program, with state-of-the-art graphical input/output interfaces, implementing a lumped parameter model of cold exposed digits. This cold digit algorithm predicts skin temperature as a function of time for selected locations along a representative digit. The successful integration of the hypertext cold weather medical manual into the cold digit simulator suggests that selected USARIEM technical reports and notes, recast as hypertext on-line references can be integrated as value-added features into complex USARIEM modeling and simulation products. Operational versions of these types of composite software could be distributed for use with field deployable computer systems. These would provide integrated computer-based environmental and deployment medical references interlinked with numerically based medical decision aids for use by unit surgeons, medics, or other military medical personnel. Finally, also discussed are design, development, and maintenance issues, challenges such as extending from hypertext to full multimedia, use of these types of products for computer-based training, resource requirements, portability of the application software across different types of computers and operating systems, as well as distribution and version control.

INTRODUCTION

A relatively recent computer publishing trend is the movement toward dynamic, interactive, hypertext and multimedia documents (Adam, 1993). The U.S. Army has, itself, begun to explore the development of hypertext and multimedia versions of generally distributed publications (e.g., Soldiers, 1994). This trend is accelerating with efforts to provide hypertext and multimedia for access on distributed information systems and networks, thereby facilitating extensive and rapid sharing of many different types of tightly integrated data. Such technology is being exploited to enhance collaborative document development and sharing within and among scientific or research organizations. It is also providing a new medium for training and updating operational and deployment military skills. The technology is available to allow military units to quickly download into handheld soldier computers (Texas Micro Public Relations Office, 1994) the latest USARIEM environmental and deployment medicine advice and recommendations for a specific area of the world. These interactive electronic documents would provide the soldier the latest updates on issues in environmental medicine and physiology as they apply to maximizing sustainability, health, and performance during deployments and field training. Such references could also incorporate built-in training and knowledge-base evaluation methods that automatically adapt to the type of user (e.g., physician, medic, infantryman).

This technical report provides an overview of a planning and development structure as well as an implementation process for converting scientific technical notes, reports, handbooks, and manuals into hypertext or multimedia documents. Additionally, a method and example are provided for merging a hypertext document into a numerically oriented biomedical simulation software program.

BACKGROUND

What is hypertext? The McGraw-Hill Dictionary of Scientific and Technical Terms (Parker, 1994) defines hypertext as "A data structure in which there are links between words, phrases, graphics, or other elements and associated information so that selection of a key object can activate a linkage and reveal the information." Hypertext is also a type of, or subset of, multimedia that is composed of links between and among nodes containing text, images, icons, as well as video and sound segments. Such a network of interlinked nodes results in complex electronic documents that elicit user interaction and provides the capability for nonlinear traversal of document contents. The user, rather than the computer or instructor, controls the rate, speed, and repetition of information. Hypertext nodes may also represent or encapsulate other

documents. A hypertext document, therefore, may serve as the gateway for accessing many other related documents. These may not necessarily be collocated on the same computer system. Links between document nodes may be defined by complex Internet addresses.

Hypertext and hypermedia documents have logical and physical structure (Shneiderman, 1993). The logical structure delineates relationships between general concepts and the various levels of detail associated with the concepts, branching patterns or response options, content segmentation scheme, and supporting database design. The logical structure often can be represented as a network of nodes and links. The physical structural components are the actual objects that comprise the nodes (e.g., windows, text, pictures, graphs, video, and sound), items that implement links (e.g., designated text or icons), as well as the supporting database files that contain node contents or that track user responses.

Multimedia's logical structure can be categorized as linear, hierarchical, nonlinear, or composite (Vaughan, 1993). The distinctions between these categories are listed below:

- Linear: components are linked in a nonbranching sequential manner.
- Hierarchical: components are linked in a structured tree-like branching pattern.
- Nonlinear: links between components form loops of various sizes and complexity.
- Composite: some sections are linear or hierarchical and other sections are nonlinear.

The widely held assumption that hypertext and multimedia paradigms are inherently effective is partly based on common experience with computers and videogames, as well as optimistic expectations that dynamic interactive documents are more interesting, easier to use, and result in more efficient learning than their nondynamic printed counterparts. Hypertext and hypermedia as instructional tools have also gained increasing acceptance among educators.

The *de novo* design and development of a hypertext or multimedia product often presents the developer with many decisions with respect to selecting the most effective content, structure, and dynamics. The developer must obtain or synthesize a guide or framework for making the best possible, consistent, and mutually supporting hypertext or multimedia authoring decisions. Such decisions will result in a high a priori likelihood of meeting the explicit and implicit user requirements that typically span a wide array of functional areas.

Park and Hannafin (1993) provide a taxonomy and references for general and specific psychological, pedagogical, and technological theories and concepts, which can serve as rational frameworks for the development of effective interactive multimedia (see Table 1). Application of such a taxonomy for structuring a hypertext or multimedia product can provide it with a cogent theoretical basis for predicting that an acceptable level of knowledge transfer and retention will likely occur.

Studies in the literature (e.g., Reisman, 1993) have demonstrated that self-paced multimedia computer-based instruction can result in statistically significant improvements in test scores for a variety of college-level subjects. Additional research, however is required to validate many of the assumed benefits of multimedia for the military setting. This is because most multimedia effectiveness studies have involved grade school, high school, and university students in civilian environments. These results may not be fully generalizable to the military setting. Soldiers have a variety of different educational backgrounds and learning styles and are employed in a wide range of operational situations throughout the world. The results of pedagogic research may eventually provide convincing criterion-based evidence for the specific advantages of different types of hypertext or multimedia products when targeted for particular military users and situations.

Although validation of the hypothesized improved learning efficiency and retention benefit imparted by hypertext or multimedia is at this time incomplete experts nonetheless indicate that hypertext is efficacious for large and/or complex documents. These are types of documents that users typically jump about in, reading small sections at a time rather than sequentially from front to back. Such documents, therefore, can be justifiably fragmented into interrelated but semiautonomous modules consistent with the manner in which they are actually used (Shneiderman, 1993). Professional reference manuals and handbooks often meet such criteria.

Converting a standard print version of a report or technical publication into a computer-based hypertext file is a basic technique for creating an interactive electronic document. Within the current Microsoft Windows 3.x operating system environment, for example, hypertext help files are typically activated from within a desktop application such as a word processor or spreadsheet program. The help hypertext files require the use of the winhelp.exe file. When the Windows operating system is installed, winhelp.exe is automatically installed in the Windows system directory as a standard hypertext viewing utility. This means that the executable engine to run a Windows hypertext file need not be distributed to clients whose computer systems also utilize the Microsoft Corporation (MS) Windows operating system.

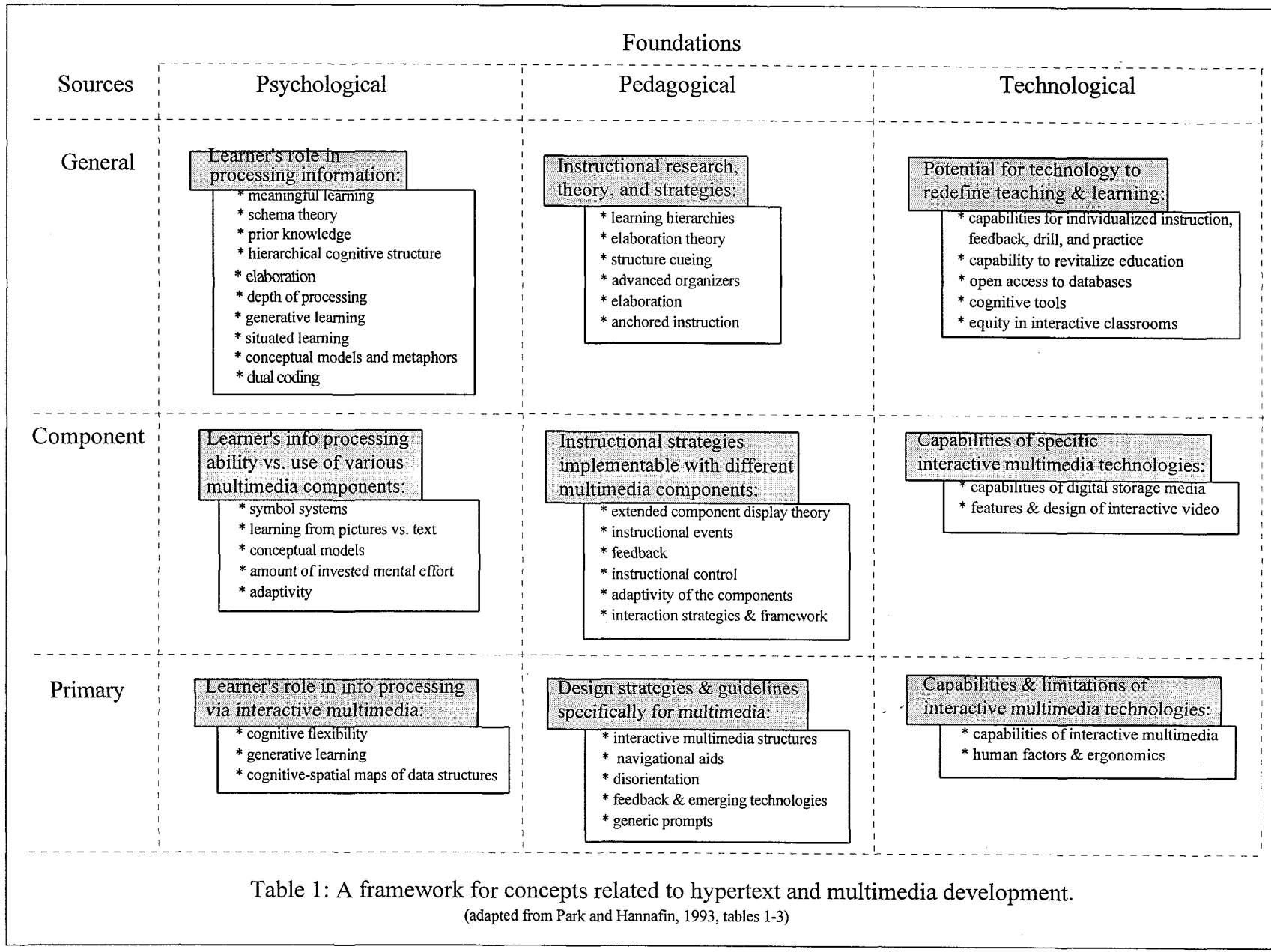


Table 1: A framework for concepts related to hypertext and multimedia development.

(adapted from Park and Hannafin, 1993, tables 1-3)

Hypertext allows a user to select and activate specially designated text, icons, or images (complete or segmented) with a point and click device such as a mouse or an equivalent sequence of keyboard entries. Such actions trigger prespecified hypertext events. These user-activated events typically include jumps to different sections of a document; pop-up windows containing such items as definitions, explanations or answers to questions; or voice or sound segments, images, or video.

Hypertext, for example, can impart dynamic capabilities to a table of contents (TOC). Each topic, subtopic, chart, table, or appendix in a TOC when designated as hypertext jumps can, when clicked on by the user's point and click device, directly transfer the user to the initial page of the selected section. Excessive scrolling through a lengthy document to locate a small section of interest can therefore be minimized. Obtrusive or optional sections can be tersely represented as icons or specially marked text that activate pop-up windows. The hypertext or multimedia document is thereby made more concise and more efficiently perused, yet retains the same or even more content than the original linear printed document.

Suppressed parts of a document represented as highlighted keywords or phrases, icons, or small (thumbnail) images are brought forward for viewing in pop-up windows. The pop-ups then revert automatically to their symbolic representation when the user types any key or clicks the appropriate mouse button. Question and answer formats, for example, can easily be constructed using these methods. The questions are formatted so that they activate answers or explanations in popups when clicked. This delays the presentation of answers, permitting the user to think of possible solutions prior to reading them as they appear in a pop-up window.

Charts and tables are also ideal objects for inclusion into hypertext or hypermedia pop-up windows. They can be hidden as highlighted words or icons that permit jumps or activation of pop-up windows containing the full image of the desired chart or table. Likewise, references to appendices can be designated as jumps. These techniques can be combined in a flexible manner to permit the design of a structured yet dynamic and easy-to-use cross-referencing system.

A standard hypertext document structure and authoring process is recommended. This simplifies development and use of hypertext and multimedia documents, as well as assures consistent, high quality, professional appearance. Standards decrease the likelihood of creating complex, inefficient, inconsistent, or insufficient structures that confuse, frustrate, or disorient the user. Improved cost and project management and control are additional areas that benefit from the implementation of design standards and product performance metrics.

An important requirement for establishing a cost-effective hypertext production system is a set of general or organization-specific guidelines, standards, and development tools that facilitate the rapid and efficient conversion of documents in standard formats to hypertext or multimedia documents.

To efficiently convert a set of documents to hypertext or multimedia, it is necessary to develop a document categorization scheme based on analysis of the structural and functional components and attributes of the different types of documents (e.g., technical reports, deployment manuals, journal articles, and regulations) that one's organization typically produces. There may be many structural or organizational features common to a majority of the paper-based document types. Common features, for example, often include such items as a TOC, section and subsection titles, lists of tables and figures, list of abbreviations, and an index. A standardized hypertext or multimedia conversion method should be established and recorded. This allows an efficient and consistent formatting of many documents of a similar type. Some hypertext authoring environments (e.g., *SmarText*, Lotus Development Corp.) provide semiautomation of the conversion process.

A TOC typically delineates the document's top-level structure and major sections and features, and is well suited as a starting point for hypertext conversion. For top-level navigational control, the document's TOC can be copied in its entirety, pasted into a hypertext editor, and reformatted as jumps to the corresponding sections. An alternative is to copy only the main topic headings in the TOC and hide subtopic lists as jumps to pop-up windows. This is desirable when portions of the TOC are lengthy or hierarchically complex.

The first page of each section for the major heading in the TOC can be collated to form the main browse group. A browse group is an ordered series or block of hypertext sections, modules, or pages. The user can page through a browse group by use of the forward and backward buttons located at the top of the standard hypertext document window. Each page of the primary or top-level browse group may also serve as the root for secondary and lower-level browse groups. Lower-level browse groups can be organized by copying the appropriate subheadings in the main TOC into the first page of the main browse group and formatting these lines as a list of jumps to the appropriate subsections of the browse group.

Navigating through a hypertext document can be accomplished, therefore, by use of browse groups in addition to jumps, popups, and topic search. Browse groups facilitate the viewing of sequential or related sections of a document. Jumps, on the other hand, permit either sequential or nonsequential browsing. Both methods should be harmonized to most effectively present the content and provide the user the

maximum amount of flexibility and navigational speed consistent with the objectives of the document. A hypertext document should be formatted so that the user can easily jump directly from any page in a subsection (subtopic) to the first page in the higher-level section (topic). The first page for a topic or section should typically contain a list of jumps to the constituent subtopics, as well as a capability for jumping back directly from topic or subtopic to the TOC. These measures also provide means for the user to rapidly reorient with respect to location within a complex hypertext or multimedia document.

Pop-up windows are useful for hiding less important or discretionary information in the background. This information can then be brought forward for viewing at the user's discretion or at the occurrence of a specified timed event. Content elements of a document that are particularly appropriate for inclusion in pop-up windows include definitions, explanatory or reference notes, footnotes, and images (e.g., scanned photos, graphs, tables). The use of transient pop-up windows tends to make a hypertext document appear more compact and permits more rapid traversal of the higher-level material. The user, therefore, is not required to scroll through unwanted detail when only a rapid overview of the contents is desired or a broad topic search is being conducted. Additionally, scrolling through a section of a document is more rapid when embedded graphical elements are hidden as highlighted words or as icons.

An index is another common organizational feature of many documents that is ideal for conversion to hypertext. An index can be copied into a hypertext editor and formatted as a series of jumps or popups. Individual index words or phrases can be converted to jumps or popups. Page numbers may be deleted or retained for cross-referencing to the hard copy version. With regard to pagination, it should be noted that the pages in the printed version of a document may not constitute natural (i.e., not the most appropriate) breaks for the hypertext version. The hypertext document is composed of sections and subsections displayed in scrolling windows.

To assist the user when scrolling through text in a hypertext scrolling window, hypertext authoring systems permit partitioning the top portion of the main document viewing window into a nonscrolling section. Items that are suitable for placement within the nonscrolling area include the section title, a jump to the first page of the current section or a pop-up jump list for the section or subsection contents, a jump to the main TOC (or title page), as well as additional navigational aids.

There are no standard rules that suggest an upper limit on the number of lines of text per hypertext window page. However, the number of lines should be sufficiently restricted so that extensive scrolling is avoided. Page lengths that require scrolling for more than two full window or screen lengths disrupt rapid scanning of section contents.

This can make finding key words more difficult, and hides the majority of text in a particular section from the viewer, particularly if the size of the viewing window is reduced. If a section is lengthy, it should be partitioned into an appropriate number of shorter hypertext pages. These can be sequentially collated to form a subtopic browse group.

METHODS

The four USARIEM medical and nutrition handbooks and the administrative research guidance document listed below were successfully converted individually into computer-based hypertext multimedia formats. These were then merged into a single hypertext operational medicine reference entitled "USARIEM Medical Handbooks."

TN 91-3 Heat Illness: A Handbook for Medical Officers (Burr, 1991)
TN 93-4 Medical Aspects of Cold Weather Operations: A Handbook for Medical Officers (Burr, 1993)
TN 94-2 Medical Problems in High Mountain Environments: A Handbook for Medical Officers (Cymerman and Rock, 1994)
TN 93-3 Nutrition for Health and Performance: Nutritional Guidance for Military Operations in Temperate and Extreme Environments (Thomas, et al., 1993)
USARIEM Type Protocol (USARIEM, 1993)

Figure 1 is a schematic that diagramatically represents a broad perspective of the hypertext conversion process. The larger and more complex a hypertext or multimedia project, the greater the advantage of establishing a structured development approach for the hypertext/multimedia project. Figure 2 is a diagram that delineates, in greater detail, the processes that can be utilized in developing the hypertext software modules described in this report.

The USARIEM medical handbooks listed above and the USARIEM Type Protocol were obtained from their authors in Word Perfect 5.1 format. These files were then converted to Word 2.0 for Windows (Microsoft Corp, Redmond, Wash.). This was done so that the Microsoft Word Rich Text Formatting (RTF) capability could be utilized to assist in the hypertext editing process. The majority of the hypertext editing, however, was accomplished with a hypertext authoring system (Herberger, 1993). Bitmap representations of graphs and tables in the various technical notes were obtained by use of a screen capture utility. A graphics and image processing

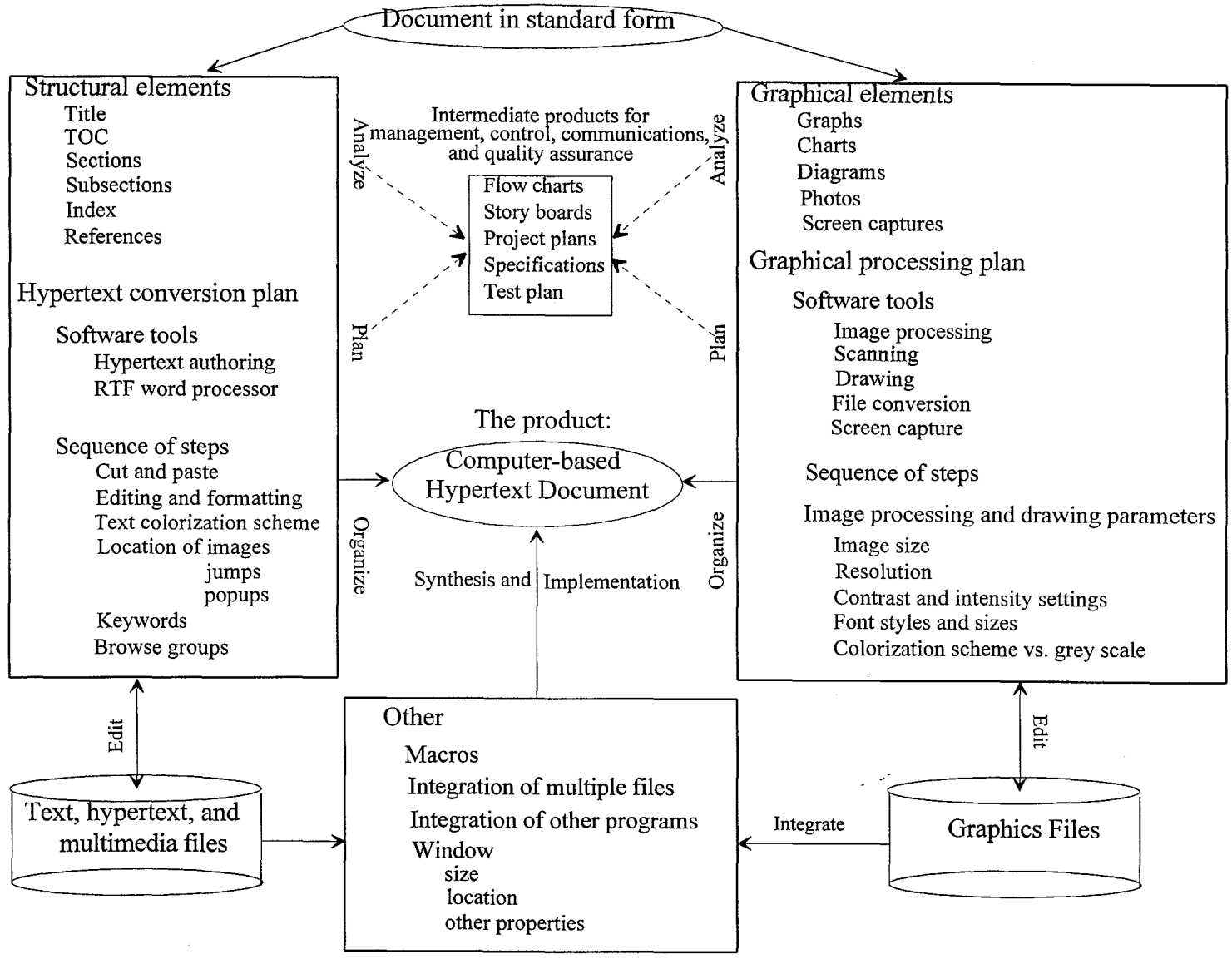


Figure 1: A conceptual framework for hypertext/multimedia product development.

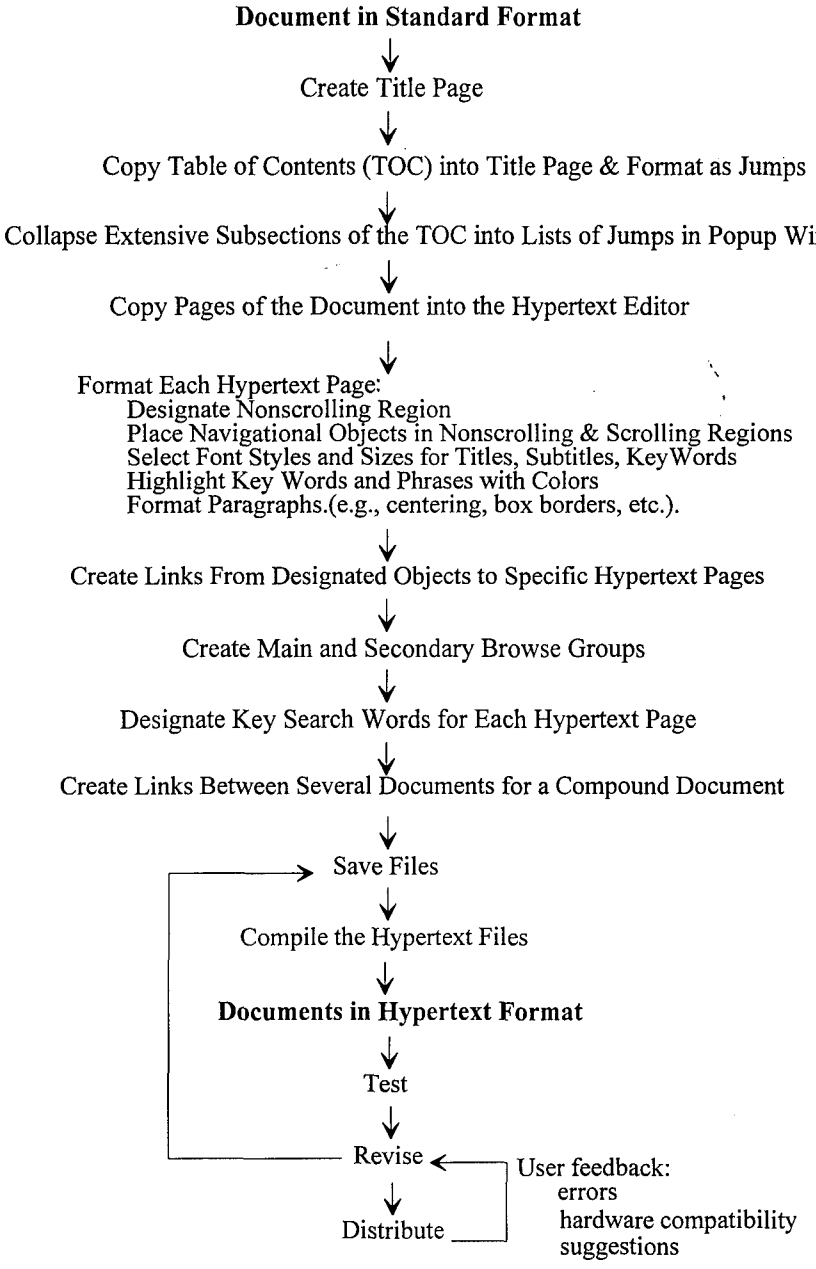


Figure 2: Flowchart for the hypertext conversion process.

application was used to optimize image properties such as size, resolution, contrast, and brightness.

In order to illustrate a method for integrating a hypertext USARIEM medical handbook into a predictive biophysical simulation, USARIEM's lumped parameter cold digit model (Shitzer, et al., 1990) was programmed in Visual Basic 3.0 (Microsoft Corp, Redmond, Wash.). A database of empirically determined thermal insulation and permeability values (Santee, et al., 1993) were linked to the glove types in the glove drop-down picklist located in the simulation's input window. The cold weather medical handbook was incorporated into the cold digit simulation program using a special subroutine call (Microsoft Corp., 1992) for that purpose.

RESULTS

Each of the four USARIEM environmental medicine medical handbooks listed in the previous section, as well as the USARIEM Type Protocol, were formatted into hypertext. They were then compiled so that they could be opened individually or accessed through a front-end interface that effectively bundled all five of these documents together into one comprehensive hypertext resource. This composite application, as well as the handbook selection window that the user encounters on activation of the application, were titled "USARIEM Medical Handbooks." This product is discussed in more detail in the User Guide section below. An additional section in this report describes how the hypertext version of USARIEM's cold weather medical handbook was integrated into a program that implements a predictive lumped parameter model for a cold exposed digit.

Table 2 below provides a quantitative characterization of the hypertext features in the USARIEM Medical Handbooks hypertext application. In the table, "Total" refers to the composite application, with the component files listed separately.

The entire application can be distributed on two 1.44 megabyte computer diskettes. The number of browse groups, keywords, jumps, popups, and bit graph images are indicators of the dynamic features added to the original documents, or a quantification of the added value. Formatting, editing, and inserting these additional features required approximately 15-30 minutes per page. Therefore, the entire 344-page application took approximately 129 hours to compose. On average, two hours per day were devoted to this project, hence hypertext conversion of the five USARIEM documents and creation of the encapsulating lead-in program (ENVMED) took about 65 days. Each document took 1.5 weeks to translate to the hypertext format using this work rate estimate. Because the hypertext conversion process is tedious, it

is unreasonable to expect an individual to be able to consistently expend more than four full hours per day on such a task. Hence, it can be estimated, that one full-time hypertext conversion person could convert three to six average-length technical reports or notes per month.

Help Files	Bytes	Pages	Browse Groups	Keywords	Jumps	Popups	Bitmap Graphics
Total	1,878,400	344	38	591	672	160	30
EnvMed	47,062	2	0	0	9	1	4
Alt	330,988	70	9	160	169	50	7
Cold	192,750	52	10	147	26	7	1
Heat	555,060	50	6	75	95	42	9
Rations	394,435	87	6	93	102	45	6
Type	358,105	83	7	116	271	15	3

Table 2: Characteristics of the USARIEM Medical Handbooks hypertext files.

USER'S GUIDE: USARIEM MEDICAL HYPERTEXT HANDBOOKS

Figure 3 illustrates the appearance of the main window containing graphic representations of the four medical handbooks that the user can select for review. These rectangular graphical areas are active in the sense that if the user clicks the mouse when the cursor is over the desired manual, the introductory screen of the selected manual with its TOC is displayed. Alternatively, the user may select the desired manual by clicking on the appropriate green underlined word in the top nonscrolling part of the window. These also activate jumps to the selected manuals. If the user clicks on the dotted-green underlined word "Instructions," a pop-up window is activated that provides basic information for setup and system configuration. This pop-up window suggests the optimum screen resolution, lists which files to copy if the application is transported to another PC, and provides other general guidance. When the user clicks the mouse or hits any key, this pop-up window disappears and the main window is brought forward again as the main focus.

At the top of the main window are pulldown menus for various options under "File," " Edit," "Bookmark," and "Help." Just below these menu options are four rectangular buttons labeled "Contents," "Search," "Back," and "History." Each window (except the popups) in this hypertext application has these two top lines of drop-down menu and button-activated options. Both lines are intrinsic components of the

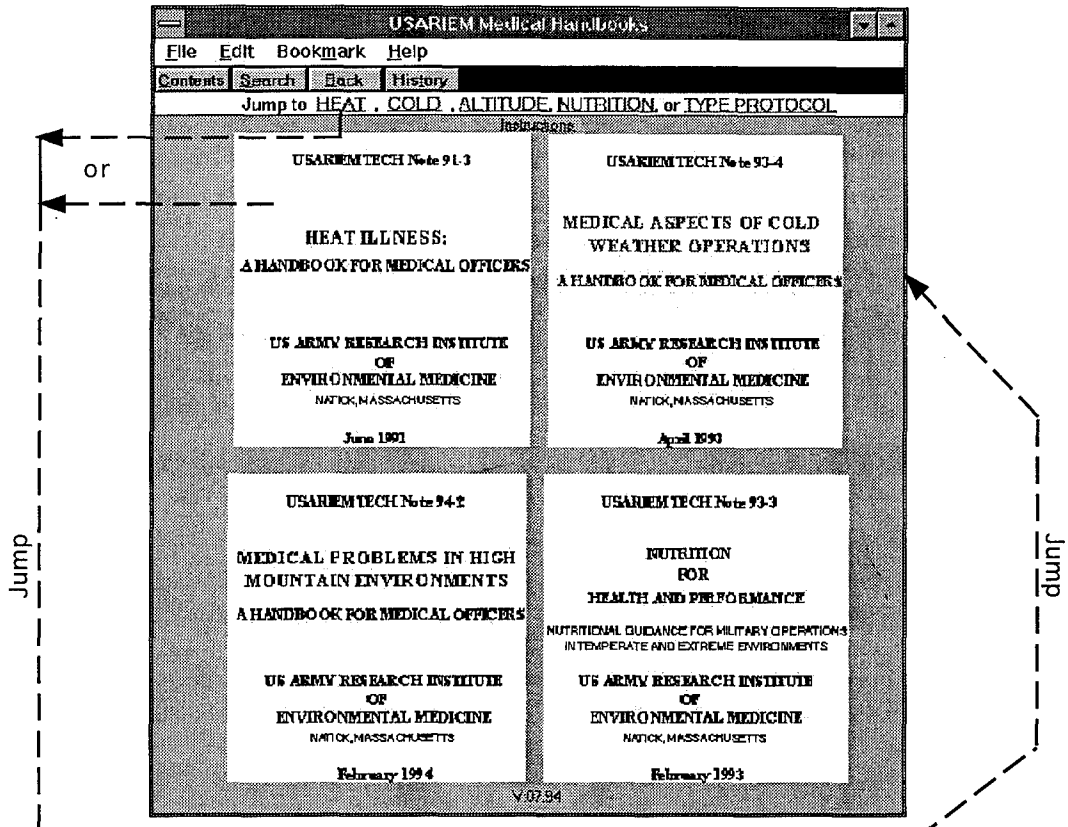


Figure 3: The USARIEM Medical Handbooks selection window.

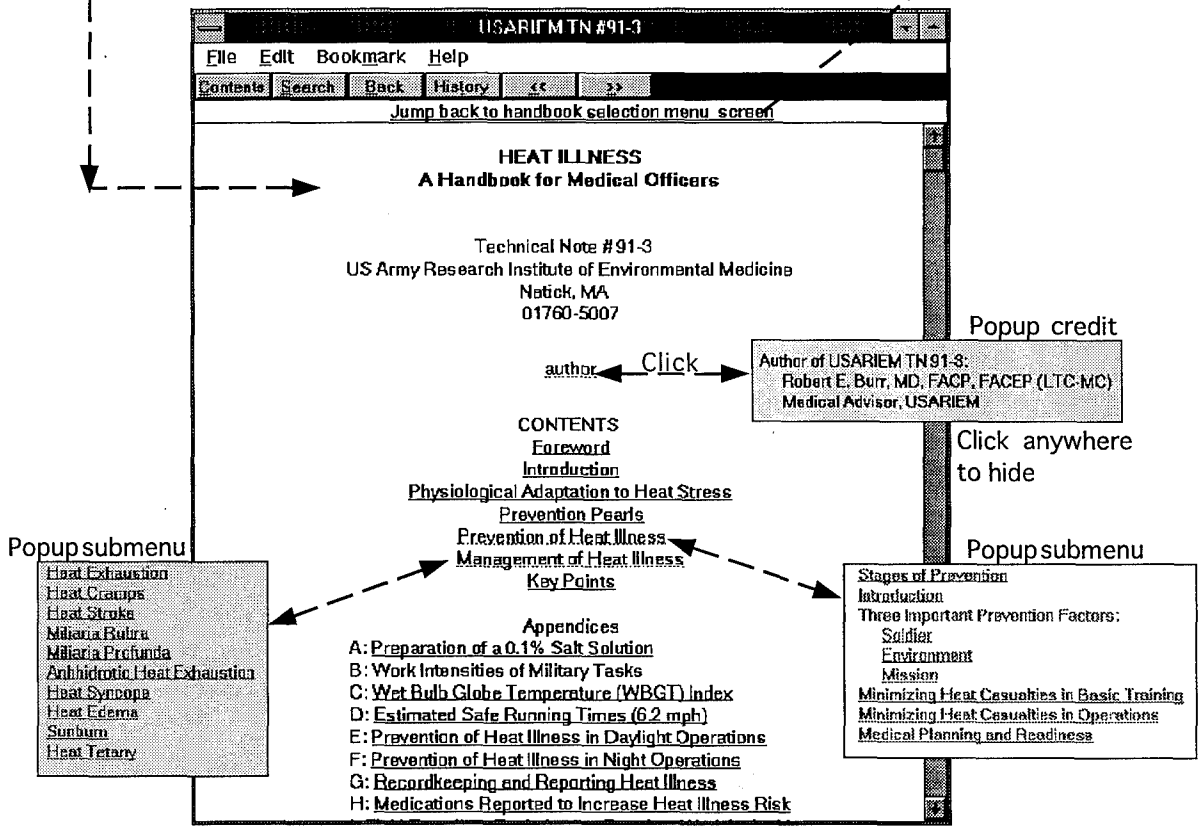


Figure 4: Jumping to the heat illness medical handbook.

Windows hypertext display engine. At the top right-hand corner of the typical display window are two arrow buttons. The down-pointing arrow button, when activated, reduces the application to an icon so that other background applications can be accessed with the point and click device. Double clicking on that icon then causes the hypertext manual to re-expand to its original size. The up-pointing arrow button causes the window to expand to full screen height and width. These also are common features in each window for the medical handbooks hypertext application.

The user may jump from the introductory screen (Figure 3) to any of the five component handbooks. For example, the user can jump to the heat illness handbook by either clicking on the green underlined word "HEAT" or by clicking the mouse button when the cursor tip overlies the picture of the heat illness handbook. When either of these actions occur, the initial page of the USARIEM technical note, "Heat Illness: A Handbook for Medical Officers," is displayed in the application's main window (Figure 4).

Note that the first page of the heat illness handbook contains its TOC. For consistency and easy access, the TOC is on the first screen of each of the five converted documents. The elements of the table of contents are displayed as underlined green text. Underlined green text is the standard indicator for text that can activate a jump.

Note that the sections or topics titled "Prevention of Heat Illness" and "Management of Heat Illness" are underlined with a dotted rather than a solid line. A dotted-green underline indicates text that, when clicked, causes a pop-up window to appear. Pop-up windows can also contain lists of jumps and can therefore be used to emulate drop-down menus. For the two sections just cited, for example, a pop-up window containing a list of jumps to the various subsections appears when the user clicks on the dotted-green underlined section title. Hiding subsections of a TOC in pop-up windows makes a lengthy TOC easier to read and puts greater visual emphasis on the main topics in the TOC.

The pop-up window that is activated when the user clicks on the topic "Management of Heat Illness" in the heat illness handbook TOC, for example, has a list of jumps to subsections on Heat Exhaustion, Heat Cramps, Heat Stroke, Miliaria Rubra, Miliaria Profunda, Anhidrotic Heat Exhaustion, Heat Syncope, Heat Edema, Sunburn, and Heat Tetany. The user may return to the main document selection page by clicking over the green underlined text, "Jump back to handbook selection menu screen," located in the top part of the nonscrolling area of the window on the first page of each of the handbooks. Each handbook uses this feature to allow the user to return to the main document selection window.

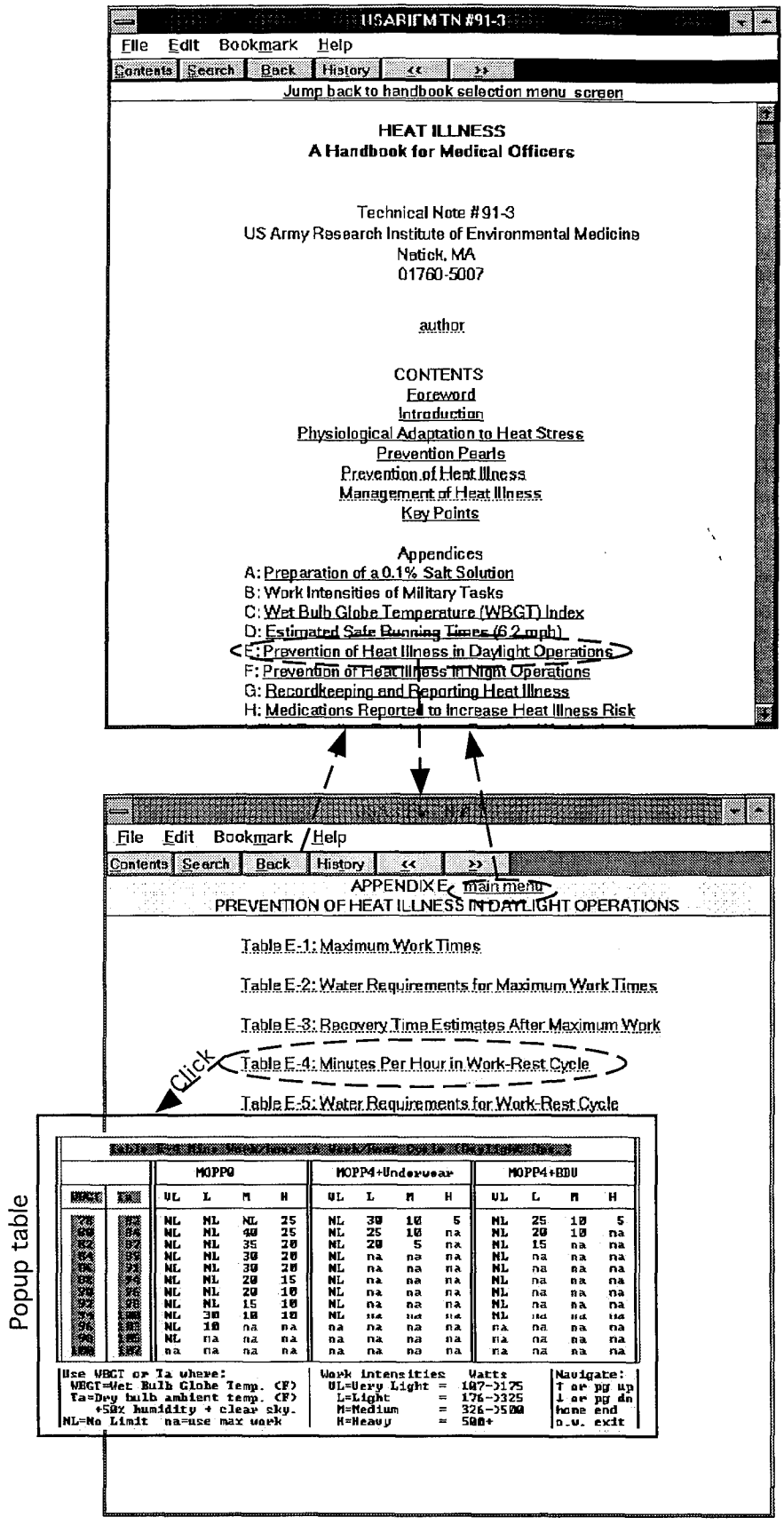


Figure 5: Pop-up table for work-rest cyles.

If the user clicks on "Prevention of Heat Illness" and then clicks on "The Soldier" within the subsequent pop-up, the initial section for this prevention topic appears in the main window. Within the nonscrolling areas at the top of the window are forward ">>" and back "<<" navigational buttons, the section title "Prevention of Heat Illness: THE SOLDIER," and underlined jump text. The jump text "other PM (Preventive Medicine) topics" causes a pop-up list of jumps to other PM topics in the section to appear. These features offer the user a variety of convenient options for jumping to different but related sections, or back to the top-level or main TOC.

The heat illness handbook has nine appendices accessible from the TOC in the title page. If the user, for example, clicks on the green underlined text, "Prevention of Heat Illness in Daylight Operations" (Appendix E), a jump is made to a list of pop-up tables (Figure 5). Clicking on the dotted-green underlined text, "Table E-4: Minutes Per Hour in Work-Rest Cycle" in the Appendix E window activates, as illustrated, a pop-up window containing a table of recommended work/rest cycles .

The user can backtrack to the initial page and TOC of the heat illness handbook by clicking on the green underlined text, "main menu," in the top part of the window. If the user does this, the title page of the heat illness handbook is brought forward. To get to the USARIEM Medical Handbooks selection the user clicks on the green underlined text, "Jump back to handbook selection menu screen," at the top of the Heat Illness Handbook screen. Having done this, the user is now in the window entitled "USARIEM Medical Handbooks."

To review "Medical Aspects of Cold Weather Operations: A Handbook for Medical Officers," the user may click over the picture of this manual or on the green underlined text, "COLD" near the top of the window. Figure 5 depicts the first page of the cold weather medical handbook. It has a hypertext structure similar to the heat illness handbook. Jumps, popups, and graphics are extensively employed. The various sections of the original document were decomposed into easily traversed interlinked blocks of hypertext.

The cold weather medical handbook, like the other handbooks, has built-in capabilities for keyword or topic searches. Lists of keywords and topics were developed during the hypertext formatting process. To find sections of a document that contain or relate to a key word or topic, the user clicks on the "Search" button at the top of the viewing window. This activates a secondary window (Figure 6) entitled "Search" that provides a scrollable list of key words. If a selected keyword such as "carbon monoxide" is clicked, a list of section titles related to that topic are displayed in an additional list box. In Figure 6, the topics related to carbon monoxide are "Alcohol," "Carbon Monoxide Poisoning," "CO, Fire, and Fume Hazards," and "Other Problems." A

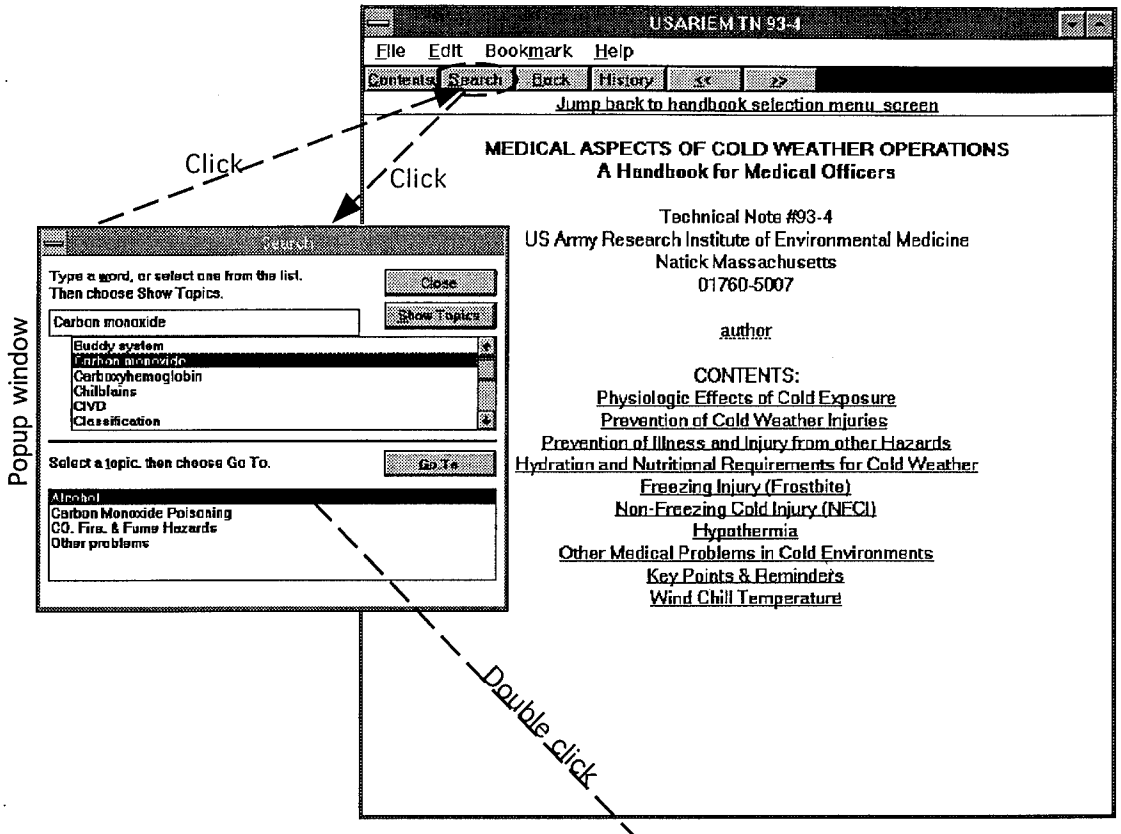


Figure 6: Topic or keyword search.

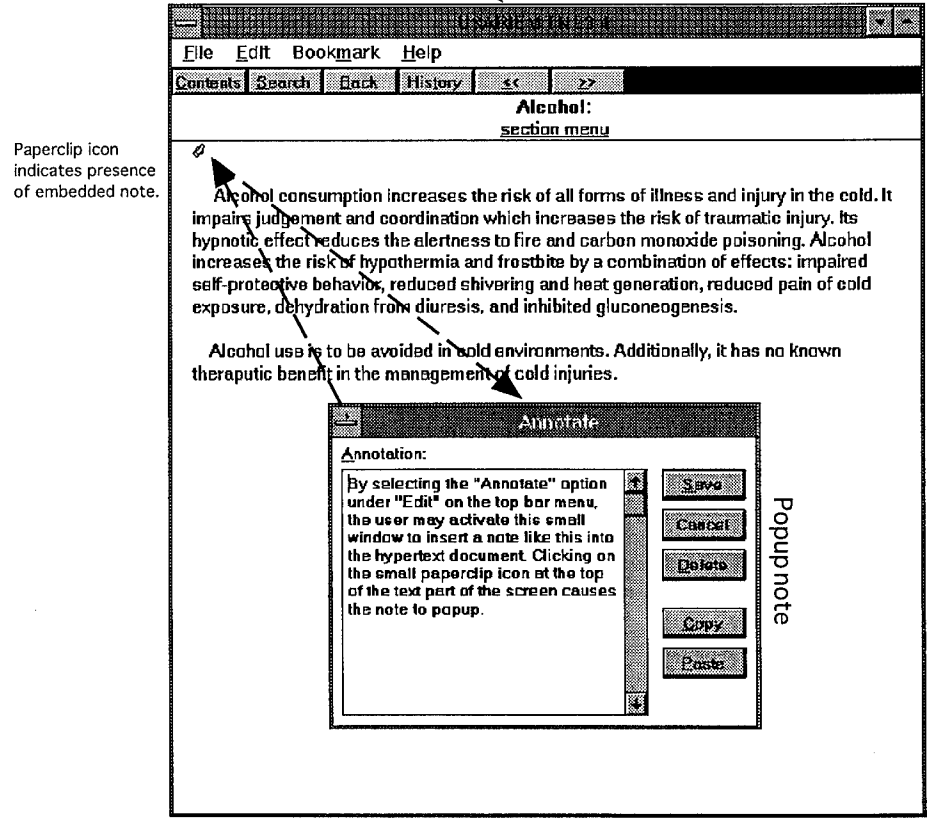


Figure 7: Appending notes to a handbook.

topic such as "Alcohol" (Figure 7) is retrieved by clicking the corresponding term in the displayed list. The search window then automatically disappears but can be reactivated at anytime, in the manner just described, in order to jump to a different topic or subtopic.

The original layout of the handbook, "Medical Problems in High Mountain Environments: A Handbook for Medical Officers," was particularly well suited for extensive use of pop-up jump lists and graphics in the hypertext version (Figure 8). For example, the lengthy TOC was visually compressed by hiding subsection contents as popups. Graphs were scanned from the original document and included as distinct pop-up windows for rapid viewing of the graphs directly from the TOC. These graphs, however, were also retained in their original text stream locations.

The key points for prevention of acute mountains sickness (AMS), high altitude pulmonary edema (HACE), and high altitude pulmonary edema (HAPE) were also rendered in pop-up format. These lists of recommendations are accessed separately from the TOC as well as similarly designated locations in the text stream of the various sections of the handbook. Pop-up lists of jumps to subtopics within a topic section were also utilized (not depicted). These pop-up lists were located in window frames in a manner that emulated pulldown menus. This provided a compact, familiar, and flexible format for rapidly navigating across many subtopics within a particular topic section.

Interspersed in the text of various sections of the altitude medicine handbook are references to the handbook's appendices. These references provided opportunities for designating direct jumps from the citation of a particular appendix to the actual appendix. Note that to immediately return to the location from where a jump was made, the user merely clicks on the button labeled "Back" in the top part of the appendix (or any) window. This allows for rapid and efficient paging back and forth between related sections. Again, the multitude of features that permit rapid and flexible traversal of topic sections is a significant advantage of a hypertext or multimedia document and directly contributes to its utility and acceptability.

From the main USARIEM Medical Handbooks screen the user may select to view "Nutrition for Health and Performance: Nutritional Guidance for Military Operations in Temperate and Extreme Environments" by clicking over the image of the handbook or on the green underlined word "NUTRITION," which appears near the top of the window. This hypertext handbook (Figure 9) has features similar to the other handbooks previously described.

A feature in this manual that was somewhat different, but amenable as an application for pop-up windows, were the question and answer sections. These

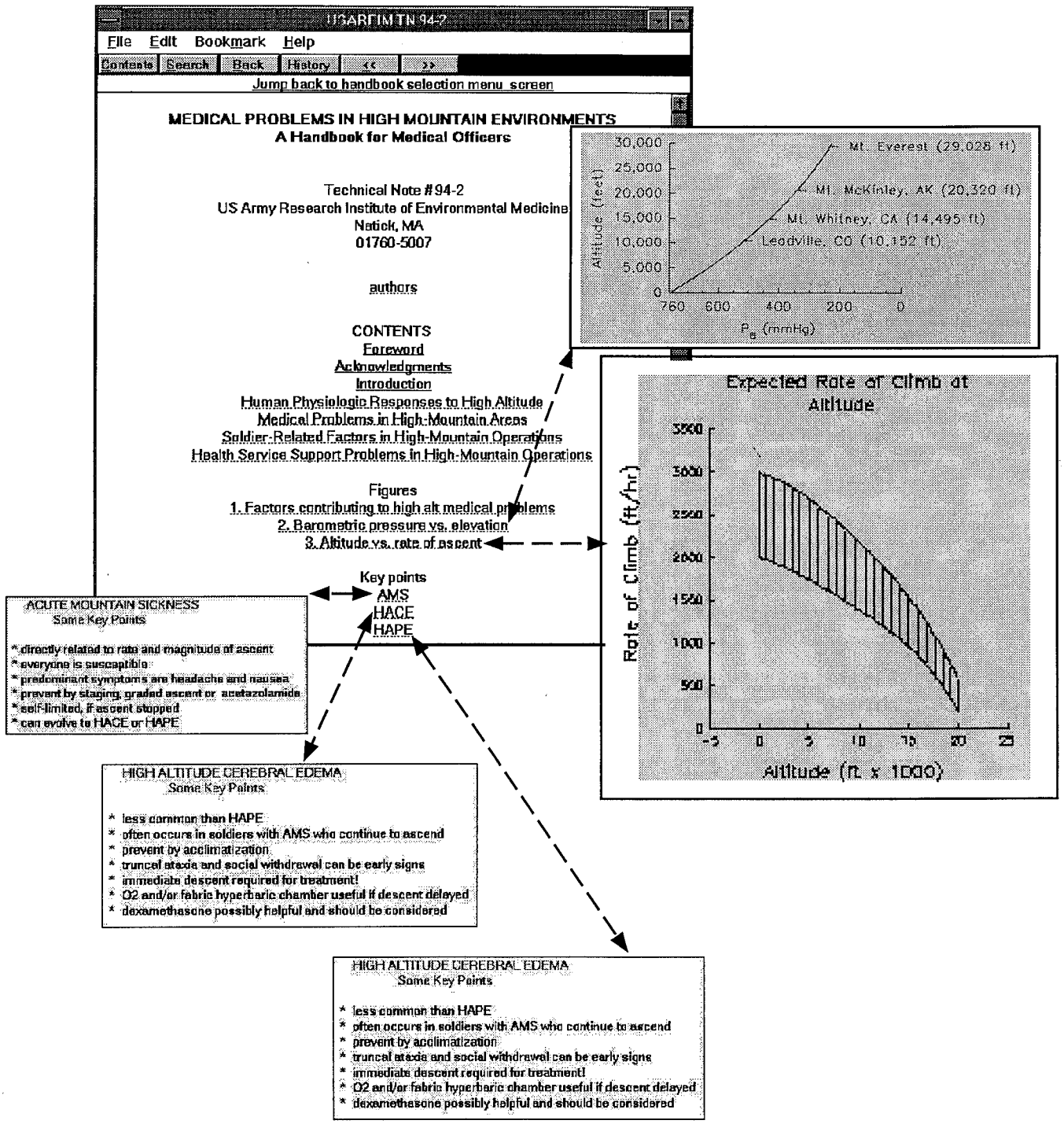


Figure 8: Example of pop-ups in the altitude medical handbook.

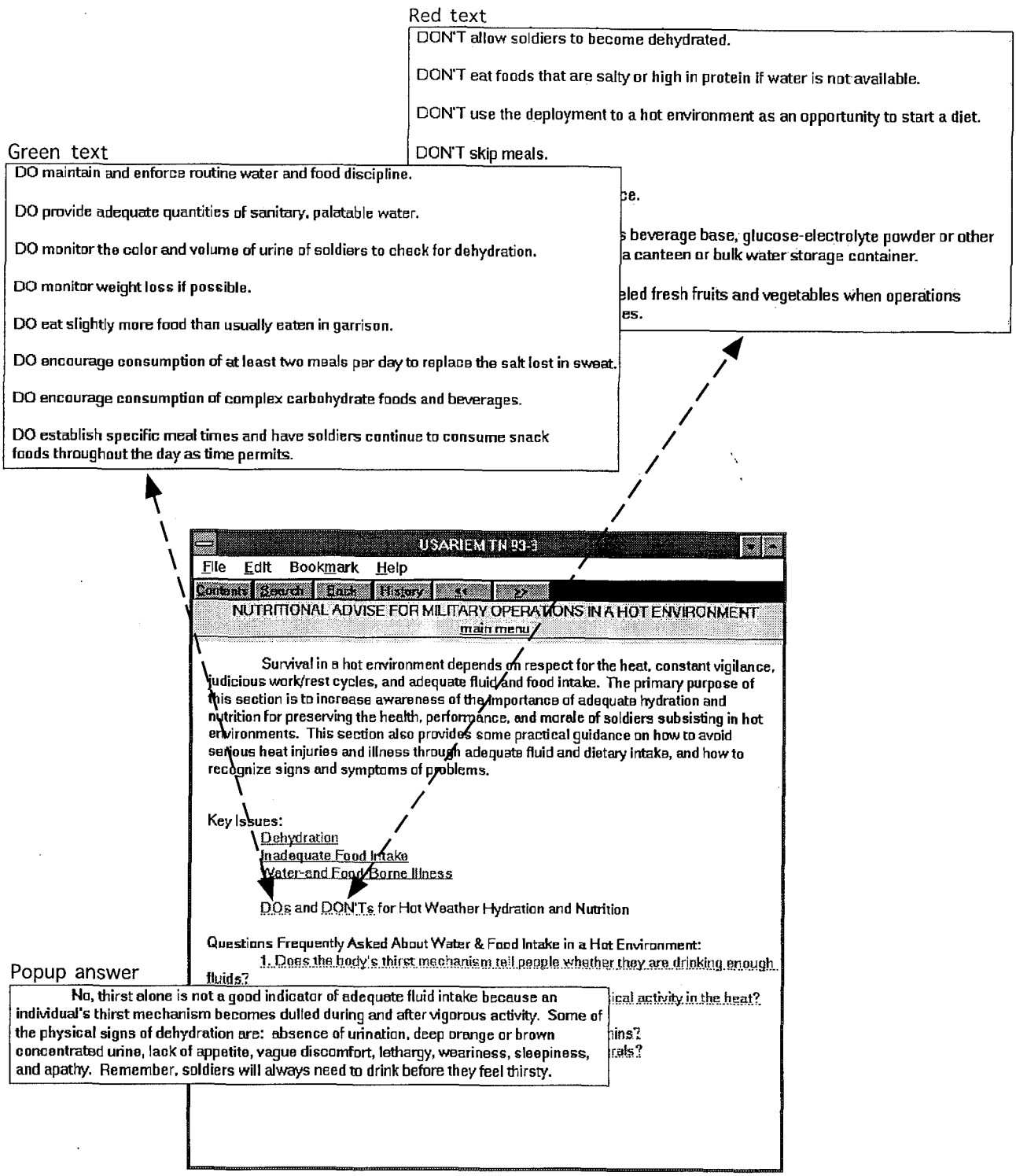


Figure 9: Embedded pop-up windows in the nutrition handbook.

sections were titled "QUESTIONS FREQUENTLY ASKED ABOUT ...." Such sections were hypertext formatted as a list of questions in green underlined text that, when clicked upon by the user, display the correct responses in pop-up windows. The user, therefore, is given sufficient time to consider the possible response(s) prior to actually reading them. This would be difficult to accomplish with the questions and answers presented in their original sequential, linear, format. The hypertext type of question and answer format makes the question and answer section visually more compact and encourages user interaction with the document in a natural type of transaction. This illustrates that hypertext provides flexible mechanisms for structuring interactive multiple-choice questions and answers. For example, questions can be formulated so that the possible answers are linked to popups that provide either the correct answer with or without additional links to supplementary information or an indication that the selected answer was incorrect with links to explanatory notes or to appropriate reference sections. This provides a framework for use of hypertext in computer-based training.

The final hypertext document to be discussed is the USARIEM Type Protocol. This can be selected for viewing by clicking on the green underlined text, "TYPE PROTOCOL," located near the top part of the main USARIEM Medical Handbooks window (Figure 3). This document is used at USARIEM as a reference for the guidance and limitations in the use of soldier volunteers in research studies imposed by the USARIEM commander, higher headquarters, and various governmental regulatory agencies. In order to maintain the selected size of the main window and to preserve the symmetry of the graphic elements, this manual was not displayed as an image. For versions of the hypertext manuals distributed to agencies or units outside USARIEM, this document would be deleted from the application by deleting its reference at the top of the main window.

The hypertext version of the USARIEM Type Protocol (Figure 10) contains similar features and functionality as described for the previous manuals. Figure 10 illustrates a representative sample of the pop-up or secondary windows that can be selectively activated by the user. The hypertext version of the USARIEM Type Protocol also demonstrates how a text color highlighting scheme is applied consistently across different but structurally related sections of a document. The list of experimental conditions and procedures (sections VII and VIII, respectively) is quite lengthy. All experimental conditions and procedures listed in the document have subsections labeled "Description," "Risks," and "Safeguards." The procedures have an additional subsection labeled "Example Volunteer Agreement Statement." In both the experimental conditions and procedures sections, the same color highlight pattern are used for the subsection labels. "Description" is blue, "Risks" red, "Safeguards" green, and "Example Volunteer Agreement Statement" maroon. Maintaining consistency in

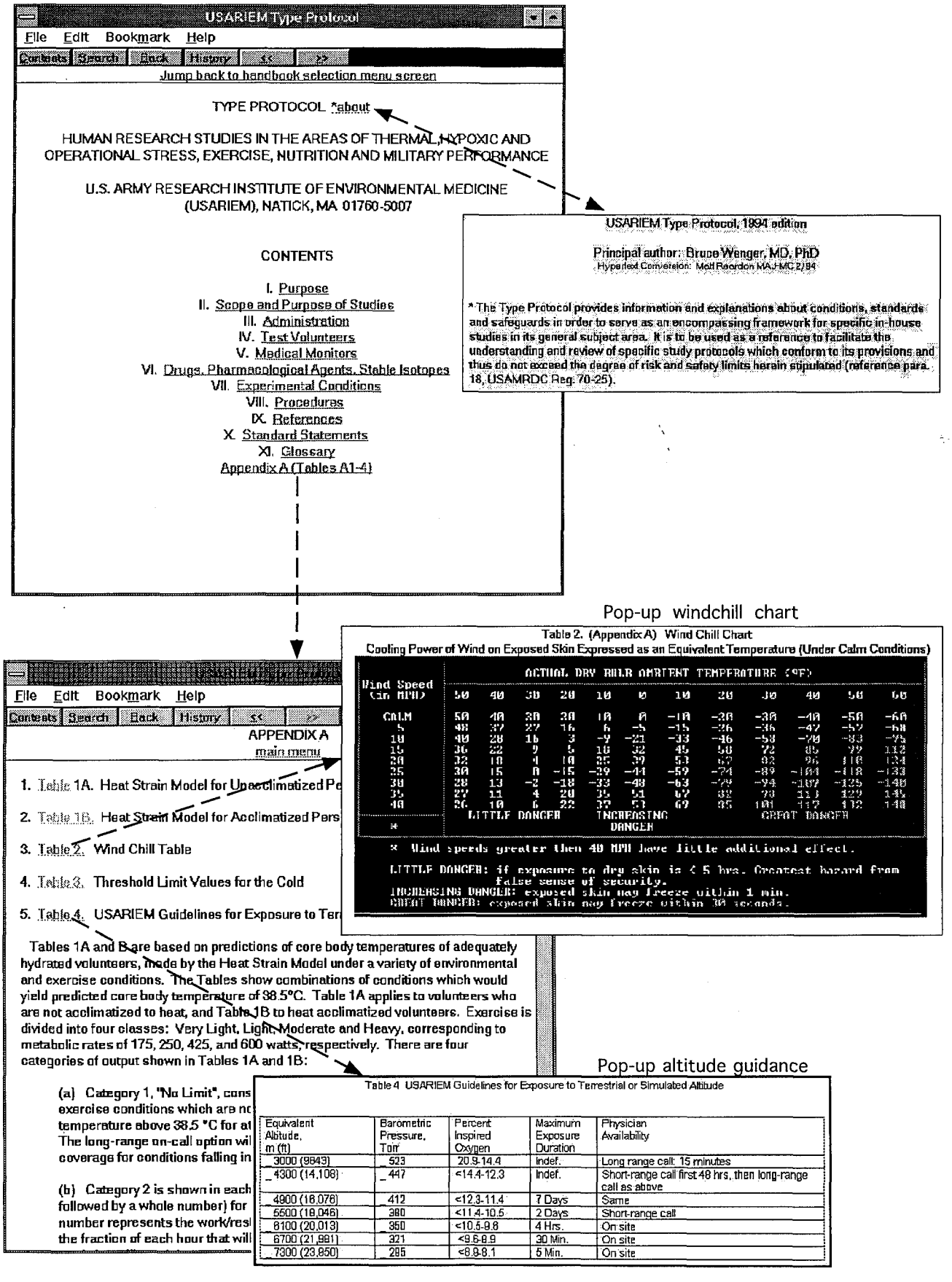


Figure 10: USARIEM Type Protocol: some hypertext features.

color highlighting allows the user to focus on a specific color when traversing a section to scan text for a specific subsection title (e.g., "Risks").

COMPARISONS TO THE U.S. NAVY'S SCAMRS

SCRAMS is an abbreviation for "SEAL Computer Assisted Medical Reference System" (Naval Special Warfare Command, 1993). This operational military field medicine reference system for laptop computers requires 131 megabytes of memory. It is distributed via compact optical ROM disk (CD-ROM). The contents are intended to provide a SEAL medic approximately 377 military medical and deployment manuals, handbooks, reports, and instructions as well as articles from a variety of military and civilian medical journals. These are collated into 37 primary subject areas.

Whereas a few of these references are stored as text files, most of the references in SCAMRS were optically scanned and are stored as TIFF image files (a common and flexible computer file standard for storing images). These image files for each reference are in separate subdirectories. Although the image files are compressed, the values for the compression rations could not be ascertained.

The SCAMRS program allows paging, scrolling, zooming in and out, panning, printing, and rudimentary background/foreground color changes. References stored as TIFF files are printed as images even though they may consist primarily of text. This is somewhat problematic since, with current technology, printing images of text takes significantly longer than printing an equivalent text file.

The version of SCAMRS that was evaluated does not have interactive hypertext or multimedia features. Keyword searches are not possible. The use of color highlighting and emphasis is nonexistent. The resolution is poor unless a high degree of zoom is utilized, which then prevents the entire width of a page from being viewed. The user cannot jump from one location to another within a document and, therefore, is forced to do extensive scrolling through a sequence of image files. Many references do not contain TOC so the user must laboriously scroll through the multiple images that comprise a document to determine its contents.

The SCAMRS software utilizes the relatively large storage capabilities of the CD-ROM. The medical references contained in SCAMRS were obtained by image scanning without optical character recognition (OCR) post-processing. The documents in SCAMRS, therefore, are stored entirely as images even though their contents are primarily text. This results in files that are very large, relative to text files containing comparable information content. The images of text also cannot be searched or printed as easily, or efficiently, as nonimage (standard) text. The hypertext medical handbooks

application, the principal product described in this report, demonstrates that by use of hypertext or multimedia authoring tools, USARIEM could readily supersede SCAMRS with a more advanced and functional, dynamic, interactive compendium of operationally oriented environmental medicine and physiology references.

Storing what are primarily text files as images results in inefficient use of data storage resources and prevents text-based enhancements and word searches. On the other hand, the methods used in the implementation of SCAMRS were undoubtedly less expensive than for a multimedia or hypertext version. Multimedia and, perhaps to a lesser extent, hypertext applications require a complex sequence of procedures and expensive tools for editing and specially formatting text, scanning graphs and figures, image processing for optimizing graphical elements, creating links and keyword lists, and integration of the different components and elements. SCAMRS was designed to provide an expedient and useful compendium of medical references for use by Navy and Marine medics in operationally remote setting. However, the USARIEM Medical Handbook hypertext application described in this report serves as an example of how SCAMRS could be substantially enhanced by reimplementing it with a hypertext or multimedia format.

A HYPERTEXT MEDICAL HANDBOOK AS A PROGRAM MODULE

This section illustrates a method for merging a hypertext version of a USARIEM technical note into a numerically oriented environmental medicine and physiology simulation software program. As a specific example, the USARIEM medical handbook for cold weather operations was incorporated into a program that implements a lumped parameter thermoregulatory finger model as described by Shitzer, et al. (1990).

The cold digit simulation program predicts superficial finger temperatures along the length of a digit (e.g., finger) from base to tip at uniformly spaced intervals. These digit temperatures are a function of duration of cold weather exposure as well as a variety of other input parameters. As illustrated in Figure 11, the USARIEM cold weather medical manual for medical officers is accessed within the framework of the cold digit program by a click of the mouse button on the top bar menu option "Med Info." When activated, the title page with the TOC for the cold weather medical handbook appears in a secondary window. The user then can navigate through the various sections of the hypertext cold weather medical handbook as previously described.

The links between the cold digit program and various sections in the cold weather medical hypertext handbook can be extended considerably in future versions.

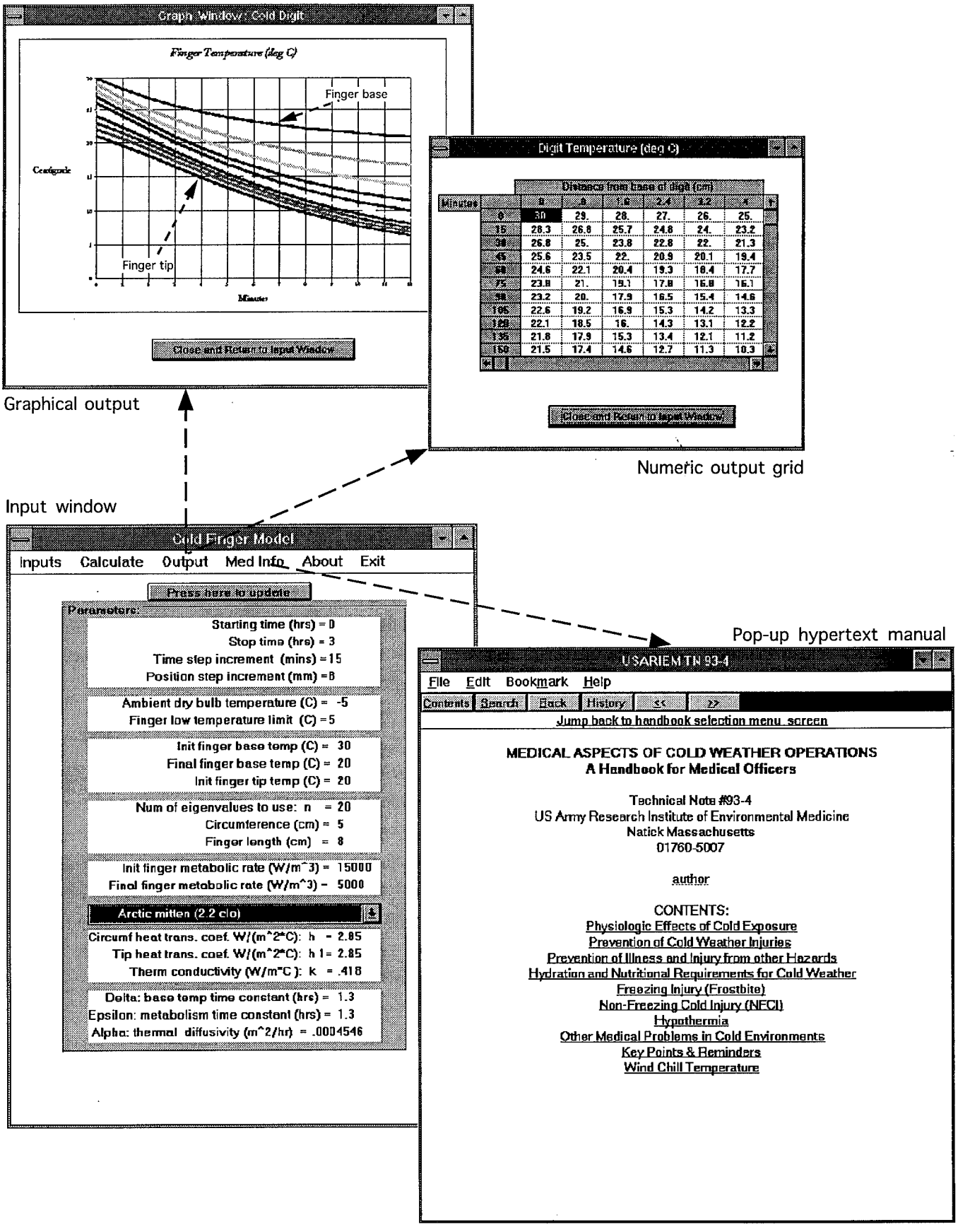


Figure 11: Features of the lumped parameter cold digit software program.

For example, context sensitive links could be created between the numerous elements or objects in the input or output windows and related sections or subsections of the supporting hypertext documents. This was not implemented in the current version but would not be technically difficult using available programming and editing tools. Likewise, a hypertext version of the USARIEM technical report describing the mathematics, the geometrical model of the digit, and assumptions used in formulating the cold digit algorithm (Shitzer, et al., 1990) could be created and linked to the main program. This would be appropriate for versions of the program customized for use in the scientific consulting or educational environment where the user might periodically want to review the origins and technical details of the numerical model.

Appendix A provides a hierarchical list of the principal interface objects and functions in the cold digit program. There are graphical user interface (GUI) objects, local subroutines that manage user initiated events, and global subroutines. The GUI objects have numerous additional data elements and methods (such as Show, Hide, etc.) that are not depicted in Appendix A. The Visual Basic 3.0 (Microsoft Windows 3.x version) code for the cold digit program is included in Appendix B. The cold digit program is a prototype written and compiled to demonstrate the feasibility of linking USARIEM predictive numerical algorithms and technical references. Some expedients were employed during the coding; therefore, there are a few subroutines that are probably not optimally structured (e.g. with respect to maximizing modularity).

The subroutine calc_eigenvals uses a bisection root-finding method to determine the specified number of eigenvalues. Shitzer, et al. (1990) used the somewhat more complicated Newton-Raphson root-finding algorithm. Both methods, however, result in identical eigenvalues out to at least six significant figures. The output temperature profile is not sensitive to such small differences in eigenvalues. The bisection root-finding method in this case is very efficient because the main term in the eigenfunction is periodic, and the range where successive eigenvalues can be found is predictable (see Boyce and Deprima, 1986; figure 11.1, page 598 for a graph from a similar eigenvalue example).

As part of the output, this version of the cold exposed digit program generates a table, or more precisely, a scrolling grid of superficial finger temperatures (Figure 10). The columns in this grid represent equi-spaced locations from base to tip along a representative digit. The rows represent successive time intervals (specified by the user in the input window). The finger temperature data can also be depicted in graphical form as illustrated in Figure 10. Successive curves in the graph represent the spatially distributed finger temperature profiles for equal time increments.

For research, development, and evaluation, the detailed and lengthy input and output depicted in this version of the cold digit model is desirable. For example, extensive input/output data may be essential for parameter sensitivity analysis. For military field use, however, input and output typically should be simplified to capture and provide concise and operationally practical data. A field version of this cold digit program could be structured to generate, for example, a table of times required to reach a specified threshold fingertip temperature, (e.g., 5°C [41°F]), as a function of a range of ambient conditions and different types of gloves. Such cold endurance time tables would be advantageous for disseminating easy to use doctrinal guidance for the maximum safe duration for outdoor sentry duty in order to prevent frostbite of the digits.

The predicted time required for the most vulnerable part of a finger (i.e., the tip) to decrease to the lowest permissible temperature can be calculated directly. This results in efficient processing, generating one, or only a few, calculations per set of inputs. That is, numerical calculations of all the spatially and temporally distributed digit temperatures from the initial time until the threshold low digit temperature is reached are not necessary. Verbose temperature profiles may be preferred for research or training purposes, but can obfuscate the operationally useful points. Computer-generated guidance for soldiers in the field, therefore, should be terse, relevant, and easy to comprehend.

This also illustrates that a program's input and output requirements for different types of users can vary considerably. Different versions of a cold digit program may be required. These may vary with respect to program structure, number of subroutines, data structures, and input/output formats. In order to have a rational basis for deciding on whether single or multiple versions of a program are required, a careful needs assessment for the major types of prospective users must precede or occur early in the design and development process. This will increase the likelihood that the final product has adequate scalability and will be accepted and considered useful by disparate types of users.

The current version of the cold digit program can also be modified or extended to allow prediction of toe temperatures for a foot exposed to a cold environment. Instead of selecting a glove, the user would select a sock and boot system from a dropdown or scrolling pick list (see Endrusick, et al., 1993, for biophysical characterization of boots in cold-wet environments).

DISCUSSION

MULTIMEDIA PROJECT MANAGEMENT

The hypertext and cold exposed digit temperature program described in this report successfully demonstrated the feasibility of converting USARIEM medical manuals to hypertext format and merging them with a numerical biomedical simulation. The software design effort was somewhat extemporaneous, partly because there were no previously documented USARIEM applications in this domain to serve as references. It is therefore worth reviewing a software project management scheme that can be applied to the development of hypertext or multimedia products.

Hypertext or multimedia product planning, development, distribution, and maintenance should be governed by a structured planning, management, and control paradigm. Skytte (1994) provides a summary of the principal steps for small software systems design. In this scheme, the project management process starts with an accurate determination of the explicit as well as the unstated but implied user requirements. These performance requirements are analyzed to enumerate, categorize, and prioritize the explicit, implied, and supporting functional, physical, and interface requirements that the hypertext, multimedia, and/or biomedical simulation programs must or should provide. A system-level plan and architecture is then formulated that allocates related groups of functions to a hierarchy of subsystems such as subprograms, subroutines, or procedures. The subsystems can often be developed and tested using a parallel bottom-up approach. This type of software development structure facilitates efficient implementation of all the requirements and allocates testing, documentation, and other support tasks to the development teams most familiar with the manner in which the particular subsystems were implemented.

The hypertext or multimedia product design should also be modular. Other software development issues such as portability and scalability should be early and high-priority design considerations. Portability means that software can be recompiled for use computers with different operating systems, with minimal, if any, modifications to the code. Portable implementations for different operating systems should result in the same functionality and interface characteristics. Scalability can have a variety of meanings but, for the purposes of this discussion, is restricted to mean that a software product's functionality can be easily expanded or reduced by inserting or removing discrete software modules, functions, or subroutines. Scalablility simplifies customization and the integration of enhancements.

System and subsystem design specifications are preferably delineated in the predesign phase. Specifications must be consistent with user requirements.

Prototyping and simulation should be utilized early in the product development cycle to determine the achievability of the processing, numeric, and interface specifications and user requirements. Elements that are required but difficult to implement are "high risk" elements and must be scrutinized for more detailed management, control, and budgeting. The use of consultants with special expertise in complex high-risk areas may be necessary.

In parallel with the system and detailed design efforts, a life-cyle model is developed for the proposed hypertext or multimedia product(s). The life cycle model includes processes such as marketing research, needs assessment, design, development, testing, distribution, training and technical support, upgrade and enhancement (product improvement) schedules, and obsolescence and transition to next generation products. A continuous program of research, development, and evaluation is required to develop a cache, or library, of software components that implement state-of-the-art advances in multimedia or hypertext techniques. These advanced components can then be efficiently mobilized and integrated into product upgrades or used as building blocks for an entirely new system or product line.

The delineation of program or system-level specifications and their allocation to objects, data structures, functions, or subroutines (a top-to-bottom process) is followed by development of detail design specifications and component integration plans (a bottom-to-top process). The software development plans and specifications are implemented and the components methodically integrated into a complete product. Test plans must be developed early, simultaneously with the system, subsystem, and detailed design phases. Testing is implemented for each subsystem by the subsystem developers as well as in a bottom-up sequence by the higher-level subsystem integrators. Early prototype testing is recommended to verify that explicit and implied software performance specifications can in fact be satisfied. The software components undergo a planned disciplined regimen of testing and optimization to minimize any differences in detected vs. desired performance.

Software validation studies or user follow-ups can determine the extent to which specifications and functional requirements are actually achieved in the locations where the product will be used. Appropriate follow-up studies and evaluations can help determine whether the distributed product is actually accepted, useful, reliable, and easy to use.

It is commonly recognized that, even within the same basic class of computer, there may be considerable differences between individual computers. These differences occur with regard to the brands or settings of hardware components (including brand and resolution of display units), previously installed software, as well as

differences in the contents of system setup files. In other words, software that works or that has acceptable interface performance on the software developer's computer may, for a variety of reasons, exhibit unacceptable behavior or performance on a different but very similar machine using the same operating system. It is highly recommended, therefore, that a software product be tested on a representative variety of different computer brands and configurations. If possible, a survey of the types and configurations of computers at user sites should be obtained. If this cannot be obtained in detail, then a reasonable estimate should be developed based on the systems that were noted at the client's site and what are likely to be the most commonly installed software suits and hardware configurations.

Software developers who do not test their products across a representative sample of platforms are likely to be unpleasantly surprised by dissatisfied customers (perhaps including an acquisition manager) who refuse to use the product because it is perceived to be functionally or visually deficient on their particular computer. The developer should not completely rely on user feedback to estimate the possible extent of program bugs, software-system incompatibilities, software-software conflicts, or other difficulties. Active solicitation and measurement of user satisfaction should also be employed prior to full-scale distribution. The use of survey instruments such as questionnaires or telephone interviews may be appropriate. Mailed or telephonic surveys should involve a random but representative sample of the potential users. On-site validation of acceptable user-program interaction is often preferable if the number of users is small and not widely distributed geographically. In the software industry, these types of quality assurance techniques are often encompassed in the "Beta" prerelease testing process.

HUMAN FACTORS

Software, as with mechanical systems, must be designed with user capabilities and limitations in mind. The capabilities and limitations may be cognitive, physiological, physical, or a combination of these categories. These may vary with age, educational levels, prior experience, and a host of other factors. Physical factors to consider include color blindness, variations in visual and auditory acuity, and detection thresholds.

Hypertext and multimedia designers and developers should be particularly aware of the limiting cognitive factors that govern the quality of user interaction with hypertext or multimedia products. The characteristics of short-term memory, for example, should always be considered because it can easily be overwhelmed if too much material in different formats is presented too quickly or without adequately designed transitions.

Studies of cognition have demonstrated that most individuals can retain and process, in short-term memory, only three to seven items at a time of recently acquired information. These items of information are typically retained in short-term memory for only 15 to 30 seconds and then either shifted into intermediate or long-term memory or forgotten. Information presented in visual and text formats usually results in better long-term memory retention rates (Fisher, 1994).

Human factor design principles for hypertext that follow from the well-established limits of short-term memory include the following: The efficiency of data transmission to the user decreases if there are more than seven items of information per window, pop-up, or bullet list. If the user must divert attention to another task (e.g., multimedia in a kiosk situation) for more than 15 to 30 seconds, timing rather than quantity of information should take precedence (particularly important for transitioning to a short video segment or sound bite). The use of graphics and video segments assist in retaining information about the structure of complex objects, interrelationships between objects, and processes. Visual elements decrease the burden of constructing mental images from the expository narrative. Preference for use of visual elements in hypertext and multimedia may permit a redirection of mental effort to analysis and memory retention.

When designing hypertext or multimedia applications a factor to consider is that rapid or frequent transitions of the senses, and associated mental processes, from one type of medium to another is difficult and distracting (Fisher, 1994). A brief (1-2 seconds) transition or warning is often recommended when shifting the user's main focus from one medium to another. Transition warnings are intended to alert and prepare the user for a change in the way information is going to be presented. A transition cue may consist, for example, of a short pleasant sound or message as a prelude into a video segment. The shorter a video segment the more useful an aural cue is likely to be. Audio cues and segments, however, should be used judiciously and be controllable by the user, since it may be considered irritating and obtrusive by some individuals. Transitional cues may lose their initial relevance after a user becomes familiar with the transitions, tempo, and options in a hypertext or multimedia document.

Park and Hannafin (1993) discuss twenty cognitive principles and their implications that should be considered when designing multimedia or hypertext applications. Their individual applicability and priority, however, is undoubtedly a function of the intended use of the application (e.g., technical hypertext reference vs. multimedia for self-paced instruction), as well as the characteristics of the intended user groups.

Studies such as those by Carlson (1991) have demonstrated that educational interactive multimedia (IMM) products are more effective when their instructional format matches the primary learning characteristic of the user. Factor analysis in Carlson's study of IMM training structures for adult learners, for example, identified two principal learning types: deductive and inductive. The deductive learners placed greater value on definitions, explicitly stated goals, and step-by-step procedures. Inductive learners preferred a sequence of examples, time to engage in analysis and abstraction in order to formulate hypotheses, and feedback to determine how close they were to the expert responses. When a study group was exposed, in a crossover study design, to both deductive and deductive versions of an IMM training product, mismatching of user learning type and IMM instructional format led to increased user dissatisfaction and frustration.

Learning theory and knowledge of cognitive processes and limitations should be drawn upon in developing a suitable instructional design process and model for linking and structuring dynamic multimedia components to form a context and user specific electronic document that will support effective self-paced, computer-based training (Jonassen, 1991). If an electronic document can be represented as a network of nodes and links, an appropriate instructional design and authoring process will provide a rational framework for selecting, for example, the number, contents, and media for the nodes; type and structure of links; and desirable patterns for feedback and interaction.

There is a large body of literature that delineates constructs and research results in learning, cognition, and instructional systems design (see Richey, 1986). These can be drawn upon to assist in deciding on the most appropriate contents, structure, and dynamic contraints for a hypertext and multimedia product that will be effective for the target user class and user environments.

TECHNOLOGY

The technology base to support hypertext and multimedia is developing and maturing rapidly (Cole, 1993). Many desktop computers, however, still lack adequate bandwidth (rate of data transmission) to display high-quality motion video with interleaved audio. This problem will be superseded by use of data buses with greater bandwidth and use of more advanced microprocessors and supporting microchips of various kinds.

The effective use of distributed multimedia products on either a local or wide area network basis often is limited by network bandwidth limitations. Accessing and transmitting complex hypertext or multimedia documents across relatively slow speed modems may result in inadequate response rates to user inputs and prevent the

desired tempo for video and audio presentation and user interactivity. In such cases the modem represents an example of a data transmission bottleneck. The use of more elaborate but higher speed digital network protocols and interconnections between multimedia servers and clients usually obviate this difficulty. Even so, initially adequate bandwidth designed for only a few network multimedia users can be swamped by multiple users simultaneously overloading the data transmission lines. For example, sufficiently high data transmission loads on a local area network line can lead to dropping, or failure to capture, a significant number of frames from a transmitted video segment. This then leads to a distracting flickering of the video. Up to a certain point, this adverse effect can be mitigated by reducing the size of the transmitted image size. However, this may result in an unacceptably small image size or excessively reduced spatial resolution.

One approach to minimizing the problems associated with limited bandwidth data transmission channels has been the extensive (and often automatic) use of a variety of digital data compression algorithms. Video and audio compression also has the desirable effect of decreasing data storage requirements. Compression algorithms that are currently in wide use include run length encoding (RLE), discrete cosine transform (DCT) methods, a Joint Photographic Expert Group standard (JPEG) for still and motion images, a Motion Picture Expert Group standard (MPEG) with or without interleaved audio, fractal-based methods, and wavelet-based algorithms. Compression of text files is often efficiently accomplished with a Huffman encoding scheme.

In addition to considering multimedia hardware issues, one must also be cognizant of the availability, capabilities, limitations, and costs of hypertext and multimedia authoring software. Such software includes authoring packages which may or may not be bundled with supporting image, video, and audio processing and editing modules.

Sophisticated hypertext and multimedia authoring tools are available. Books and sample diskettes or CD-ROMs (e.g., Badgett and Sandler, 1994) are available that provide comparative information for many different available authoring tools so that the most appropriate can be selected for a particular class of applications. A selection may be based on the features and design structure that need to be implemented to satisfy contractual requirements or to best tailor the product to the characteristics of the target user groups. Other considerations include the evaluation of the available computers and hardware components to support multimedia authoring, the type of operating system(s), software portability requirements, cost, and programming experience if any programming will be required.

Although authoring tools can facilitate multimedia development, they are not absolutely necessary. Recent versions of most operating systems as well as software programming environments are providing a larger array of features to facilitate the integration of multimedia into custom developed software. It is therefore becoming somewhat easier for the experienced programmer to create multimedia software using standard programming tools and appropriate libraries of functions or subroutines.

HYPERTEXT AUTHORING STANDARDS

The Standard Generalized Markup Language (SGML) is an example of a document tagging or indicator system that has become an international standard (SGML; ISO/IEC 8879-1986). This standard is a technical reference that established cross-platform guidelines and specific syntax for identifying the structural components and attributes of computer-based documents (Doyle, 1994). The SGML standard was established in 1986 and has been widely accepted.

SGML was incorporated as a requirement for the electronic document products supporting the U.S. Department of Defense (DoD) Computerized Acquisition and Logistics System (CALS). That is, electronic documents submitted for inclusion into CALS must be appropriately annotated with SGML. SGML has also been incorporated as a requirement for the DoD Interactive Electronic Technical Manuals (IETMs) project (Newcomb, et al., 1991). The common structural and content elements for a group of documents forms the basis for type classifying documents according to SGML templates. These templates are referred to as Document Type Definitions (DTD). A major advantage of the SGML is that it adds the capability for searching and locating structural elements across many interconnected documents. For example, one can search for a word or phrase in SGML tagged second-level subtopics across a multitude of interconnected documents developed on different platforms using different text processors. Without SGML, word or phrase searches are typically restricted to a single document and the structural elements to search can not be specified.

More advanced hypertext and multimedia authoring standards are currently in various stages of development and review. Such standards are essential in maximizing document interconnectivity and providing efficient navigation, control, and search capabilities for complex hypertext products that tie together documents and multimedia components from disparate sources (Newcomb, et al., 1991).

CONCLUSIONS

This technical report illustrated the feasibility of rendering USARIEM medical handbooks, deployment manuals (e.g., Jones, et al., 1993; Modrow, et al., 1993; Sawka, Roach, et al., 1994; Sawka, Modrow, et al., 1994), and other technical publications (e.g. Kraning II, 1991) into hypertext electronic documents. The ability to create a complex but easily traversed network of jumps, links, and pop-up windows with multimedia contents was demonstrated. Five USARIEM documents (Burr, 1991; Burr, 1993; Cymerman and Rock, 1994; Thomas, et al., 1993; USARIEM, 1993) were restructured as individual networks of hypertext nodes and links. It was also shown that such hypertext handbooks or publications can themselves be represented as hypertext nodes in a higher-level hypertext application. This higher-level hypertext application functioned as the entry point into the component hypertext documents. Additionally, an example was provided that demonstrated a method for integrating, into a dynamic lumped parameter cold digit simulation software program, a hypertext version of the USARIEM medical handbook for cold weather operations.

Figure 12 is a diagram that suggests a structure for developing, controlling, and maintaining a database of electronic hypertext and multimedia versions of the many technical reports, technical notes, and other publications authored by the USARIEM staff. Hypertext and multimedia provide additional formats for USARIEM informational and scientific products. These electronic multimedia or hypertext publications could then be distributed or accessed worldwide by U.S. and allied military or scientific organizations via digital communications networks such as the Internet. Access clearance would first be obtained from the USARIEM commander through an electronic document access, version control, and quality assurance committee. These hypertext and multimedia documents would be for more than just reading. They could also be structured for training, interactive browsing, and implement complex links between many widely scattered documents to tie together specific areas of knowledge. These documents would be periodically revised or replaced to keep the knowledge base current. The changes would be immediately available to on-line users. Document handling and shipping costs would be avoided. This is another major advantage of the electronic interactive document management scheme. Document version control and document distribution are simplified. Users could directly query the USARIEM hypertext or multimedia database to obtain the most current editions of USARIEM publications in environmental, deployment, and preventive medicine for performance enhancement, optimal nutrition, and injury prevention.

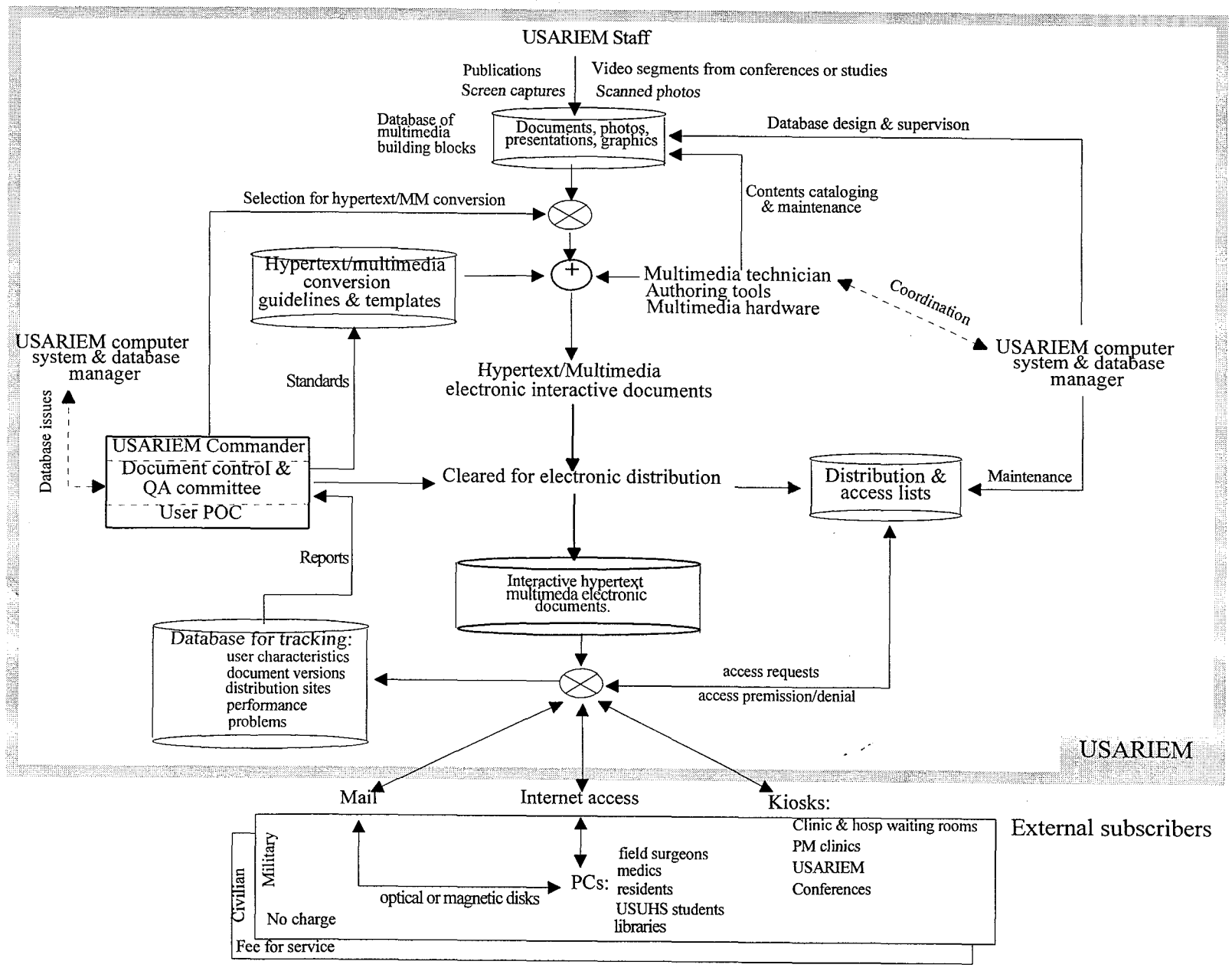


Figure 12: Hypertext document management diagram.

REFERENCES

Adam, J.A. Special report: Interactive multimedia. IEEE Spectrum, 30(3): 23-31, 1993.

Boyce, W.E. and DiPrima, R.C. Elementary differential equations and boundary value problems, 4th ed. John Wiley & Sons, Inc., NYC, NY, 1986.

Badgett, T. and Sandler, C. Creating Multimedia on Your PC (Includes the IBM Ultimedia Tools Series CD with 22 commercial sample programs for creating multimedia.) John Wiley and Sons, Inc., NYC, NY, 1994.

Burr, R.E. Heat illness: a handbook for medical officers. Natick, MA: U.S. Army Research Institute of Environmental Medicine, Technical Note TN91-3, 1991.

Burr, R.E. Medical aspects of cold weather operations: a handbook for medical officers. Natick, MA: U.S. Army Research Institute of Environmental Medicine, Technical Note TN93-4, 1993.

Carlson, H.L. Learning style and program design in interactive multimedia. Educational Technology, Research and Development, 39 (3): 41-48, 1991.

Cole, B. Special report: multimedia, the technology famework. IEEE Spectrum, 30(3): 32-39, 1993.

Cymerman, A. and Rock, P.B. Medical problems in high mountain environments: a handbook for medical officers. Natick, MA: U.S. Army Research Institute of Environmental Medicine, Technical Note TN94-2, 1994.

Doyle, C.D. (project editor). Computer Dictionary, 2nd ed. Microsoft Press, Redmond, WA, 1994.

Endrusick, T.L., Santee, W.R., DiRaimo, D.A., Blanchard, L.A., and Gonzalez, R.R. Physiologic Responses While Wearing Protective Footwear in a Cold-Wet Environment. Performance of Protective Clothing: Fourth Volume, ASTM STP 1133, James P. McBriarty and Norman W. Henry, Eds., American Society for Testing and Materials, Philadelphia, PA, 1992.

Fisher, S. Multimedia Authoring, Building and Developing Documents. AP Professional (a division of Harcourt Brace & Co.), Cambridge, MA, 1994.

Herberger, R.B. The Windows Help Magician Hypertext Authoring System for Windows User's Guide. Software Interphase Inc., Foster, RI, 1993.

Jonassen, D.H. Hypertext as instructional design. Educational Technology, Research and Development, 39 (1): 83-92, 1991.

Jones, B.H., Rock, P.B., Sawka, M.N., Modrow, H.E., Lindsay, G.C., Petrucelli, B., Mays, M.Z., O'Mara, M.A., Young, A.J., Strochein, L.A. and Krueger, G.P. (editors). Sustaining soldier health and performance in the former Republic of Yugoslavia: guidance for small unit leaders. Natick, MA: U.S. Army Research Institute of Environmental Medicine, Technical Note TN93-6, 1993.

Kraning II, K.K. A computer simulation for predicting the time course of thermal and cardiovascular responses to various combinations of heat stress, clothing and exercise. Natick, MA: U.S. Army Research Institute of Environmental Medicine, Technical Report T13-91, 1991.

Microsoft Corporation. Help Compiler Guide. In: Microsoft Visual Basic Programming System for Windows Version 2.0, Professional Features. Microsoft Corporation, Redmond, WA, 1992.

Modrow, H.E., Burr, R.E., Jones, B.H., Mays, M.Z., Rock, P.B., Sawka, M.N., Petrucelli, B., Matthew, W.T., Murphy, T.C., King, N., Hursh, S.R., Lindsay, G.C. and Krueger, G.P. (Eds.). Sustaining soldier health and performance in Somalia: guidance for small unit leaders. Natick, MA: U.S. Army Research Institute of Environmental Medicine, Technical Note TN93-1, 1993.

Newcomb, S.R., Kipp, N.A., and Newcomb, V.T. The HyTime hypermedia/time-based document structuring language. Communications of the ACM, 34(11): 67-83, 1991.

Park, I. and Hannafin, M.J. Empirically-based guidelines for the design of interactive multimedia. Educational Technology, Research and Development, 43(3): 63-85, 1993.

Parker, S.P. (editor in chief). McGraw-Hill Dictionary of Scientific and Technical Terms, 5th ed. McGraw-Hill, Inc., NYC, NY, 1994.

Reisman, S.A comparative study of multimedia personal computing and traditional instruction in a business school curriculum. Information Resources Management Journal, 6(4): 15-21, 1993.

Richey, R.C. The Theoretical and Conceptual Bases of Instructional Design. Nichols Publishing Co., NYC, NY, 1986.

Santee, W.R., Blanchard, L.A., Chang, S.KW., and Gonzalez, R.R. Biophysical model for handwear insulation testing. Natick, MA: U.S. Army Research Institute of Environmental Medicine, Technical Report T7-93, 1993.

Sawka, M.N., Roach, J.M., Young, A.J., Modrow, H.E., Reardon, M.J., Jones, B.H., Petruchelli, B., and Rock, P.B. Sustaining soldier health and performance in operation Support Hope: guidance for small unit leaders. Natick, MA: U.S. Army Research Institute of Environmental Medicine, Technical Note TN94-3, 1994.

Sawka, M.N., Modrow, H.E., Kolka, M.A., Roach, J.M., Stephenson, L.A., Young, A.J., Lipton, R.E., Reardon, M.J., Strochein, L.A., Tartarini, K.A., and Lindsay, G.C. Sustaining soldier health and performance in Southwest Asia: guidance for small unit leaders. Natick, MA: U.S. Army Research Institute of Environmental Medicine, Technical Note TN95-1, 1994.

Shneiderman, B. Hypertext. In: Encyclopedia of Computer Science, 3rd ed. Ralston, A. and Reilly, E. D. (Eds.) Van Nostrand Reinhold, NYC, NY, 1993.

Skytte, K. Engineering a small system. IEEE Spectrum. 31(3): 63-65, 1994.

Shitzer, A., Stroschein, L.A., Santee, W.R., Gonzalez, R.R. and Pandolf, K.B. Quantification of lower bounds for endurance times in thermally insulated fingers and toes exposed to cold stress. Natick, MA: U.S. Army Research Institute of Environmental Medicine, Technical Report T18-90, 1990.

SOLDIERS on the internet. (Eds.) Soldiers, 49(10): 6-7, 1994.

Texas Micro Public Relations Office. New field PC coming. Soldiers, 49(10): 8, 1994.

Thomas, C.D., Baker-Fulco, C.J., Jones, T.E., King, N., Jezior, D.A., Fairbrother, B.N., and Askew, E.W. Nutritional guidance for military field operations in temperate and extreme environments. Natick, MA: U.S. Army Research Institute of Environmental Medicine, Technical Note TN93-3, 1993.

USARIEM Type Protocol. Human research studies in the areas of thermal, hypoxic and operational stress, exercise, nutrition, and military performance. Natick, MA: U.S. Army Research Institute of Environmental Medicine, 1993.

Naval Special Warfare Command. SEAL Computer Assisted Medical Reference System User's Guide. Coronodo, CA, 1 October 1993.

Vaughan, T. Multimedia: Making it Work. Osborne McGraw-Hill, Berkley, CA, 1993.

APPENDIX A

Graphical User Interface Objects	Local (Form associated) Subroutines	Global Module Subroutines
Input Form	Form_Load	DefaultDigit
	LoadGloveList	
	Form_Unload	
Menu Items: Parameters	Parameters_Click	
Menu Items: Initialize	Init_Click	DefaultDigit
Menu Items: Calculate	CheckValueRanges	
	Calc_Click	MainCalc
		calc_eigenvals
		calc_Bn
		calc_lambda_sq
		calc_Cn
		calc_Gn
		calc_Hn
		Calc_T
		show_graph
Output		
Table	Table_Click	
Graph	Graph_Click	
Med Info	MedInfo_Click	WinHelp
About	About_Click	
Exit	Exit_Click	
Update Button	Update_Button_Click	inputcalc
	CheckValueRanges	
Parameter Input Boxes (18)	Parameter_Change	
	Parameter_LostFocus	
Parameter Descriptive Labels (18)	None	
Glove List Combo Box	GloveList_Click	
Calculate Frame	None	
Calculate Gauge	None	
About Form		
Descriptive Text Box	None	
Close Button	CloseButton_Click	
Graph Form	Form_Load, Form_Unload	
Graph	None	
Close Button	CloseButton_Click	
OutputGrid Form	Form_Unload	
Grid	None	
Close Button	CloseButton_Click	
Descriptive Labels (2)	None	

Graphical User Interface Objects & Associated Functions

APPENDIX B

The following is the Visual Basic 3.0 program listing for the cold digit model. The mathematics of the algorithm were described in USARIEM technical report T18-90 (Shitzer, et al., 1990). There are five principal sections in this implementation: the main module, input form, graph form, grid form, and the about form. The main module contains the data definitions, initialization, and numeric calculation subroutines. Comments within the program listing are preceded by a single quotation mark. Subroutines are identified by the prefix "Sub." This software program is event oriented; therefore, subroutines are not necessarily activated in the sequence in which they are listed below. Subroutines that define the response to user-activated events are identifiable by the format: " Sub id_event (...)." As an example, Sub CloseButton_Click() is the subroutine that defines the response to the clicking on a close button.

MAIN MODULE

```
Global Const HELP_CONTENTS = &H3
Global Const HELP_QUIT = &H2            'Terminate help

Declare Function WinHelp Lib "User" (ByVal hWnd As Integer, ByVal lpHelpFile As String, ByVal
wCommand As Integer, dwData As Any) As Integer

Type DigitCIVDdata_type
  on As Integer
  threshold_C As Double
  peak As Double
  period_mins As Double
  pulsedur_mins As Double
End Type
```

N.B.: Cold Induced Vasodilitation (CIVD) is not implemented in this version.

```
Type Digitdata_type
  h  As Double          'heat transfer coef @ cyl circ W/(m^2*C)
  h1 As Double          '        "    "    " " " tip   "
  k  As Double          'thermal conductivity W/(m*C)
  l  As Double          'cylinder (digit) length in meters
  n  As Integer         'number of terms in summation
  p  As Double          'cylinder (digit) circumference in meters
  q_i As Double         'initial digit tissue metabolism  W/m^3
  q_f As Double         'final or steady state " "   "   "
  Tdb As Double         'ambient dry bulb temperature
```

```
  Tbase_i  As Double    'initial base temp
  Tbase_f As Double     'final base temp
  Ttip_i  As Double     '  "   tip  "
  rho As Double         'tissue density
  d  As Double          'digit diameter
  c  As Double          'tissue specific heat J/(kg*C)
  delta As Double       'time const for digit base temp in hrs
  epsilon As Double     ' " "   "    "  metab heat generation in hrs
  alpha As Double       'thermal diffusivity m^2/hr
  tau As Double         'result of an intermediate function
  LL As Double          '4h/kd  m^2
  Area As Double        'cross-sectional area
  Bi As Double          'Biot modulus
  hrs_i  As Double      'initial start time in mins
  mins_incr As Double   'mins time increment
  hrs_f  As Double      'stop time in mins
  z_incr As Double      'distance from base increment
  T_thresh As Double    'digit threshold temp
  CIVD As DigitCIVDdata_type
End Type

Global digit As Digitdata_type

Global Const pi = 3.14159265

Global h_value(5) As Double

' Colors
Global Const BLACK = &H0&
Global Const RED = &HFF&
Global Const GREEN = &HFF00&
Global Const YELLOW = &HFFFF&
Global Const BLUE = &HFF0000
Global Const MAGENTA = &HFF00FF
Global Const CYAN = &HFFFF00
Global Const WHITE = &HFFFFFF

Sub DefaultDigit ()
  digit.h = 7.12              'heat transfer coef @ cyl circ W/(m^2*C)
  digit.h1 = 7.12             '  "    "     " " " tip   "
  digit.k = .418              'thermal conductivity W/(m*C)
```

```
digit.l = .08                    'cylinder (digit) length in meters
digit.n = 20                     'number of terms in summation
digit.p = .05                    'cylinder (digit) circumference in meters
digit.d = digit.p / pi
digit.q_i = 15000#               'initial digit tissue metabolism  W/m^3
digit.q_f = 5000#                'final or steady state " "   "   "
digit.Tdb = -5#                  'ambient dry bulb temperature
digit.Tbase_i = 30#              'initial base temp
digit.Tbase_f = 20#              'final base temp
digit.Ttip_i = 20#               '  "   tip  "
'Digit.c tissue specific heat J/(kg*C)
digit.delta = 1.3                'time const for digit base temp in hrs
digit.epsilon = 1.3              ' " "   "    "  metab heat generation in hrs
digit.alpha = .0004546                   'thermal diffusivity m^2/hr
digit.LL = 4 * digit.h / (digit.k * digit.d)  '4h/kd  m^2
digit.Area = digit.p ^ 2 / (4 * pi)      'cross-sectional area
digit.Bi = digit.h1 * digit.l / digit.k
digit.hrs_i = 0#                 'initial sim time
digit.hrs_f = 3          'final or end sim time
digit.mins_incr = 15#            'mins time increment
digit.z_incr = 8#                'mm increment from base
digit.T_thresh = 5#              'C finger threshold temperature
digit.CIVD.threshold_C = 10
digit.CIVD.peak = 5
digit.CIVD.period_mins = 15
digit.CIVD.pulsedur_mins = 5

Form1.h1_box.Text = digit.h1
Form1.k_box.Text = digit.k
Form1.l_box.Text = digit.l * 100 'meters --> cm
Form1.n_box.Text = digit.n
Form1.p_box.Text = digit.p * 100  'meter -->cm
Form1.q_i_box.Text = digit.q_i
Form1.q_f_box.Text = digit.q_f
Form1.Tdb_box.Text = digit.Tdb
Form1.Tbase_i_box.Text = digit.Tbase_i
Form1.Tbase_f_box.Text = digit.Tbase_f
Form1.Ttip_i_box.Text = digit.Ttip_i
Form1.Delta_box.Text = digit.delta
Form1.Epsilon_box.Text = digit.epsilon
Form1.Alpha_box.Text = digit.alpha
```

This block of lines involves implicit numeric-to-text conversion.

```
  Form1.hrs_i_box.Text = digit.hrs_i
  Form1.hrs_f_box.Text = digit.hrs_f
  Form1.mins_incr_box.Text = digit.mins_incr
  Form1.z_incr_box.Text = digit.z_incr
  Form1.T_thresh_box.Text = digit.T_thresh
  Form1.h_box.Text = digit.h
End Sub

Sub inputcalcs ()
  digit.d = digit.p / pi
  digit.LL = 4 * digit.h / (digit.k * digit.d)  '4h/kd  m^2
  digit.Area = digit.p ^ 2 / (4 * pi)'cross sectional area
  digit.Bi = digit.h1 * digit.l / digit.k
End Sub

Sub maincalc ()
  calc_eigenvals
  calc_Bn
  calc_lambda_sq
  calc_Cn
  calc_Gn
  calc_Hn
  calc_T
  show_graph
End Sub

Sub calc_eigenvals ()
  Fnum = FreeFile
  tol = .00001
  nn = digit.n
  Path = CurDir$
  Open "eigvals.dat" For Output As Fnum
  For i = 1 To nn
    x1_0 = (i * pi) - (pi / 2)
    x2_0 = i * pi
    x1 = x1_0
    x2 = x2_0
    x = x2 - .1
    diff = 1000000#
    Do While Abs(diff) > tol
      If diff > 0# Then x2 = x
```

These are intermediate calculations that must precede calculation of the main parameters in the subroutine below.

This subroutine calls the subroutines that calculate the parameters Bn, Cn, and Hn, as well as the eigenvalues (per Shitzer, et al., 1990). These values are saved in files to permit debugging and cross checking. The temperature profiles are calculated in the subroutine calc_T and displayed in graphical form by the show_graph subroutine.

A bisection method is used here to find the eigenvalues to an accuracy of less than or equal to "tol".

```
    If diff < 0# Then x1 = x
    x = (x2 + x1) / 2#
    diff = (-digit.Bi / x) - cot(x)
    If Abs(diff) < tol Then Write #Fnum, x
   ' If Abs(diff) < tol Then Debug.Print "eigenval# " & i & " =" & x
   Loop
  Next i
  Close #Fnum
End Sub

Sub calc_Bn ()
 Fnum_eig = FreeFile
 Open "eigvals.dat" For Input As Fnum_eig
 Fnum_Bn = FreeFile
 Open "bn.dat" For Output As Fnum_Bn
 For i = 1 To digit.n
  Input #Fnum_eig, eig
  numerator = 2 * (eig ^ 2 + digit.Bi ^ 2)
  denom = digit.l ^ 2 * (eig ^ 2 + digit.Bi ^ 2 + digit.Bi)
  Bn = numerator / denom
  Write #Fnum_Bn, Bn
 Next i
 Close Fnum_eig
 Close Fnum_Bn
End Sub

Sub calc_Cn ()
 Fnum_Bn = FreeFile
 Open "Bn.dat" For Input As Fnum_Bn
 Fnum_eig = FreeFile
 Open "eigvals.dat" For Input As Fnum_eig
 Fnum_Cn = FreeFile
 Open "Cn.dat" For Output As Fnum_Cn

 term2 = digit.LL * (digit.Tbase_f - digit.Tdb)
 term3 = digit.q_f / digit.k
 For i = 1 To digit.n
  Input #Fnum_eig, eig
  Input #Fnum_Bn, Bn
  term1 = Bn * digit.l / eig
  term4 = Cos(eig) - 1#
```

```
    Cn = term1 * (term2 + term3 * term4)
    Write #Fnum_Cn, Cn
   Next i
   Close Fnum_Bn
   Close Fnum_eig
   Close Fnum_Cn
End Sub


Sub calc_Gn ()
   Fnum_Bn = FreeFile
   Open "Bn.dat" For Input As Fnum_Bn
   Fnum_eig = FreeFile
   Open "eigvals.dat" For Input As Fnum_eig
   Fnum_lambda = FreeFile
   Open "lambda.dat" For Input As Fnum_lambda
   Fnum_Gn = FreeFile
   Open "Gn.dat" For Output As Fnum_Gn

   term2 = (digit.Tbase_i - digit.Tbase_f) * (digit.alpha * digit.delta * digit.LL - 1#)
   For i = 1 To digit.n
     Input #Fnum_Bn, Bn
     Input #Fnum_eig, eig
     Input #Fnum_lambda, lambda_sq
     term1 = Bn * digit.I / eig
     term3 = digit.alpha * lambda_sq * digit.delta - 1#
     Gn = term1 * (lambda_sq * term2) / term3
     Write #Fnum_Gn, Gn
   Next i
   Close Fnum_Gn
   Close Fnum_lambda
   Close Fnum_eig
   Close Fnum_Bn
End Sub

Sub calc_Hn ()
   Fnum_Bn = FreeFile
   Open "Bn.dat" For Input As Fnum_Bn
   Fnum_eig = FreeFile
   Open "eigvals.dat" For Input As Fnum_eig
   Fnum_lambda = FreeFile
```

```
  Open "lambda.dat" For Input As Fnum_lambda
  Fnum_Hn = FreeFile
  Open "Hn.dat" For Output As Fnum_Hn

  term4 = digit.q_i - digit.q_f
  For i = 1 To digit.n
    Input #Fnum_Bn, Bn
    Input #Fnum_eig, eig
    Input #Fnum_lambda, lambda_sq
    term1 = digit.alpha * digit.l * Bn / (digit.k * eig)
    term2 = Cos(eig) - 1#
    term3 = lambda_sq * digit.epsilon * (digit.q_i - digit.q_f) / (digit.alpha * digit.epsilon * lambda_sq - 1#)
    Hn = term1 * term2 * term3
    Write #Fnum_Hn, Hn
  Next i

  Close Fnum_Hn
  Close Fnum_lambda
  Close Fnum_eig
  Close Fnum_Bn
End Sub

Sub calc_lambda_sq ()
 Fnum_eig = FreeFile
 Open "eigvals.dat" For Input As Fnum_eig
 Fnum_lambda = FreeFile
 Open "lambda.dat" For Output As Fnum_lambda
 For i = 1 To digit.n
  Input #Fnum_eig, eig
  lambda_sq = (eig / digit.l) ^ 2 + digit.LL
  Write #Fnum_lambda, lambda_sq
 Next i
 Close Fnum_eig
 Close Fnum_lambda
End Sub

Sub calc_T ()
  Fnum_T = FreeFile
  Open "Tdig.dat" For Output As Fnum_T

  tau = digit.tau
```

```
epsilon = digit.epsilon
alpha = digit.alpha
delta = digit.delta
Tbase_i = digit.Tbase_i
T1 = Tbase_i
Tbase_f = digit.Tbase_f
Ttip_i = digit.Ttip_i
Tdb = digit.Tdb
Bi = digit.Bi

numloops_t = Int((digit.hrs_f - digit.hrs_i) / (digit.mins_incr / 60#)) + 1
numloops_z = Int(digit.l / (digit.z_incr * 10 ^ -3))
numloops = digit.n * numloops_z * numloops_t      'digit.n is # of eigenvals per summation

'Label top row on output grid with z incr from digit base
OutputGrid_Form1.Grid1.Row = 0
OutputGrid_Form1.Grid1.Col = 1
For i = 0 To numloops_z
  OutputGrid_Form1.Grid1.Text = .1 * (digit.z_incr * (i))  'mult by 0.1 for mm -> cm
  OutputGrid_Form1.Grid1.Col = OutputGrid_Form1.Grid1.Col + 1
  OutputGrid_Form1.Grid1.FixedAlignment(i) = 2
Next i

'Label left col on output grid with time in mins
OutputGrid_Form1.Grid1.Row = 1
OutputGrid_Form1.Grid1.Col = 0
For i = 0 To numloops_t - 1
  OutputGrid_Form1.Grid1.ColWidth(i) = 700
  If i = 0 Then
    OutputGrid_Form1.Grid1.FixedAlignment(i) = 2   'center align the data
  Else
    OutputGrid_Form1.Grid1.ColAlignment(i) = 2
  End If
  OutputGrid_Form1.Grid1.Text = (digit.mins_incr * (i))
  OutputGrid_Form1.Grid1.Row = OutputGrid_Form1.Grid1.Row + 1
Next i


loops = 0

OutputGrid_Form1.Grid1.Row = 1  ' 1st position in output grid
```

```
OutputGrid_Form1.Grid1.Col = 1

Form1.Calc_gage.Visible = True

For z = 0# To (digit.l + 10 ^ -6) Step digit.z_incr * 10 ^ -3
 For t = digit.hrs_i To digit.hrs_f Step digit.mins_incr / 60#
  Fnum_Bn = FreeFile
  Open "Bn.dat" For Input As Fnum_Bn
  Fnum_Cn = FreeFile
  Open "Cn.dat" For Input As Fnum_Cn
  Fnum_Gn = FreeFile
  Open "Gn.dat" For Input As Fnum_Gn
  Fnum_Hn = FreeFile
  Open "Hn.dat" For Input As Fnum_Hn
  Fnum_eig = FreeFile
  Open "eigvals.dat" For Input As Fnum_eig
  Fnum_lambda = FreeFile
  Open "lambda.dat" For Input As Fnum_lambda

  sum = 0#
  For n = 1 To digit.n Step 1
   Input #Fnum_Bn, Bn
   Input #Fnum_Cn, Cn
   Input #Fnum_Gn, Gn
   Input #Fnum_Hn, Hn
   Input #Fnum_eig, eig
   Input #Fnum_lambda, lambda_sq

   'first calc tau:
   'T1_t = Tbase_f + (Tbase_i - Tbase_f) * Exp(-t / delta)
   term1 = Bn * digit.l / eig
   term2 = ((Ttip_i - Tbase_i) / eig) * Sin(eig)
   term3 = Tbase_i - T1 - (Ttip_i - Tdb) * Cos(eig)
   tau = term1 * (term2 + term3)
   'now calc the sumation in the T(z,t) eqn:
   term1 = (lambda_sq * tau + Cn + Gn + Hn) * Exp(-alpha * lambda_sq * t)
   term2 = -Cn - Gn * Exp(-t / delta) - Hn * Exp(-t / epsilon)
   term3 = (Sin(eig * z / digit.l)) / lambda_sq
   sum = sum + (term1 + term2) * term3
  Next n
  Close Fnum_Bn
```

```
    Close Fnum_Cn
    Close Fnum_Gn
    Close Fnum_Hn
    Close Fnum_lambda
    Close Fnum_eig

    term4 = (Bi / (1# + Bi)) * (z / digit.l) - 1#
    term5 = Tbase_f + (Tbase_i - Tbase_f) * Exp(-t / delta) - Tdb

    T_z_t = Tdb + digit.l * sum - (term4 * term5)
    Write #Fnum_T, T_z_t
  ' Debug.Print T_z_t
    loops = loops + digit.n

    Form1.Calc_gage.Value = Int(100 * loops / numloops)

    If OutputGrid_Form1.Grid1.Row < OutputGrid_Form1.Grid1.Rows And
    OutputGrid_Form1.Grid1.Col < OutputGrid_Form1.Grid1.Cols Then
      OutputGrid_Form1.Grid1.Text = Format(T_z_t, "##.#")
    End If
    OutputGrid_Form1.Grid1.Row = OutputGrid_Form1.Grid1.Row + 1

  Next t
  OutputGrid_Form1.Grid1.Row = 1
   ' i.e. go back to 1st row for each new point along the finger
  OutputGrid_Form1.Grid1.Col = OutputGrid_Form1.Grid1.Col + 1
 Next z

 Close Fnum_T
 Form1.Calc_gage.Visible = False

End Sub


Function cot (x)
  cot = Cos(x) / Sin(x)
End Function
```

```
Sub show_graph ()
  Fnum_T = FreeFile
  Open "Tdig.dat" For Input As Fnum_T
  NumLocations = Int(digit.l / (digit.z_incr * 10 ^ -3))
  NumTimeIncrs = Int((digit.hrs_f - digit.hrs_i) / (digit.mins_incr / 60#)) + 1

  Graph_frm.T_graph.DataReset = 1  'clear any data

  Graph_frm.T_graph.NumPoints = NumTimeIncrs
  Graph_frm.T_graph.NumSets = NumLocations
  For pt = 1 To NumLocations
   For t = 1 To NumTimeIncrs
    Input #Fnum_T, T_dig
    Graph_frm.T_graph.GraphData = T_dig
   Next t
  Next pt
  Close Fnum_T
  Graph_frm.Show
  Graph_frm.T_graph.DrawMode = 2
End Sub
```

INPUT FORM

```
Sub Calc_Click ()
  Param_frame.Refresh
  Update_button.SetFocus
  CheckValueRanges badvals
  If badvals = True Then
     Calc_frame.Visible = False
     Exit Sub
  End If
  Graph_frm.DrawMode = 1 'clear the old graph
  Calc_frame.Visible = True
  Form1.Refresh
  inputcalcs  'calc diameter, area, Biot modulus, & LL
  maincalc
  Calc_frame.Visible = False
  OutputGrid_Form1.Visible = True
```

```
End Sub

Sub CheckValueRanges (badvals)
  badvals = False
    If digit.hrs_i < 0# Or digit.hrs_i > 100# Then
        hrs_i_box.SetFocus
        badvals = True
 ElseIf digit.hrs_f < digit.hrs_i Or digit.hrs_f > 100# Then
        hrs_f_box.SetFocus
        badvals = True
 ElseIf digit.mins_incr < 5# Or digit.mins_incr > 60 Then
        mins_incr_box.SetFocus
        badvals = True
 ElseIf digit.z_incr < 2# Or digit.z_incr > digit.l * 1000 / 4 Then
       'digit.l was input as cm & changed to meters in another sub
       'therefore the use of the 1000 multiplier
        z_incr_box.SetFocus
        badvals = True
 ElseIf digit.Tdb < -50 Or digit.Tdb > 70 Then
        Tdb_box.SetFocus
        badvals = True
 ElseIf digit.T_thresh < -10 Or digit.T_thresh > 30 Then
        T_thresh_box.SetFocus
        badvals = True
 ElseIf digit.Tbase_i < 30 Or digit.Tbase_i > 45 Then
        Tbase_i_box.SetFocus
        badvals = True
 ElseIf digit.Tbase_f < 0 Or digit.Tbase_f > 45 Then
        Tbase_f_box.SetFocus
        badvals = True
 ElseIf digit.Ttip_i < 0 Or digit.Ttip_i > 45 Then
        Ttip_i_box.SetFocus
        badvals = True
 ElseIf digit.n < 1 Or digit.n > 100 Then
        n_box.SetFocus
        badvals = True
 ElseIf digit.p < .01 Or digit.p > .5 Then
        p_box.SetFocus
        badvals = True
 ElseIf digit.l < .01 Or digit.l > .5 Then
        l_box.SetFocus
```

```
            badvals = True
    ElseIf digit.q_i < 0 Or digit.q_i > 100000 Then
            q_i_box.SetFocus
            badvals = True
    ElseIf digit.q_f < 0 Or digit.q_i > 100000 Then
            q_f_box.SetFocus
            badvals = True
    ElseIf digit.h < 0 Or digit.h > 50# Then
            h_box.SetFocus
            badvals = True
    ElseIf digit.h1 < 0 Or digit.h1 > 50# Then
            h1_box.SetFocus
            badvals = True
    ElseIf digit.k < 0 Or digit.k > 10# Then
            k_box.SetFocus
            badvals = True
    ElseIf digit.delta <= 0 Or digit.delta > 5# Then
           delta_box.SetFocus
           badvals = True
    ElseIf digit.epsilon <= 0 Or digit.epsilon > 5# Then
           epsilon_box.SetFocus
           badvals = True
    ElseIf digit.alpha <= 0 Or digit.alpha > 1# Then
           alpha_box.SetFocus
           badvals = True
    End If
End Sub

Sub circ_box_Change ()
    digit.p = circ_box / 100#  'change circ from cm to meters
End Sub


Sub Delta_box_LostFocus ()
  digit.delta = delta_box.Text
End Sub

Sub Epsilon_box_LostFocus ()
  digit.epsilon = epsilon_box.Text
End Sub
```

```
Sub Exit_Click ()
  End
End Sub

Sub Form_Load ()
   DefaultDigit
   LoadGloveList
   GloveList.ListIndex = 0
   Load Graph_frm
   Load OutputGrid_Form1
   Load About_form
End Sub

Sub Form_Unload (Cancel As Integer)
  End
End Sub

Sub GloveList_Click ()
   digit.h = h_value(GloveList.ListIndex)
   h_box.Text = digit.h
   digit.h1 = h_value(GloveList.ListIndex)
   h1_box.Text = digit.h1
End Sub

Sub Grafs_Click ()
  Graph_frm.Visible = True
  Graph_frm.SetFocus
End Sub

Sub h_box_LostFocus ()
  digit.h = h_box.Text
End Sub


Sub h1_box_Change ()
  digit.h1 = h1_box.Text
End Sub

Sub hrs_f_box_LostFocus ()
   digit.hrs_f = hrs_f_box.Text
End Sub
```

```
Sub hrs_i_box_LostFocus ()
  digit.hrs_i = hrs_i_box.Text
End Sub

Sub Init_Click ()
    DefaultDigit
    Param_frame.Refresh
End Sub

Sub k_box_LostFocus ()
    digit.k = k_box.Text
End Sub

Sub l_box_LostFocus ()
   digit.l = .01 * l_box.Text  'cm to meters
End Sub

Sub LoadGloveList ()
   'n.b.: h_value = clo * 7.9111
   GloveList.AddItem "          Bare Hand (0.3 clo)", 0
   h_value(0) = 25#
   GloveList.AddItem "    Light-duty glove (0.8 clo)", 1
   h_value(1) = 8.33
   GloveList.AddItem "  Vehicle crew glove (1.0 clo)", 2
   h_value(2) = 6.25
   GloveList.AddItem "Trigger-finger mitten (1.3 clo)", 3
   h_value(3) = 4.76
   GloveList.AddItem "      Arctic mitten (2.2 clo)", 4
   h_value(4) = 2.85
   GloveList.AddItem "   Other (customize h values)", 5
   h_value(5) = 7.12
End Sub


Sub MedInfo_Click ()
   dwData& = 1
   Return_val% = WinHelp(hWnd, "c:\digit\cold2.hlp" + Chr$(0), HELP_CONTENTS, ByVal dwData&)
End Sub
```

```
Sub mins_incr_box_LostFocus ()
  digit.mins_incr = mins_incr_box.Text
End Sub

Sub n_box_LostFocus ()
   digit.n = n_box.Text
End Sub

Sub p_box_LostFocus ()
  digit.p = .01 * p_box.Text
End Sub

Sub Parameters_Click ()
  Param_frame.Visible = True
  Update_button.Visible = True
End Sub

Sub q_f_box_LostFocus ()
    digit.q_f = q_f_box.Text
End Sub

Sub q_i_box_LostFocus ()
   digit.q_i = q_i_box.Text
End Sub

Sub T_thresh_box_LostFocus ()
  digit.T_thresh = T_thresh_box.Text
End Sub

Sub Table_Click ()
  OutputGrid_Form1.Visible = True
  OutputGrid_Form1.SetFocus
End Sub

Sub Tbase_f_Box_LostFocus ()
  digit.Tbase_f = Tbase_f_box.Text
End Sub
```

```
Sub Tbase_i_Box_LostFocus ()
   digit.Tbase_i = Tbase_i_box.Text
End Sub

Sub Tdb_box_LostFocus ()
  digit.Tdb = Tdb_box.Text
End Sub

Sub Ttip_i_Box_LostFocus ()
   digit.Ttip_i = Ttip_i_box.Text
End Sub

Sub Update_button_Click ()
   Update_button.SetFocus
   Param_frame.Refresh
   CheckValueRanges badvals  'sets focus on input field that is out of range
   inputcalcs
End Sub

Sub z_incr_box_LostFocus ()
  digit.z_incr = z_incr_box.Text
End Sub
```

GRAPH FORM

```
Sub CloseButton_Click ()
  Graph_frm.Hide
  Form1.SetFocus
End Sub

Sub Form_Load ()
   T_graph.DataReset = 1'reset the data
End Sub

Sub Form_Unload (Cancel As Integer)
   Hide
End Sub
```

OUTPUT GRID FORM

```
Sub CloseButton_Click ()
   OutputGrid_form1.Hide
   Form1.SetFocus
End Sub

Sub Form_Unload (Cancel As Integer)
  Hide
End Sub
```

ABOUT FORM

```
Sub CloseButton_Click ()
  About_Form.Hide
  Form1.SetFocus
End Sub
```

DISTRIBUTION LIST

2 Copies to:

Defense Technical Information Center
ATTN: DTIC-DDA
Alexandria VA 22304-6145

Office of the Assistant Secretary of Defense (Hlth Affairs)
ATTN: Medical Readiness
Army Pentagon
Washington DC 20310-0103

Commander
U.S. Army Medical Research and Materiel Command
ATTN: MCMR-OP
Fort Detrick
Frederick MD 21702-5012

Commander
U.S. Army Medical Research and Materiel Command
ATTN: MCMR-PLE
Fort Detrick
Frederick MD 21702-5012

Commander
U.S. Army Medical Research and Materiel Command
ATTN: MCMR-PLC
Fort Detrick
Frederick MD 21702-5012

Commandant
Army Medical Department Center and School
ATTN: HSHA-FM, Bldg. 2840
Fort Sam Houston TX 78236

1 Copy to:

Joint Chiefs of Staff
Medical Plans and Operations Division
Deputy Director for Medical Readiness
Army Pentagon
Washington DC 20310-2300

HQDA
Office of the Surgeon General
Preventive Medicine Consultant
ATTN: SGPS-PSP
5109 Leesburg Pike
Falls Church VA 22041-3258

HQDA
Office of the Surgeon General
ATTN: DASG-ZA
5109 Leesburg Pike
Falls Church VA 22041-3258

HQDA
Office of the Surgeon General
ATTN: DASG-MS
5109 Leesburgh Pike
Falls Chaurch VA 22041-3258

HQDA
Office of the Surgeon General
Assistant Surgeon General
ATTN: DASG-RDZ/Executive Assistant
Room 3E368, Army Pentagon
Washington DC 20310-2300

HQDA
Assistant Secretary of the Army
(Research, Development and Acquisition)
ATTN: SARD-TM
103 Army Pentagon
Washington DC 20310-2300

Uniformed Services University of the Health Sciences
ATTN: Dean, School of Medicine
4301 Jones Bridge Road
Bethesda MD 20814-4799

Uniformed Services University of Health Sciences
ATTN: Department of Military and Emergency Medicine
4301 Jones Bridge Road
Bethesda MD 20814-4799

Commander
U.S. Army Environmental Hygiene Agency
Aberdeen Proving Ground MD 21010-5422

Director, Biological Sciences Division
Office of Naval Research - Code 141
800 N. Quincy Street
Arlington VA 22217

Commanding Officer
Naval Medical Research and Development Command
NNMC/ Bldg. 1
Bethesda MD 20889-5044

Commanding Officer
U.S. Navy Clothing & Textile Research Facility
ATTN: NCTRF-01
Natick MA 01760-5000

Commanding Officer
Naval Environmental Health Center
2510 Walmer Avenue
Norfolk VA 23513-2617

Commanding Officer
Naval Medical Research Institute
Bethesda MD 20889

Commanding Officer
Naval Health Research Center
P.O. Box 85122
San Diego CA 92138-9174

Commander
USAF Armstrong Medical Research Laboratory
Wright-Patterson Air Force Base OH 45433

Strughold Aeromedical Library
Document Services Section
2511 Kennedy Circle
Brooks Air Force Base TX 78235-5122

Commander
USAF School of Aerospace Medicine
Brooks Air Force Base TX 78235-5000

Commander
U.S. Army Medical Research Institute of Chemical Defense
ATTN: MCMR-UVZ
Aberdeen Proving Ground MD 21010-5425

Commander
U.S. Army Medical Materiel Development Activity
ATTN: MCMR-UMZ
Fort Detrick
Frederick MD 21702-5009

Commander
U.S. Army Institute of Surgical Research
ATTN: MCMR-UMZ
Fort Sam Houston TX 21702-6200

Commander
U.S. Army Medical Research Institute of Infectious Diseases
ATTN: MCMR-UIZ
Fort Detrick
Frederick MD 21702-5011

Director
Walter Reed Army Institute of Research
ATTN: MCMR-UWZ-C (Director for Research Management)
Washington DC 20307-5100

Commander
U.S. Army Natick Research, Development & Engineering Center
ATTN: SATNC-Z
Natick MA 01760-5000

Commander
U.S. Army Natick Research, Development & Engineering Center
ATTN: SATNC-T
Natick MA 01760-5000

Commander
U.S. Army Natick Research, Development & Engineering Center
ATTN: SATNC-MIL
Natick MA 01760-5000

Director
U.S. Army Research Institute for the Behavioral Sciences
5001 Eisenhower Avenue
Alexandria VA 22333-5600

Commander
U.S. Army Training and Doctrine Command
Office of the Surgeon
ATTN: ATMD
Fort Monroe VA 23651-5000

Commandant
U.S. Army Medical Department Center & School
Stimson Library
ATTN: Chief Librarian
Bldg. 2840, Room 106
Fort Sam Houston TX 78234-6100

Commandant
U.S. Army Medical Department Center & School
ATTN: Director of Combat Development
Fort Sam Houston TX 78234-6100

Commander
U.S. Army Aeromedical Research Laboratory
ATTN: MCMR-UAX-SI
Fort Rucker AL 36362-5292

Director
U.S. Army Research Laboratory
Human Research & Engineering Directorate
Aberdeen Proving Ground MD 21005-5001

Technical Director
Defence and Civil Institute of Environmental Medicine
1133 Sheppard Avenue W.
P.O. Box 2000
Downsview, Ontario
CANADA M3M 3B9

Commander
U.S. Army Military History Institute
ATTN: Chief, Historical Reference Branch
Carlisle Barracks
Carlisle PA 17013-5008

Commander
U.S. Army Natick Research, Development and Engineering Center
ATTN: SATNC-TM
U.S. Marine Corps Representative
Natick MA 01760-5004