N93-22195

Operator Performance Support System (OPSS)

Marlen Z. Conklin (Systems Engineer)
Naval Command, Control, and Ocean Surveillance Center
Research, Development, Test and Evaluation Division
Code 793, 271 Catalina Blvd. San Diego, CA 92152-5000
(enrolled in the Doctor of Science in Information Systems at NOVA University)

ABSTRACT

In the complex and fast reaction world of military operations, present technologies, combined with tactical situations, have flooded the operator with assorted information that he is expected to process instantly. As technologies progress, this flow of data and information have both guided and overwhelmed the operator. However, the technologies that have confounded many operators today can be used to assist him -- thus the Operator Performance Support System. In this paper we propose an operator support station that incorporates the elements of Video and Image Databases, Productivity Software, Interactive Computer Based Training, Hypertext/Hypermedia Databases, Expert Programs, and Human Factors Engineering. The Operator Performance Support System will provide the operator with an integrating on-line information/knowledge system that will guide expert or novice to correct systems operations. Although the OPSS is being developed for the Navy, the performance of the workforce in today's competitive industry is of major concern. The concepts presented in this paper which address ASW systems software design issues are also directly applicable to industry. They will make a dramatic impact on the way we work in the future, both the military and the industry. The OPSS will propose practical applications in how to more closely align the relationships between technical knowledge and equipment operator performance.

INTRODUCTION

As the basis of our society moves from the Agricultural, Industrial, and the Information Ages into the Knowledge Age it becomes apparent that the work environment must follow these progressions. The Navy's Combat Systems that were developed some years ago attempted to integrate information and systems, and provided the operator with the opportunity to successfully perform expected tasks. However, with extra capabilities, options, and "nice to have" features added to these basic systems, the flow of data and information to the operator was vastly increased. The resultant more complex systems and equipments have flooded the operator with assorted information that he is now expected to process instantly. Combat systems do not allow for performance that is less than perfect.

The purpose of this paper is to describe an on-line or independent support station that focuses on the information/knowledge available to the Anti Submarine Warfare (ASW) systems operators onboard U.S. Navy ships. The OPSS will systematically incorporate modern informational display techniques and tools so the operator has assess to the proper information he needs at the time he needs it. By providing the operator with this support he will respond more predictably and properly to the requirements of the situation as it is presented to him at his combat systems position. Today's systems must be directed towards user-oriented environments. The design of the OPSS must allow the operator to freely interact with all the CBT, hyperdocumentation and other materials available within the system. The user must have access to appropriate information to accurately integrate a broad range of processing functions.

The OPSS station hardware is identified as Device S10H7. This computer based hardware is configured with Intel 486/33 CPU, 1.2MB 5.25" floppy disk drive, 1.44MB 3.5" floppy disk drive, CD-ROM disk drive, 520 MB Internal Hard Disk Drive, 32 MB RAM, 17" color monitor, Keyboard, Track Ball, Stereo Ear Phones, Video Graphics Accelerator card, Sound Blaster Pro board, DVI playback board and modem. The system is the latest in technology as applied to a multimedia support and training delivery systems for U.S. Navy ships. Device S10H7 is designed to withstand the rigors of use onboard ships and meets all ELF/VLF low emission standards, EMI/RDI low radiation standards, and is certified safe for use. The software is MS-DOS-based, supports Microsoft Windows 3.1 and will be delivered on CD-ROM discs. These standardized commercial programs provide flexibility to support existing and future courseware and hardware enhancements.

In the development process of the OPSS software, one or more members of the end user community were included. Users' members have contribute valuable prospective during system design, assisted in the Fleet introduction of OPSS, and provide feedback to the OPSS design engineers during early production. They are also responsible for the ongoing system "sanity checks" as well as maintaining focus on the operator.

This paper will address the following steps of the systematic implementation of an operator centered workstation system:

- Knowledge acquisition
- Selection of information presentation
- Development of models
- Database system
- Development of OPSS

KNOWLEDGE ACQUISITION

The objective of knowledge acquisition is to identify all the information and knowledge that the operator must possess or have available to perform satisfactorily. For selecting data, one should ask "if I am the operator, what information is needed to get the job done?".

For centuries written material has been "the" conduit of information. We all know what a book or a manual looks like and how to use it to obtain information. One method of knowledge acquisition which OPSS utilizes is to use existing manuals and documentation and, with the help of the end user community, develop an outline of operation and maintenance of ASW equipment. This outline includes a Table of Contents, an Index, and a Glossary. The Table of Contents provides the information and knowledge domain in a hierarchical, linear format. However, the advantage of the OPSS system is that it is computer based, and therefore, information does not need to be linear! Items from the Table of Contents are "linked" and replicate "book browsing" behavior. One way to link the different contents of information from the Table of Contents is to develop a concept map in the format of a tree map. Tree map graphically depicts the information architecture. This is a useful technique to graphically represent meaningful linking relationships between two or more concepts. It also represents the relationship among concepts and the relationships across levels of the hierarchy. Tree mapping possesses a structural plasticity and externalizes concepts and propositions as they are organized in the mind. Changes to the system do not present problems as the open architecture style of development can be easily modified. Concept mapping is also a good technique for negotiating a problem domain with another individual. Tree-maps are designed for visualization of the hierarchical structure, and linking of knowledge structures and their relationships. An example of the mapping, showing simple hierarchical and linking formats, is shown in Figure 1.

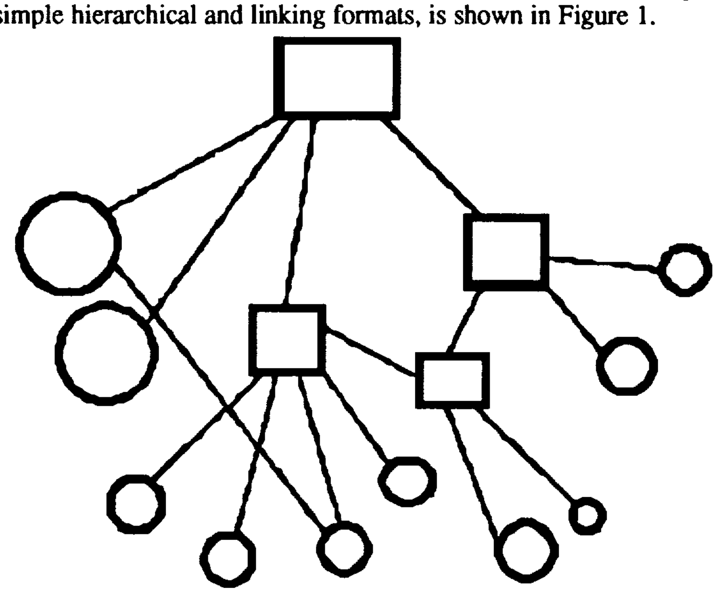

Figure 1. Concept tree mapping

Tree mapping is also very useful to convert the information into a model suitable for a system display such as the menus display format. Menus relate to user needs or problems, different user circumstances, different models of information, and allows the operator the flexibility to exit from one point and to link a different section of the system. For example the OPSS allows the operator to perform maintenance with the support of the on-line electronic maintenance documentation. If at any time he has problems performing a specific task, he may call up an interactive computer based training (CBT) which will explain how to perform the task.

The pragmatic user oriented approach is used to develop this system. Only what is useful for the operator is extracted and retained. Nice to have flexibility is scrutinized. If it does not meet operator needs, it is deleted. Needless information and knowledge are therefore discarded. What the system retains is limited to essential functions, and is justified by the end user. We keep it simple and efficient. The boundaries of information are not absolute because there is a modem network capability that establishes, with some operational restrictions, a network interface between U.S. Navy ships, school houses, developers and sponcers. The concept is to start with an essential kernel of information/knowledge and only add to the system if the requirement is established by the field operators.

The knowledge represented by concept mapping may be classified as formal rule-based processing of information, or it may be represented by non rule-based processing. Rule-based knowledge is the foundation of expert systems. The OPSS therefore has a rule-based front end that is designed to quickly navigate the operator through the hierarchy of menus. Non rule-based knowledge is utilized in the hypertext and hypermedia systems of the OPSS. Hypertext and hypermedia are used a vast diversity of conceptual frameworks and can be referenced while using metaphors, similes, analogies, diagrams, images, animations, sound and video all of which an operator can use but the computer cannot. Consequently the graphic user interface (GUI) representation of the ASW OPSS is an initial menu selection that eventually links the operator with hypermedia documentation or interactive CBT.

SELECTION OF INFORMATION PRESENTATION

Selected information will be presented in various media. The ASW OPSS programs incorporates elements of text, graphics, animation, picture, sound, video and software modules, while OPSS hardware is the latest in available technology applied to a multimedia workstation for U.S. Navy ships. Members of the end user community have reviewed the media selection process and have helped determine how the information should be expressed to support the operator's needs. The paradigm of how to represent knowledge is defined by the domain experts. We have attempt to minimize the incongruity between the developers and the system operators, with the objective of aligning relationships between technical knowledge and equipment operator.

Information and knowledge of the OPSS may be presented in several formats, such as electronic documentation, information retrieval system, hyperdocuments, content sensitive help, on-line advisory, interactive computer based training, simulation and scenario playback capability. Those formats may be references, advisor, or tutor as the operator works to solve his problem.

Operator needs will vary depending what function is being performed. For example the operator may be assigned to a surface combatant Maintenance Division, or may be involved in ship-wide training mode. Information to support those activities is time dependent and requires a different treatment then if an operator needed to perform a specific task immediately. Again we try to be pragmatic and apply the user oriented approach for the development of the system. A conscious effort has been used to keep the system tight and simple. Whatever becomes incorporated into the system must help the operator's performance. The system is therefore performance-oriented, rather than information or transaction-centered.

DEVELOPMENT OF MODELS

Models of the OPSS interface are built with simple application software and are empty shell representations of what the final system display will be. The models are based on the knowledge acquisition process and the proposed selection of information presentation. They are representations of the proposed GUI and how the human/computer interaction will be executed. There are two purposes for the development of models. One is for the end user community to verify the useability of the designed item and to ensure that the user's perspective has been taken into consideration before system development. The second purpose is for system requirement definitions to be used by the developers.

Design characteristics involve typical forms of computer interactions such as menu selection, command manipulation, forms fill and direct manipulation. Elements are accessed directly by offering hot spots in the displays such as a word, a group of words, a marked area in a picture and jump ahead command. Tools for information search are provided so the operator may search for information using words, combination of words and multiple selection. Nontrivial feedback dialogue are provided only when needed. The design is operator centered. With design features that allow the operator's logical intuitive interaction with the system. Motivational factors such as attention, relevance, confidence and satisfaction have also been considered. The operator will become involved and will be confident that progress towards his goals are being made.

DATABASE SYSTEM

The database system definition is performed by a systematic analysis and definition of the specification of the database management system (DBMS), the database (stored data) requirements, and the complete set of application programs (tools) used in the OPSS.

Database management systems (DBMS) have proven to be cost-effective tools for organizing and maintaining large volumes of data in the OPSS. Its primary function is to store data and provide operations on the databases. The operations required for OPSS are: create, delete, update and search (ad hoc query) of data. OPSS data processing requires databases that store large quantities of information having complex structures. A database schema or class hierarchy is developed describing the logical structure of the database supporting the overall system design, the relationship between individual components, and the operations that must be performed. We had to analyze the relative merits of relational and object-oriented database management systems (RDBMS and ODBMS) and the available commercial DBMS. A systematic approach to this analysis entailed a careful evaluation of needs, and how well those needs were met by available products. An ODBMS was eventually chosen for this project.

OPSS ODBMS is able to handle inheritance linking, polymorphism, run time binding, dynamic binding, ad hoc query, security and semantic integrity. It also has a seamless integration to C++ programming language interface. The application programming interface (API) adheres to the industry standards for this language. The ODBMS has a database browser, debugger and a graphical schema design utility that allow for quick design and modification of the database schema.

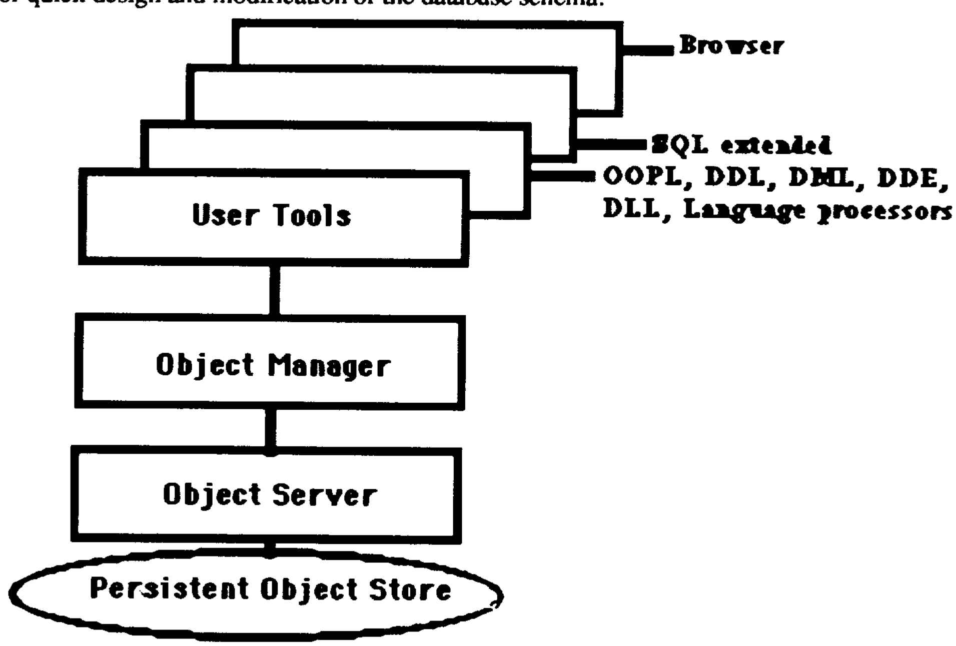

Figure 2. Overall Architecture of the OPSS ODBMS

terminology: Object -oriented Programming Languages (OOPL)
Dynamic Link Library (DLL)
Data Manipulation Language (DML)
Dynamic Data Exchange (DDE)
Dynamic Link Library (DLL)

DEVELOPMENT OF OPSS

The system is developed in modules. Most of those modules are developed with commercial available application software or tools. Many of the OPSS application software are Macintosh based, however the OPSS user program is Windows 3.1 on an MS-DOS operating system. This has not presented any problems because software is available to allow Macintosh files to be ported to the Windows MS-DOS environment. Those modules are then encapsulated with their internal state hidden, to be called up by the ODBMS. Intuitively one may visualize the encapsulation of an object as making it into a "black box" and the DBMS as a pointer that calls on it's functionalaties .

The ongoing OPSS development effort is performed by a group of specialized experts. The developer has the details of the requirement definition established by the concept tree mapping and from the models described earlier. As the modules are developed, the end user participates in the operational test and evaluation process. The implementation of OPSS is being performed in batches.

CONCLUSION

New technologies have become available to us today that provide us with tools to develop a system with complex databases. Those complex databases are composed of information and knowledge in an array of media. Today's database management system allows information and knowledge to be retrieved and manipulated quickly. By incorporating members from the end user community in the project, and by developing the system in an "end-user environment", OPSS promises to provide the operator with the electronic support required to allow him improved performance in the complex and fast reaction world of the military.

BIBLIOGRAPHY

Chin, John P. and Norman, Kent L.. "The menu metaphor: food for thought". Behavior and Information Technology, 1989, Vol. 8, No. 2, p. 125-134.

Colley, Grant "Retain the object model and retain the benefits". Object Magazine May/June 1992, p.27-28.

Dadrowski, Christopher E., Fong, Elizabeth N. and Yang, Deyuan. Object Database Management Systems: Concepts and Features. National Institute of Standards and Technology Special Publication 500-179, April 1990.

Deitel, H. M. and Kogan, M. S.. The Design of OS/2. Addison-Wesley: Reading, MA, 1992.

Gancarz, Robert and Colley, Grant "Consideration for evaluating object database management systems". Object Magazine March/April 1992, p. 57-61.

Gery, Gloria J.. Electronic Performance Support System. Weingarten Publication: Boston, 1991.

Interrante, L. and Biegel, J. (1989). "Automatic Knowledge Acquisition for an Intelligent Simulation Training System". Proceedings 1989 Annual Conference of the International Association of Knowledge Engineers.

Keller, John M.. "Development and Use of the ARCS Model of Instructional Design". Journal of Instructional Development, 1987, Vol. 10, No. 3, p. 2-10.

Laverson, Alan, Norman, Kent and Schneiderman, Ben. "An evaluation of jump-ahead technology in menu selection". Behavior and Information Technology, 1987, Vol. 6, No. 2, p. 97-108.

Martin, James. Hyperdocuments and How to Create Them. Prentice Hall: Englewood Cliffs, New Jersey, 1990.

Mandelkern, Dave. "Visual programming". Object Magazine September/October 1992, p. 39-43.

Mills, Carol B. and Weldon, Linda J.. "Reading Text from Computer Screens". ACM Computing Surveys, 1987, Vol. 19, No. 4, p. 329-358.

McNeese, Michael D. and Zaff, Brian S.. "Design Acquisition: Translating User Knowledge into Design Solutions". Proceedings of the Human Factors Society - Interface 1991, (p. 42-49).

McNeese, Michael D., Zaff, Brian S., Peio, Karen J., Snyder, Daniel E., Duncan, John C., McFarren, Michael R. An Advance Knowledge and Design Acquisition Methodology: Application for the Pilot's Associate. Defense Technical Information Center AD-AA233 700. Armstrong Aerospace Medical Research Laboratory Technical Report # 90-060.

Moshiri, E. (1989). "An Integrated Knowledge Acquisition Environment". Proceedings 1989 Annual Conference of the International Association of Knowledge Engineers.

Norman, Kent L., Weldon, Linda J., and Shneiderman. "Cognitive layouts of windows and screens for user interfaces". International Journal Man-Machine Studies, 1986, Vol.25, p. 229-248.

Shneiderman, Ben. "Designing Menu Selection Systems". Journal of the American Society for Information Science, 1986, Vol. 37, No. 2, p. 57-70.

Taylor, David "The coming convergence of object and relational databases". Object Magazine September/October 1992, p. 16-18.

Vasan, Robin "Integrating object and relational databases". Object Magazine July/August 1992, p. 59-61.