

# HYPERMEDIA-BASED SUPPORT FOR REQUIREMENTS ANALYSIS: PROMISE AND PROBLEMS

Peter Aiken  
Department of Information Systems  
Virginia Commonwealth University  
Richmond, VA 23284-4000

paiken@cabell.vcu.edu

James Bach  
Development Systems Quality Group  
Apple Computer  
Mountain View, CA 94043

bach04@applelink.apple.com

## ABSTRACT

This paper describes our efforts at providing hypermedia-based support for requirements analysis. We present: an overview of our concept of hypermedia-based support for the requirements analysis process; our experiences using this technology; and some remaining technological challenges highlighted by our experiences. Our analysis of requirements analyst's needs indicates several productivity related benefits will result from the successful application of hypermedia technologies especially in rapid prototyping contexts. These relate to hypermedia's ability to more faithfully capture and represent system requirements, providing increased requirements traceability and giving analysts direct, repeatable access to multimedia requirements information that would otherwise be unavailable.

## INTRODUCTION

The conceptual basis for hypermedia-based technology has been described elsewhere [1], [6], [7], and [9]. Our research has leveraged hypermedia-specific capabilities to increase the productivity of requirements analysis efforts. The basic goal was to increase the time and brain power able to be devoted to analysis by decreasing the effort spent on the mechanics of requirements analysis. The next section of this paper provides an overview of the four basic information processing functions common to all requirements analyses. Section three briefly describes our experiences testing the functionality of our software in a series of software development exercises. The remaining section of the paper examines the technological challenges identified from an analysis of our experiences with this new technology.

## REQUIREMENTS ANALYSIS INFORMATION PROCESSING FUNCTIONS

The focus of much requirements analysis literature is on methodological approaches to requirements analysis. Our research has sought to provide computer-based support for a set of common information processing functions, part of any requirements analysis regardless of the specific methodology employed. This section describes how, from an information processing perspective, requirements analysis can be viewed in terms of four basic functions. As a framework we adopted Andriole's rapid prototyping methodology consisting of the following tasks: requirements elicitation, task/user/environmental profiling, requirements information analysis, task/user/environmental modeling, solution synthesis/design, prototype development, feedback, and evaluation [2]. Figure 1 shows a mapping between the requirements analysis tasks and the four basic functions. The functions are: 1) **capturing** information about the task to be implemented, the system users, and any relevant organizational/situational characteristics, 2) **organizing** the information into an easily accessible form for use in subsequent phases, 3) **synthesizing** a problem solution from the information gathered, and 4) **presenting** the solution to the user to obtain corrective feedback.

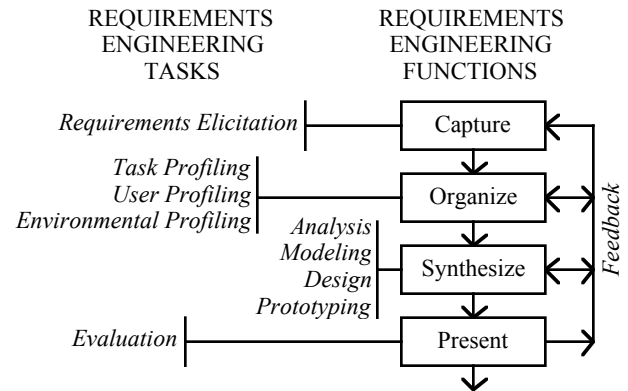


Figure 1. Requirements Analysis Tasks Mapped Onto Requirements Analysis Functions.

Information flow during conventional software development can be viewed as two damaging transformations. The first, a keyboard-based transformation, strips the information of bandwidth as requirements are reduced to text-based problem representations. The second, a printer-based transformation, occurs when the text-based representation of the real-world data is printed on paper for distribution. This distribution media effectively prohibits automated searching and direct branching information retrieval functions. Thus, during requirements analysis, real world multimedia inputs are reduced to static text and graphics-based requirements as they become part of a traditional requirements document. As this occurs, the user's ability to access the requirements information becomes restricted to using predefined tables of contents and indices, and to using sequential access or manual browsing. (Although over time these can be augmented by direct access based on the experiential familiarity of the user with the document.) Hypermedia-based support for requirements analysis offers the opportunity to use the information retrieval functions searching and direct access, and to incorporate higher bandwidth requirements information into the analysis. Figure 2 summarizes the four basic requirements analysis functions, describes potential benefits of the implemented function, and presents our software implementation of each function. (For more information see [1].)

## EXPERIENCES

### Workstation Development

We used an early version of the software to illustrate the functionality of the proposed workstation during its development. prototyping screen layouts, program flows, and program control features. We used the resulting prototype to explain the system concept and justify budget requests. An improved version of the software was used to further define the various functions and sub-functions by interacting with two requirements analysis experts.

Function	Form Of Support	Goals Of Support	Implementation Of Support
<b>CAPTURE</b>	Augmenting the processes used to obtain requirements information	Increase the amount and quality of information available to the requirements analyst for direct analysis	Increasing the types, amount and quantity of media able to be captured and manipulated during requirements analysis
<b>ORGANIZE</b>	Augmenting the process of arranging captured requirements information into easily accessible task, user, and environmental profiles	Assist the requirements analyst to effectively and efficiently group requirements information into personalized collections	Providing means of transforming inaccessible information into accessible information and simplifying object manipulations
<b>SYNTHESIZE</b>	Augmenting the ability of the requirements analyst to use multimedia requirements information to analyst problem solutions	Assist the requirements analyst in gaining increased situation comprehension and augmented solution development	Facilitating the process of creating associative indices capable of increasing the requirements analyst's visual and cognitive momentum
<b>PRESENT</b>	Augmenting the process used to ensure that the requirements analyst and the package user/sponsor both have the same understanding of the proposed system	Provide for increased communication between the analyst and the package user/sponsor via prototyping efforts	Providing support for interface design, simulation of program flow/control and support for prototype presentation sequencing in an environment with integrated access to requirements information

Figure 2. Requirements Analysis Functions.

Over a period of three months the experts reviewed and commented on several iterations, each representing an enhanced implementation of the software. The review process consisted of having the experts interact with the software, sometimes bringing requirements information to the exercise to decide how well the system met their needs. Again the process was judged successful because of the short, six month, development that took the system from initial concept to an alpha release of the software.

### Software Testing

Another opportunity occurred on a project concerned with the development of software automating the process of creating software test plans from system requirements. (See: [10] for details.) We used the software to establish program flow from a series of interactive screens. This exercise highlighted the need for several features including: the ability of the user to click on various screen objects and receive the appropriate screen, and the ability to modify aspects of the interface during the prototype evaluation session in an on-the-fly mode. These additional features were implemented and judged to have successfully contributed to the development of the prototype.

### ScholaRite

The first author participated in a requirements exercise during the second author's development of a personal productivity tool called ScholaRite. At points during a recorded interview, the sponsor created a series of off-the-cuff sketches of desired interface features. Later in the analysis, we added the sketches and portions of the audio tape to the requirements information. The audio sound-bites supplemented the text and graphics-based requirements information. The exercise indicated the need to create composite information chunks. We used these composite chunks as the basis for a prototype of the ScholaRite system. The ability to create composite information chunks was particularly helpful in rapidly gaining an understanding of the user and software requirements. Subsequent use of these chunks in presentations to groups not familiar with ScholaRite have since justified the effort spent creating them. There is no doubt hypermedia support permitted us to become closer to and more involved with the requirements information and become more knowledgeable about the contents of the information.

### Transportation System Problem Analysis

Most recently we have applied the support to exercises relating to identification of user requirements for software supporting the efforts of transportation planners concerned with regional mobility issues. This effort, while currently incomplete, has offered the first opportunity to incorporate video-based information chunks into the system. Initial results seem to confirm that sponsor interest in the prototype evaluation sessions can be increased with motion based information. Increased interest in the prototype leads to increased communication between the prototypers and sponsors and seems to indicate a shorter development time due to more productive iterations.

### PROBLEMS REMAINING

Our experience with the software has led to the identification of several problems requiring solutions before the hypermedia support for requirements analysis can be said to be truly viable. The problems fall into the categories of: information management, approach to requirements analysis, technological constraints, data compatibility, economic considerations, and social factors.

### Information Management

From an information management point of view: 1) non-textual information chunks are inherently difficult to index, 2) methods of navigating associative indices are not well understood, and 3) non-textual information chunks outnumber textual chunks by a large amount. These factors combine to make the process of managing multimedia requirements information difficult. Figure 3 shows how requirements information is used in the life cycle.

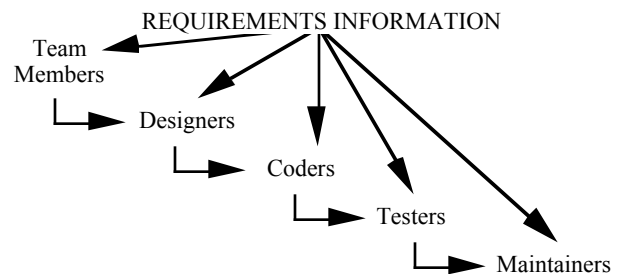


Figure 3. Uses of Requirements Information in the Life Cycle.

Text-based information chunks are inherently easier to index than non-textual chunks. Consider the current the process for indexing non-textual information chunks. The analyst notes the index numbers at the occurrence of significant events during the recording process. Additional occurrences are noted during subsequent reviews of the information. The analog information is then manually divided into smaller chunks. A programmatic means of aiding the requirements analyst with this process is under development but not yet implemented.

Methods of navigating associative indices are not well understood. In addition, the links created can be highly personal in nature. That is, they will serve to reinforce associations between information chunks meaningful to a specific requirements analyst. These same links can have little meaning to subsequent personnel accessing the information. Figure 4 illustrates how information chunks from different domains can be linked together to form a web of information.

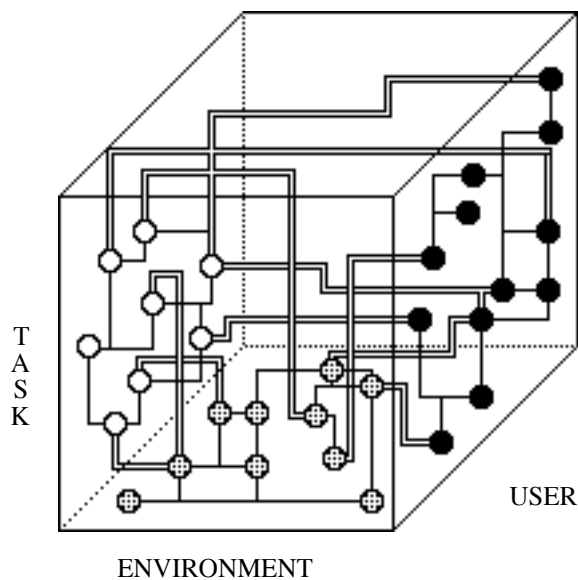


Figure 4. Conceptual Requirements Model.

Especially in during the capture phase of analysis, non-textual information chunks outnumber comparable textual information chunks by several orders of magnitude. The world is full of audiovisual project information. It is easier to create a large volume of information with a video-based recording system than with a word processor. Even with a high degree of filtering, the analyst can be required to process and index hours of recorded information. We have created hundreds of information chunks from the recordings and currently each chunk must be manually spliced and indexed. Even with hypermedia support, the workload required to process these chunks far exceeds the work associated with processing a text-based transcript. We can automatically index the transcript and incorporated the index in the final requirements document. Working with hypermedia to get these results requires more work when compared to current methods. Research has yet to establish guidelines for the characteristics of useful multimedia information chunks. Once identified, these can reduce the required processing volume.

## Approach To Requirements Analysis

From a methodological point of view it is not clear if this form of support can integrate/supplement traditional approaches to requirements analysis or whether a new requirements analysis methodology will be required. It is possible with the focus on organizing and managing requirements information, a hypermedia approach can augment existing techniques. Our we have used several variations of Andriole's requirements methodology but have not identified the answers to these questions.

It is also not clear how information generated and used by this form of support can be integrated with information used by contemporary analysis methods. We need to identify the focus of the information gathering process from huge potential amounts of information ranging from casual conversations to formal meetings to blackboard sketches. More research needs to be done to create or identify situations containing enough high quality information to be worth recording.

## Technological Constraints

From a technological point of view, devices for recording non-textual information chunks tend not to be integrated. They weren't designed to work with each other. Standard interfaces for remote control, similar to the MIDI protocol for electronic musical instruments, and further miniaturization help. Our first attempt at the workstation has been mostly successful at providing the required degree of integration but the workstation is not portable.

Digitization of information chunks requires memory. Ten seconds of digitized voice can consume a quarter megabyte of memory. Advances in information compression techniques and higher capacity storage, combined with the continuing drop in memory prices and the arrival of erasable optical storage will somewhat help this situation. Currently, manipulating these information chunks is bulky and time consuming.

Automated indexing of non-textual information chunks is not possible with current technology. This is a crippling constraint, because the time needed to construct a hypermedia information base analogous to a full-word index is currently prohibitive. Our current technique of inserting machine readable index points at the time of recording helps, but it is likely that the major savings in effort will result from advances in speech and pattern recognition technology.

## Data Compatibility

The selection of the host computing environment has given the requirements analyst access to any application program/ processing technique made for the Macintosh environment, one of the most integrated computing environments. In spite of this, we do not have full data compatibility. There is a plethora of incompatible information storage formats. Consider, there are more than fifteen different graphics file formats alone. While this problem can be reduced somewhat by limiting the applications used by the requirements analyst and through a set of file interchange utilities, this must be balanced against the ability to use the most appropriate tool or application. Information gained with existing Macintosh-based tools must be shared between other applications or risk loosing the high degree of integration.

## ACKNOWLEDGMENTS

### Economic Considerations

Applying hypermedia-based technology to a given requirements analysis exercise currently requires expensive hardware including: optical storage technology, audio/video playback and recording equipment, and multi-synchronous display technology. Add these to the time required to learn the equipment and it's clear the price of using hypermedia is high. Until we can collect empirical data, it will be difficult to evaluate the increased efficiency and effectiveness of the efforts of requirements analysts resulting from the application of hypermedia technology. For the time being it will not be cost effective to use these techniques in situations where the requirements are straight forward. Favorable cost/benefit analyses must demonstrate obvious software productivity benefits.

### Social Factors

From a social perspective, recording technologies can be intrusive. Participants sometimes feel inhibited just knowing they are being recorded. In addition, hypermedia technology is simply unfamiliar to most analysts. The process of adding key words to otherwise inaccessible information can lead to over-interpretation of trivial events, communications or other information. Manual indexing of non-textual information is fatiguing. (Although, perhaps not as fatiguing as the current processes.)

## CONCLUSIONS

Demonstrating the feasibility of hypermedia-based support for requirements analysis could largely satisfy the need for integrated techniques supporting the elicitation, analysis, validation, and maintenance of software requirements. Sufficient to say hypermedia has significantly reduced barriers to requirements information collection. However this is not enough, the question is what "flavor" tools and techniques do analysts need to effectively and efficiently organize, manage, and analyze information to produce quality software. It is also clear hypermedia is valuable in situations where access to the requirements information is limited. In addition, hypermedia-based support for requirements analysis offers the analyst an edge in situations involving support for:

- great ambiguity - where the problem and or solution cannot initially be described procedurally;
- great amounts of requirements information (subject to current technological limitations);
- great complexity - where current, text and static graphic-based requirements analysis tools and techniques cannot adequately capture and represent the issues involved;
- software systems specifically concerned with sound-based, motion-based requirements;
- multimedia or hypermedia-based software systems; and
- situations where the sponsor requires an early and complete visualization of the proposed solution.

Technological challenges associated with the technology fall into several categories the authors propose as focus for further research in this area. These include: information management, direct support for requirements analysis, specific technological constraints, data compatibility issues, economic considerations, and social factors.

Portions of this research were sponsored by Apple Computer and the Center for Software Systems Engineering at George Mason University. Portions of this paper benefited from the observations of Eugene Lubchenko, a Principle Research Scientist at Philips Laboratories in Briarcliff Manor, NY. who noted an important gap in a presentation of some of the material from this paper. Dr. James D. Palmer, Associate Dean of the School of Information Technology and Engineering at George Mason University deserves special recognition and thanks for his ongoing participation and sponsorship of this line of research. Finally, the comments of Lee Scott Ehrhart also helped to improve the quality of this paper .

Apple®, HyperCard®, and Macintosh® are registered trademarks of Apple Computer, Inc. HyperTalk, Stackware are trademarks of Apple Computer, Inc. SuperTalk, SuperCard, and SuperEdit are trademarks of Silicon Beach Software, Inc.

## REFERENCES

- [1] Aiken, P., *A Hypermedia Workstation for Requirements Engineering* (dissertation), George Mason University, 1989.
- [2] Andriole, S. J., *Handbook for the Design, Development, Evaluation, and Application of Interactive Military Decision Support Systems*, Marshall VA: International Information Systems, 1986.
- [3] *HyperCard User's Guide*, Apple Computer, 1988
- [4] Bolt, R. A., *The Human Interface: Where people and computers meet*, Belmont, California: Lifetime Learning Publications, 1984.
- [5] Brooks, F. P., "Grasping Reality Through Illusion - Interactive Graphics Serving Science," *Proceedings: ACM/SIGCHI Conference on Human Factors in Computing Systems* May 1988, pp. 1-10.
- [6] Bush, V., "As We May Think," *Atlantic Monthly* July 1945, pp. 101-108.
- [7] Conklin, J., "Hypertext: An Introduction and Survey," *IEEE Computer* September 1987, 20(9):17-41.
- [8] Davis, A., *Software Requirements*, New York: Prentice-Hall, 1990.
- [9] Engelbart, D., *Augmenting Human Intellect: A Conceptual Framework* Summary Report, Stanford Research Institute, on Contract AF 49(638)-1024, October 1962, 134 pp.
- [10] Samson, D., "EXTEND: Automated Assistance for Test Plan Generation," Sonex Enterprises, 1989.
- [11] *SuperCard User Manual*, Silicon Beach Software, 1989.
- [12] Webster, D. E., "Mapping the Design Information Representation Terrain," *IEEE Computer* December 1988, 21(12):8-23.