SPAWAR
Systems Center
San Diego

TECHNICAL REPORT 1822
July 2000

Open Systems Advanced Workstation Transition Report

H. Ko

Approved for public release;
distribution is unlimited.

SSC San Diego

TECHNICAL REPORT 1822
July 2000

Open Systems Advanced Workstation Transition Report

H. Ko

Approved for public release;
distribution is unlimited.

SPAWAR
Systems Center
San Diego

SSC San Diego
San Diego, CA 92152-5001

SSC SAN DIEGO
San Diego, California 92152-5001

Ernest L. Valdes, CAPT, USN
Commanding Officer

R. C. Kolb
Executive Director

ADMINISTRATIVE INFORMATION

The Open Systems Advanced Workstation (OSAW) project was initiated in October 1995 and completed in September 1999. The sponsor is the NAVSEA PMS 440 Processors and Displays Systems Division. Funding was provided by the Office of Naval Research Human Systems Department under program element 0603707N. The work detailed in this report was performed by the OSAW project team of the Simulation and Human Systems Technology Division (D44) of the Command and Control Department (D40) of SSC San Diego, Pacific Science & Engineering, and Carlow International, Inc.

Released by
R. J. Smillie, Head
Collaborative Technologies Branch

Under authority of
J. L. Martin, Acting Head
Simulation and Human
Systems Technology Division

ACKNOWLEDGMENTS

This study was conducted with the assistance and cooperation of the Project Officer Mr. Percy Tolbert from NAVSEA PMS 440. The author would like to thank the OSAW project team members who worked in specialized areas:

Human Computer Interface:	Dr. Glenn Osga, Ms. Nancy Campbell, Mr. David Kellmeyer, and Mr. Jack Houghton
Display Technology:	Mr. Rick Worthen
Speech Recognition:	Mr. Dan Lulue
3-D Audio:	Dr. Jerry Kaiwi
Touch:	Mr. Tom Enderwick

SB

EXECUTIVE SUMMARY

U. S. Navy Command and Control systems require complex task support from shipboard workstations that receive information from different sources. For future workstations, it is expected that information displays will use a multi-modal interface. Operator multi-modalities involve touch and voice inputs with visual and 3-D auditory outputs. This report describes the development of the Open Systems Advanced Workstation (OSAW) and presents guidelines for using multi-modal technologies.

The OSAW was developed to conduct research for the next generation of U.S. Navy Command, Control, Communications, Computers, and Intelligence (C^4I) system workspaces. Workspace hardware and software will require careful integration to meet operators' needs. The goal of OSAW was to implement a user-centered design for a next-generation workstation with the integration of commercial displays, input devices, and software. Studies and analyses were completed in the following areas: (1) task analysis and modeling of human–computer interaction modalities, (2) evaluation of multiple displays in a multi-tasking environment, (3) ergonomic assessment of workstation design, and (4) development of design guidelines for touch screen, speech recognition, and 3-D sound localization technologies.

ERGONOMIC WORK STATION DESIGN

The OSAW Workstation is designed to accommodate research and testing of design parameters for operator interactions and resulting performance. The Workstations also meets the existing criteria of MIL–STD–472.

The OSAW addresses the following other problem areas:

- Ergonomic arrangement of displays and controls. (Guidelines are needed for development of ergonomic workstations under these conditions.)
- Optimum design for the largest proportion of the population ranging from the 5th percentile female to 95th percentile male in reach, viewing distance, and visual angle.
- Need for flexibility in changing mission demands such as the increased task demands from non-lethal to low-intensity through major regional conflict planning, monitoring, and execution.
- The shift from individual to collaborative decision-support tools requiring an adaptable workstation hardware, software, and ergonomic architecture that accounts for the needs of small-team interaction.

The OSAW Workstation, when fully extended in all directions, is 60 inches wide and 36 inches in depth with a height of 53 inches. While the base of the horizontal row of three displays is fixed at 31 inches in height, the keyboard tray has a variable 45-degree tilt and can be pushed forward for storage. All displays can be tilted vertically toward or away from the operator through 45 degrees of angle. The two side displays can also be rotated toward or away from the operator's position up to 45 degrees. The footrest is also adjustable to accommodate operators of different statures.

The four displays can be used as one integrated display surface (i.e., as if the physical separations of the display units did not exist or each display could be an independent display surface). The displays can also be configured in various combinations of display surfaces (e.g., the center and right side displays can be one display surface and the remaining two displays each could be a single display surface).

TASK ANALYSIS AND MODELING

The task analysis and modeling was done to (1) identify benchmark sequences of a typical C^4I mission area (strike coordination), (2) analyze OSAW Human Computer Interface (HCI) operations using GOMS (Goals-Operations-Methods-Selection) techniques, (3) develop a prediction model for benchmark tasks, (4) perform trial runs of the model, and (5) assess techniques for further application. This work was accomplished by using GOMS task analysis techniques, modeling techniques associated with GOMS, and a suitable model for evaluation. The resulting assessment method was suitable for general HCI tasks, and for specific operational tasks such as Strike Coordination.

Time-event task network software was used to produce a hybrid model, combining the Cognitive-Perceptual-Motor GOMS (CPM-GOMS) analysis into a computer simulation of the strike coordination tasks. This model was developed using MicroSaint IBM PC DOS-based software. CPM-GOMS is based on the Model Human Processor (MHP) and is divided into three interacting subsystems: the cognitive system, the perceptual system, and the motor system.

The model contained two types of top-level tasks: (1) an HCI event task whose time is determined by the HCI activity in the model, and (2) other operator tasks (such as read, decide) that are modeled with a fixed estimated time. Consequently, over time, differences would be because of HCI modal differences and not other human task variability.

The MAUI (Model for Analysis of the User Interface) produced three types of measures: Productivity Measures (measurable as the number of tasks completed), Workload Measures (a matrix of transitions; e.g., the number of HCI events per unit of time), and HCI link measures (a matrix of transitions; e.g., the number of HCI events for which the hand moved from the mouse to the keyboard, etc.).

The GOMS analysis shows that multi-modal HCI has a strong potential over conventional workstation design. With such a model, the design can be optimized in minimum CPM workload, maximum productivity, and best efficiency for a given task scenario.

EVALUATION OF MULTIPLE DISPLAYS

Two experimental evaluations were performed on various aspects of multi-monitor workspace designs. The evaluation of multi-monitor workspace designs consisted of multiple monitors and virtual workspaces.

One of the most serious short-comings of current workstations is that they do not provide efficient access to the large amounts of information required for supervision and multi-tasking because each task involves multiple application settings (e.g., AN/UYQ-70 Consoles and AEGIS Combat Information Center, etc.). There is a practical limit to the amount of screen space that can be used effectively on a given monitor. Additional monitors place information further away from the center of the workstation, increasing the number and size of movements. One alternative solution is to increase screen area through larger monitors. Another solution is to provide virtual workspaces (screens of information) or virtual desktops such that several workspaces can be brought successively into view on either a single monitor or multiple monitors.

The two experiments were performed to evaluate different workstation designs for various user tasks. In experiment 1, the evaluation included alert perception and display monitoring in a dual-task situation. Four workspaces were employed and they were presented on one monitor with virtual workspaces, two monitors with virtual workspaces, and four monitors without virtual workspaces. The switching interface consisted of a hot-key-operated workspace control diagram with indicators for alerts. In experiment 2 we evaluated and compared multiple display configuration combining displays and virtual workspaces on a number of common human-computer interaction operations, such as finding and accessing workspaces, transferring information and monitoring.

Experiment 1 supported the hypothesis that having only one monitor degraded performance compared with two monitors. However, the more interesting finding may be that there was little difference between the two-monitor condition and the four-monitor condition in either tracking performance or alert detection. If anything, having only two monitors plus virtual workspaces with a switching interface allowed for better monitoring performance because of the enhanced workspace control diagram (red light alerts were indicated on the diagrams as well as next to the gauge) and the fact that the more peripheral monitors were not used.

Experiment 2 results indicated that fewer monitors support better performance for tasks that involve frequent information transfer and monitoring. These findings support the use of the 2 horizontal or 2 vertical workstation configurations rather than either 3 horizontal or 4 horizontal/vertical combination ones.

These multiple display studies suggest that two monitors with virtual workspaces enhanced by a hot-key-operated workspace control diagram affords optimal performance across a variety of common multi-tasking environments.

DESIGN GUIDELINES DEVELOPMENT

There are three capabilities developed to complement the visual display. The three state-of-the-art technologies include touch screens, speech recognition and 3D audio localization.

Other than voice recognition, touch input is probably the most natural human interface to any computing device. It is particularly useful and popular in those applications where the user is relatively unskilled in the operation of computer input devices.

The five most common touch screen technologies include Near Field Imaging (NFI), Capacitive, Infrared, Resistive, and Surface Acoustic Wave (SAW). Each technology offers its own unique advantages and disadvantages. SAW touch screens were integrated with the FPDs in OSAW to evaluate pressure sensitivity. In addition to the X and Y coordinates, SAW technology can also provide Z-axis (depth) information or pressure sensitivity. SAW technology is the latest of the touch input technologies and uses inaudible acoustic waves traveling over the surface of a glass panel at precise speeds in straight lines.

An exploratory experiment was conducted to evaluate the use of an operator's touch to dual-activate a given on-screen button. The question arose as to whether or not the differential of an operator's finger pressures could be used to activate the same on screen button for two functions. This would be accomplished by having the operator press a button either softly or hard for two respective functions, whatever they may be. Individuals have a personal perception as to what constitutes a soft or hard touch for whatever purpose that sense will be used. How good is that perception? Can people be trained to decrease the variation among individuals in their perception of hard and soft? These were the questions addressed in this exploratory experiment.

We found out that operators are not able to activate more than two levels. The analysis of the data indicated that training did not result in a significant improvement in performance.

The speech field encompasses topic areas that range from baseline feature extraction of the speech signal via digital signal processing (DSP), to speaker and language identification, to speech recognition and synthesis, to natural language discourse systems. In general, speech technologies are not as mature or as well performing as business software (word processing, spreadsheets, databases, etc). Technologies that support interaction between a human and a computer via speech have great promise, but they are still too unreliable and immature for wide deployment in the commercial domain. There are still too many unanswered questions about what makes an effective speech interface, and about what the metaphors and paradigms are. In other words, there is not yet an accepted concept of operations for how a user speaks to a computer interface, be it the desktop, an application, or an agent. In the military computing environment, even less is known about how to build software with speech technologies.

The current state of speech technology is an odd mixture of research projects, COTS dictation products, deployed single-purpose telephony systems, and notional natural language systems. To a large extent speech is an immature technological solution looking for a problem. Work in the area has been driven not by a systematic analysis of the requirements, but rather on the idea that people speak to each other, they should be able to speak to their machines.

Taking an ad-hoc, non process-oriented approach to the design and development of an entire technology leads to the same result as when it is done with a system or with an application. One ends up with a collection of stand-alone things, some that work reasonably well in dictation and test-to-speech (TTS), some that show promise in speaker identification, and others that need a lot more work in natural language.

For years the holy grail of the speech development community was speaker-independent, continuous recognition. Large resources were deployed to solve the problem, and the result was remarkably effective DSP techniques optimized to the problem. Once this was accomplished, the belated question of "what is this good for?" was addressed. That is how we got to where we are—dictation products that do a remarkably good job of translating human speech to text, and that are most appropriately used by individuals with physical handicaps. Neither the average typist nor the computer power user considers dictation an effective way to interact with the windowed desktop or with an application.

Essentially, the individual technologies were not originally designed to complement each other, or to work well together. This is easily seen in the various ways that developers have tried to retrofit automatic speech recognition (ASR) to the desktop metaphor, and to business applications. The desktop metaphor does not work well with speech because speech cannot compete with the efficiency and convenience of the keyboard and the mouse. Similarly, speech is particularly ineffective in executing atomic application features that are better accessed via key shortcuts.

Rather than retrofitting speech recognition to existing keyboard/mouse user interfaces, we need to rethink how best to design computer interfaces so that speech is one of several equally effective input and output modalities. The keyboard and mouse reign supreme in the desktop metaphor, which in itself does a very poor job of providing an intuitive interface. Thus, rethinking and redesigning the HCI from scratch would be a very productive effort. The designers would be able to learn from the past, would be able to apply a modern software engineering process, would be able to design so as to not preclude accommodating future, unanticipated advances in computing capabilities.

For the time being, it is important to keep these considerations in mind when deciding at the outset of a software development effort what the "ins" and the "outs" will be. Designing from the outset with speech, and other I/O modalities in mind, is of critical importance to the ultimate success of all future projects.

The use of spatialized 3-D audio can increase the task-related information made available to operators. Headphone listening is ubiquitous throughout the Navy with pilots, traffic-controllers, flight-deck personnel, fire-control teams, weapons-console operators, sonar operators, etc. They are required to monitor multiple aural channels while simultaneously sending and receiving voice communications and responding to system generated auditory alarms and instructions, often in the presence of interfering ambient noise. But, current headphone technology is clearly deficient in terms of the information processing requirements of these tasks. The effective spatial bandwidth of current Navy headphone technology is limited to the region between the two ears of a listener. Consequently, current headphone displays consist of only two or three auditory channels, far below the number of auditory information sources. This problem is dealt with by either selective filtering via a switchboard device, or simply adding multiple headphone sets and/or speaker systems and letting the listener deal with the resulting cacophony. In modern Fleet systems, headphone based displays have become significant information-processing bottlenecks that severely constrain system performance. Headphone delivered synthetic 3D audio is an enabling technology for meeting reduced manning requirements while simultaneously maintaining or improving system performance. The advantage offered is that it provides headphone listeners with auditory spatial cues comparable to those heard under natural listening conditions. In effect, 3D audio synthesis technology promises to provide headphone listeners with a virtual anechoic chamber that includes multiple virtual sound sources mimicking physical speaker devices. Such a virtual three-dimensional sound-field can significantly improve the ability of listeners to process multiple auditory information sources and maintain a new and better level of situation awareness.

This report describes the Open Systems Advanced Workstation (OSAW), the research that was accomplished by using it, and the subsequent guidelines that were developed based upon that research. The capabilities inherent in the OSAW will enable console designers managers to exercise design options in controlled settings. All the major human modalities, visual, tactical, auditory and speech, can be evaluated for proposed workstation designs. This will provide the means to optimize operator interface designs for shipboard applications and contribute toward reduced manpower requirements.

CONTENTS

APPENDICES

Figures

Tables

1. INTRODUCTION

U.S. Navy Command and Control systems will require complex task support from shipboard workstations that can receive information from various sources and display that information by using multiple modalities of the human operators. The specific multi-modalities of current interest involve touch and speech inputs and three-dimensional (3-D) auditory outputs combined with advanced displays such as Flat Panel Displays (FPDs). Guidelines are needed to exploit these technologies for various mission-related activities in future Command, Control, Communications, Computers, and Intelligence (C^4I) systems. This report describes the research on flat-panel and multiple displays, touch screens, speech recognition systems, 3-D audio localization, and the integration of the results into a workstation (i.e., the Open Systems Advanced Workstation [OSAW]) using state-of-the-art Commercial Off-the-Shelf/Government Off-the-Shelf (COTS/GOTS) hardware and software to support the shipboard Command and Control task environment.

1.1 PROBLEM/DEFICIENCY

The advent of open system architecture and commercial workstation components present numerous configuration options to the system acquisition manager. The recent shift from custom-designed consoles to open system architecture platforms using COTS/GOTS products might solve timing, affordability, some procurement problems, and reduce maintenance costs while improving end-user performance.

With changing mission demands, the operator is overloaded with visual and aural information from non-lethal to low-intensity through major regional conflict planning, monitoring, and execution. Command and Control missions in future Combat Information Centers will require complex task support from a workstation that can receive information from multiple sources and provide displays in multiple modalities. Guidelines for the multi-modal use of touch and speech input and 3-D audio output in combination with FPDs are needed to exploit these technologies for various tasks in future C4I systems.

Display and control arrangement is one major workstation problem area. Current workstations are packed with displays, controls, multiple VME card cages, and peripheral devices such as communication panels and power supplies. The weight, bulk, and maintenance requirements are based upon the entire suite of equipment located in the console enclosure. The ergonomic workstation should be adjustable to support the 5th percentile female to the 95th percentile male U.S. Navy operators in viewing distance, visual angle, and reach. FPDs and remote racking of console electronics should enable a task-supportive design. Use of multiple FPDs will increase available workspace and, hopefully, improve the performance and efficiency of the operator in multi-tasking environments.

In summary, the OSAW with multi-modality was developed by integrating COTs and GOTs products to meet the needs of the human operators while using their capabilities and offsetting their limitations. OSAW was developed to conduct research for the next-generation U.S. Navy C4I systems.

1.2 TECHNICAL APPROACH

This project focused on the development of OSAW by the integration of commercially available displays, input devices, and software with either a TAC-4 military-specified computer or an IT-21-compliant Windows NT PC. The OSAW is based upon a multi-modal and multi-channel interaction model. The OSAW research studies and analyses included: (1) task analysis and modeling of human–computer interaction modalities, (2) evaluation of multiple displays in multi-tasking environments, (3) ergonomic assessments of workstation design, and (4) development of design guidelines for touch screen, speech recognition, and 3-D sound localization technologies.

1.3 OSAW DESCRIPTION

Table 1 describes the OSAW specification. The initial OSAW was developed in a TAC-4 environment, but the OSAW has been migrated to a Windows NT environment to support IT-21 compliance.

OSAW is designed to accommodate research and testing of design parameters for operator interactions and resulting performance. The workstations also meet the existing criteria of MIL–STD–1472 to adapt to the largest proportion of the population.

Table 1. OSAW specification.

	OSAW – TAC-4 (FY 1996–1997)	OSAW – PC (FY 1998–1999)
Ergonomic Workstation		

Table 1. OSAW specification. (continued)

	OSAW – TAC-4 (FY 1996–1997)	OSAW – PC (FY 1998–999)
Computer	TAC-4: HP-J210 - HP-UX10	IT-21-compliant PC - Windows NT 4.0
Flat Panel Display	Sharp 14" TFT Panel - 1024 x 768 pixels - 8-bit RGB - Viewing angle: 45° (H), 10° (Down), 30° (Up)	NEC 20.1" TFT Panel - 1280x1024 pixels - 8bit RGB - Viewing Angle: 160°
Touch Screen	Caroll Touch Guided Acoustic Wave - Z axis support	Elo Surface Acoustic Wave - Z axis support
Speech	Verbex Speech Recognizer	IBM VIAVOICE
3-D Audio	Crystal River Engineering ACOUSTETRON II	AuSIM Engineering Solutions AuSIM Gold Series

Four 20-inch FPDs are integrated to support the multi-tasking environment. Four displays increase the display workspaces and reduce the footprint and weight compared with conventional Cathode Ray Tubes (CRTs).

Surface Acoustic Wave (SAW) touch screens are mounted on the FPDs. The SAW technology provides pressure-sensitive Z-axis (depth) information.

The Speech interface was developed in Java for the Lightweight Extensible Information Framework (LEIF) and Military Language Processor (MLP) using the IBM ViaVoice speech recognition engine and development tool kits. The IBM ViaVoice supports continuous speech and a large vocabulary. It requires about 30 to 40 minutes of recognition training for users to achieve high performance in Command and Control and dictation modes.

The AuSIM, Inc., Gold Series S101 Audio Vectorization System provides a very high-fidelity 3-D audio synthesis capability for the OSAW. The AuSIM 3-D audio system supports 16 channels of input and 16 channels of output in 44.1-kHz high-quality audio. The 3-D audio interface has been developed in 3-D graphics using Java 3-D. The interface allows users to manipulate sound sources and locate them any place around the user's head.

2. ERGONOMIC WORKSTATION DESIGN

The OSAW is designed to accommodate research and testing of design parameters for operator interactions and resulting performance. The workstations also meets the existing criteria of MIL–STD–1472.

The OSAW addresses the following problem areas:

- Ergonomic arrangement of displays and controls. (Guidelines are needed for development of ergonomic workstations under these conditions.)
- Optimum design for the largest proportion of the population ranging from 5th percentile female to the 95th percentile male in reach, viewing distance, and visual angle.
- Need for flexibility in changing mission demands such as the increased task demands from non-lethal to low-intensity through major regional conflict planning, monitoring, and execution.
- The shift from individual to collaborative decision-support tools requiring an adaptable workstation hardware, software, and ergonomic architecture that accounts for the needs of small-team interaction.

Figure 1 shows the major positioning features that support an ergonomic workstation design. The OSAW, when fully extended in all directions, is 60 inches wide and 36 inches in depth with a height of 53 inches. While the base of the horizontal row of three displays is fixed at 31 inches in height, the keyboard tray adjusts at a 45-degree tilt and can be pushed forward for storage. All the displays can be tilted vertically toward or away from the operator to a 45° angle (figure 1). The two side displays can also be rotated toward or away from the operator's position up to 45°. The footrest also adjusts to accommodate operators of different statures.

The four displays can be used as one integrated display surface (i.e., as if the physical separations of the display units did not exist, or each display could be an independent display surface). The displays can also be configured in various combinations of display surfaces (e.g. the center and right side displays can be one display surface and the remaining two displays could each be a single display surface).

The following subsections cover research areas of design interest that can be tested using the OSAW.

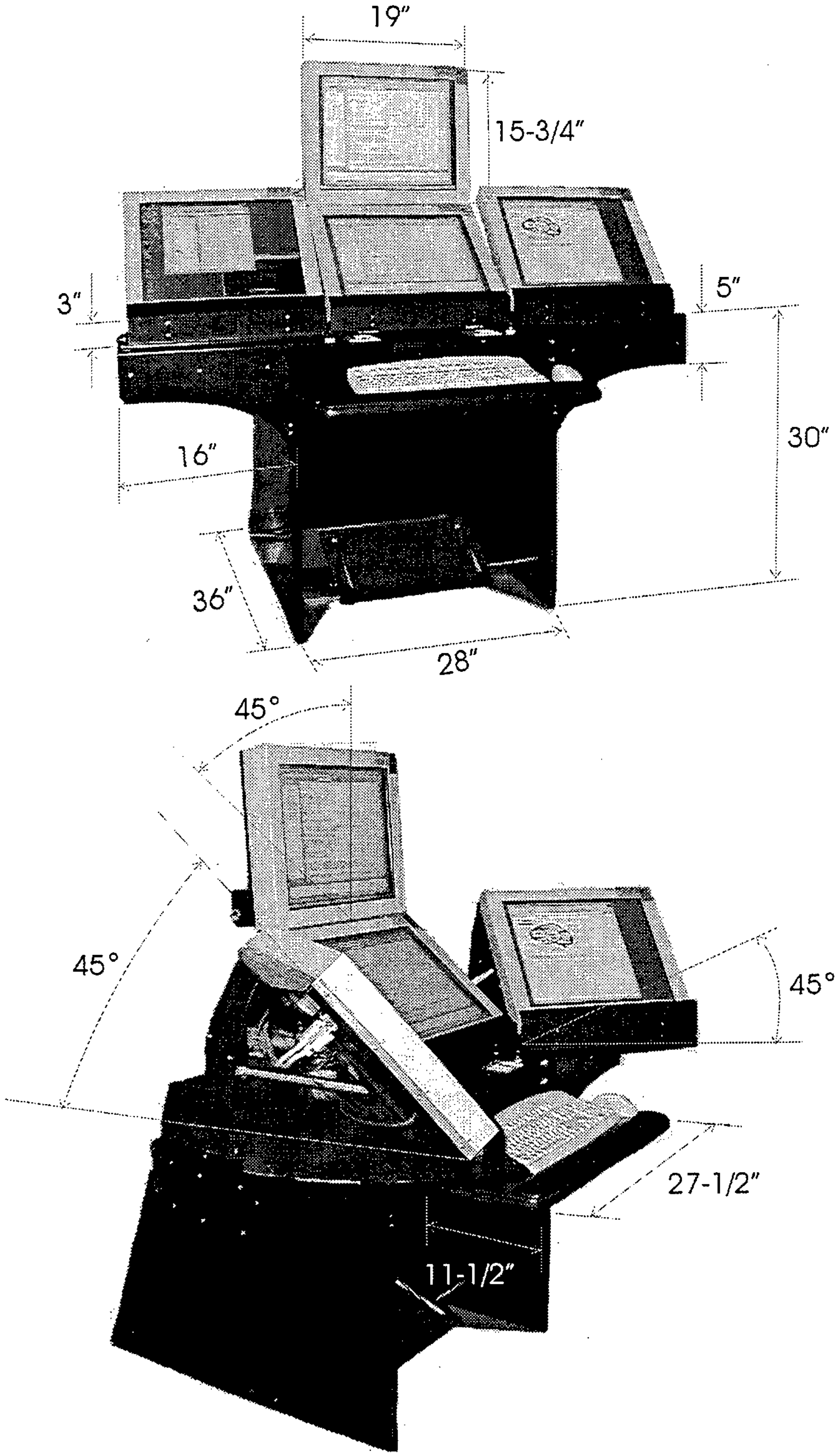


Figure 1. Ergonomic OSAW.

3. TASK ANALYSIS AND MODELING[1]

Modeling techniques associated with GOMS and other modeling techniques produced a suitable evaluation model. The resulting assessment method should be suitable for general human–computer interface (HCI) tasks and specific operational tasks such as Strike Coordination.

The OSAW console includes a range of HCI modes. Multiple-screen visual outputs, and inputs through a keyboard, mouse/trackball, touch screen, and voice recognition, are appropriate for GOMS analysis. However, GOMS techniques do not track display efficacy of spatial auditory output and display metaphors. Spatial auditory output could be included as demands for auditory perception resources, and resulting conflicts could be analyzed; however, this information is not included in this study.

While a general comparison of the various modes on the OSAW console is interesting, this report emphasizes the development of a technique suitable for providing multiple modes for a given task, and the means for selecting the appropriate modes suitable for various situations. Furthermore, each mode may be affected in different ways by external tasks (manual, visual, auditory), and the technique should provide for assessing these effects.

The task analysis and modeling was used to (1) identify benchmark sequences of strike coordination tasks, (2) analyze OSAW HCI operations using GOMS techniques, (3) develop a prediction model for the benchmark tasks, (4) perform trial runs of the model; and (5) assess techniques for further application.

This effort produced a series of benchmark tasks identified for interleaved Strike Coordination Window tasks (preparation and planning for the next day's strikes) and Execution tasks (conducting the current day's strikes). Appendices A through D list these tasks, along with illustrations of the operator windows.

A hybrid Model for Analysis of the User Interface (MAUI) was developed, combining two GOMS (Goals-Operations-Methods-Selection) techniques with a time-event task network model created with Micro-Saint (MSAINT) software. MAUI provided measures for (1) productivity, (2) HCI workload, (3) link analysis, and (4) HCI complexity.

Although significant effort is required to develop MAUI, and more effort is required for validation, the example output indicated that the information produced should be worth the effort.

3.1 TASKS

Benchmark tasks were generated based on Strike Coordination tasks to include a liberal sampling of HCI widgets and interleaved processing of external events. The Strike Coordination tasks generate Tomahawk strikes with various strikes using other weapon systems.

Two types of strike coordination activities are included: (1) a Strike Coordination Window task, which involves preparation and planning for the next days' activities, and (2) a Strike Coordination Execution task. Appendix A presents the Window tasks, and Appendix B presents drawings of the

---

[1] This section is a summary of the report on task analysis and modeling of human-computer interaction modalities (Obermayer, Linville, and Calantropio, 1999).

user interfaces; Appendix C presents the Execution tasks, with the corresponding user interfaces provided in Appendix D.

3.2 GOMS TASK ANALYSIS

GOMS is available as the following family of techniques (Card, Moran, and Newall, 1983; John, 1990; Kieras, 1993):

- CMN-GOMS (Card-Moran-Newell GOMS)
- KLM (Keystroke Level Model)
- NGOMSL (Natural GOMS Language)
- CPM-GOMS (Cognitive-Perceptual-Motor or Critical-Path-Method GOMS)
- Q-GOMS (Quick-and-Dirty GOMS)

This family of GOMS methods was examined. For the current requirement, the GOMS family members considered useful were the Keystroke Level Model, and Cognitive-Perceptual-Motor GOMS (CPM-GOMS).

To predict execution time, GOMS requires the analyst to determine how many memory (cognitive) operations are required, and values for fundamental operation times. These determinations depend on the HCI user's level of expertise, and the analyst must perform empirical testing to achieve confidence in the GOMS.

CPM-GOMS is based on the Model Human Processor (MHP) (figure 2), as introduced by Card, Moran, and Newell (1983). The MHP is divided into three interacting subsystems: (1) the Perceptual System, (2) the Motor System, and (3) the Cognitive System (each with its memories and processors).

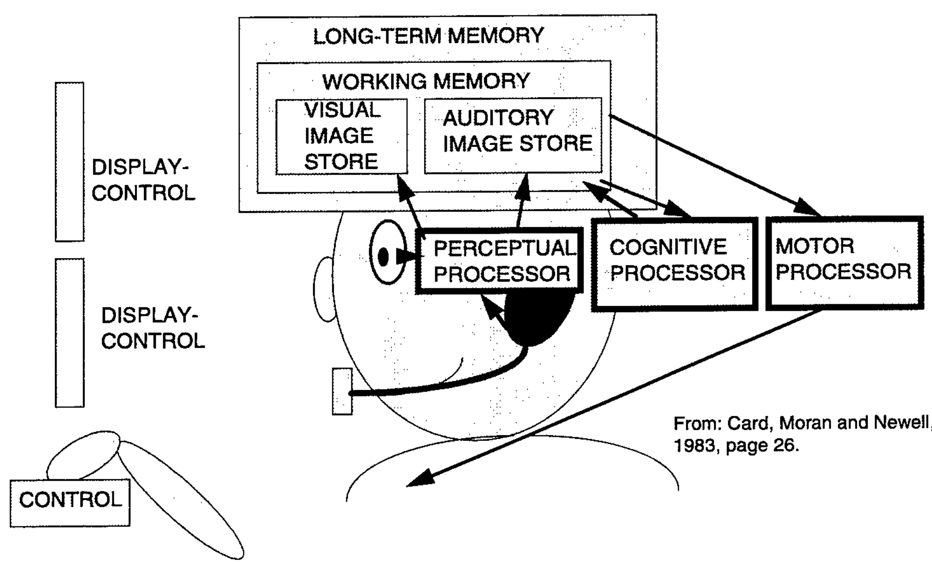


Figure 2. Model Human Processor.

CPM-GOMS analyses were performed for mouse, touch, and voice input modes (John, 1990; Kieras, 1993). The analyses were performed using Operation Sequence Diagrams (OSDs) showing parallel cognitive, perceptual, and motor processing activity. Two OSDs were used to analyze the mouse, one showing Homing and Find Pointer operations, and the other showing Pointing and Clicking operations.

The analyses were preliminary estimates based on the cautions presented in the literature and, therefore, must be checked empirically. Initial parameter estimates and assumptions, presented in Appendix E, were used only for model checkout and example output development.

3.3 MODEL DEVELOPMENT

Time-event task network software produced a hybrid model, combining the CPM-GOMS analysis into a computer simulation of the Strike Coordination tasks. This model was developed using MicroSaint DOS-based software.

The time-event task-network model times tasks determines branching between tasks, performs computation at the beginning and end of each task, and determines that conditions are suitable before a task is released (e.g., a task which requires the hands cannot begin if the hands are busy doing something else). Execution tasks have priority in this model leaving Window tasks to be performed as time permits between the three parts of the Execution sequence. Additionally, three types of interrupting tasks may occur (depending on the model setup) that require the hands, eyes, or ears. When these interrupting tasks occur, other Execution or Window tasks that require these resources (hands, eyes, ears) cannot begin.

Note that the modeling software did not permit instantaneous interruption, and the modeled user completed a HCI event (such as pointing and clicking) before turning to the interrupting task.

As figure 3 shows, the model contained two types of top-level tasks: (1) an HCI Event task using one of the modes whose time is determined by the HCI activity, and (2) Other Operator tasks (such as read, decide) modeled with a fixed estimated time.

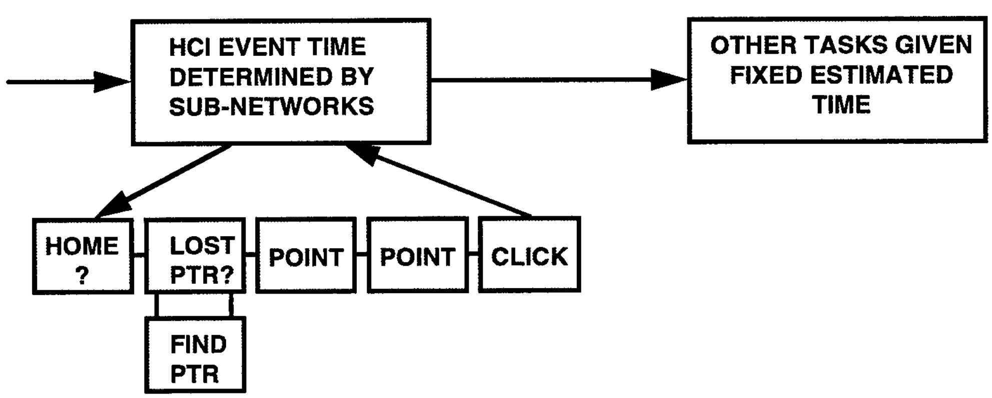


Figure 3. Two types of tasks in the model: (1) HCI Events and (2) Other Tasks.

Note that over time, differences would be because of HCI modal differences and not other human task variability. These HCI modal differences produced the need to develop OSAW.

HCI events were modeled for mouse, touch, voice and typing (Appendix F). The C, P, and M in the block diagrams in Appendix E stand for Cognitive, Perceptual, and Motor, respectively. The time for each event is specified (at this time, by KLM parameters), and branching is determined by other parameters (such as probability of a lost pointer and the probability of utterance recognition).

As computations in the tasks End Effect, the number and time of Cognitive, Perceptual, and Motor activities were accumulated. Consequently, there are two sets of parameters associated with each HCI event, one determined by KLM values and the other determined by the CPM-GOMS analysis.

3.3.1 Measures

The model for the analysis of the user interface produced three types of measures: (1) Productivity, (2) Workload, and (3) Link Analysis Measures. These measures are examples of output, and many additional variations are possible.

Productivity is measurable as the number of tasks completed (in a designated amount of time or per unit time). Only complete blocks of tasks were counted as completed (e.g., all Window tasks completed, or one of the three blocks of Execution tasks completed).

The Link Analysis measures produced a matrix of transitions (e.g., the number of HCI events for which the hand moved from the mouse to the keyboard or the number of times the hand stayed at the mouse for the event).

3.3.2 Model Versions

Three versions of MAUI were created for checkout and testing: (1) MSTRIKE, using mouse and keyboard; (2) TSTRIKE, using touch-screen and keyboard; and (3) VSTRIKE, using voice recognition, touch-screen, and keyboard. In VSTRIKE, some of the tasks, such as selecting items from a list, are completed through touch-screen because these would be awkward to implement with voice recognition.

3.4 EXAMPLE MODEL OUTPUT

Trial MAUI runs produced examples output; however, the model was not validated because there were many parameters that were arbitrary initial selections. The MAUI output should be viewed only as examples of the information that could be produced.

The independent variables in these runs were the amount and type of interrupting tasks. For each of the three models (MSTRIKE, TSTRIKE, and VSTRIKE), 10 runs (only one run for condition) were made:

- 0% interruption
- 10%, 20%, and 30% manual interruption
- 10%, 20%, and 30% auditory interruption
- 10%, 20%, and 30% visual interruption

Each run was for 1 hour of simulated time (3600 sec). Each interruption was 36 sec. For 10% interruption, 10 interruptions occurred; for 20% interruption, 20 interruptions occurred; and for 30% interruption, 30 interruptions occurred during the 1-hour trial. Note that there are random occurrences (e.g., number of repeats) in these trials, and that many trials would be required for statistical inferences.

Figures 4 and 5, and Appendix G provide example outputs. Note that the amount of workload did not include any variability because of non-HCI work since the time for non-HCI tasks was fixed in the model.

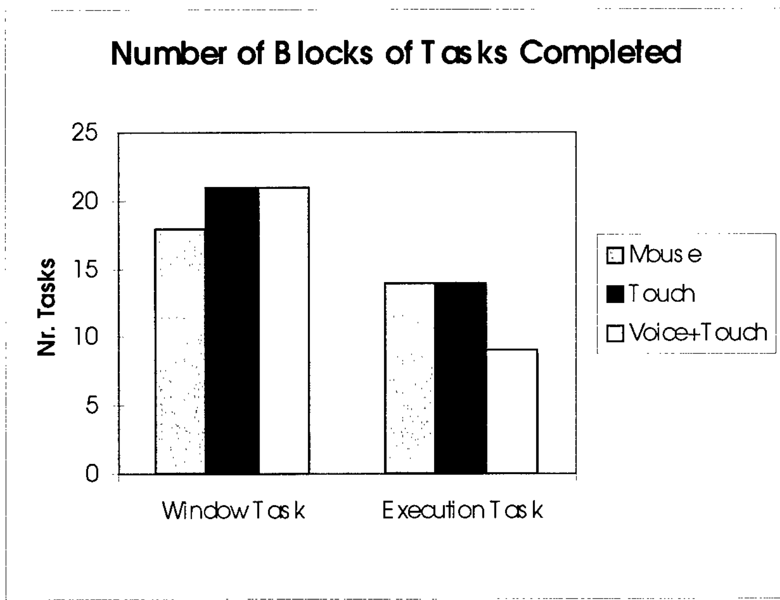


Figure 4. Example output: number of blocks of tasks completed.

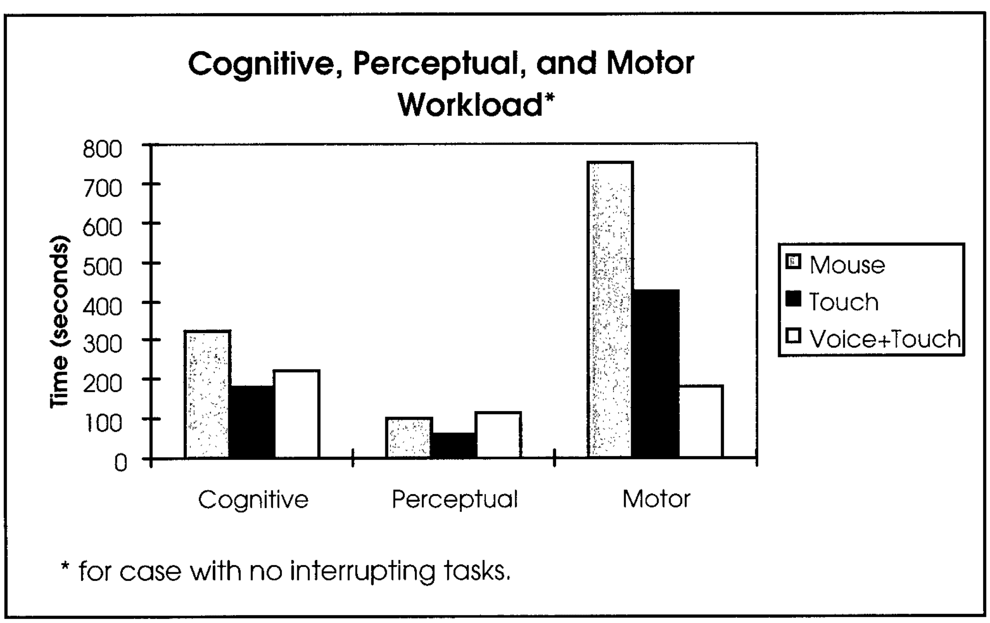


Figure 5. Example output: Cognitive, Perceptual, and Motor workload.

3.5 TASK ANALYSIS SUMMARY

In conclusion, creating a new model requires a significant amount of effort; however, when multiple design iterations are examined, the result is worth the effort. The current model requires the collection of empirical data and adjustment of model parameters. However, this analysis shows that multi-modal HCI has strong potential over conventional workstations. Such a model can optimize the design in minimum CPM workload, maximum productivity, and best-efficiency for a given scenario of tasks.

4. MULTIPLE-DISPLAY STUDIES

Multiple displays provide large workspaces for the multi-tasking environments. Integration of multiple FPDs into a single OSAW console uses smaller footprints, lighter weight, and less power consumption than CRTs.

One of the most serious shortcomings of current workstations is that they do not provide efficient access to the large amounts of information required for supervision and multi-tasking because each task is involved in multiple application settings (e.g., AN/UYQ-70 Consoles and AEGIS Combat Information Center, etc.). Unfortunately, many current workstations are not designed to view multiple screens of information in quick succession. While there are various potential solutions to this problem of information access, two classes of solutions are practical and feasible alternatives, given the current technology. The first solution is to provide more screen space by adding more displays, larger displays, or both. The second solution is to provide virtual workspaces (screens of information) or virtual desktops such that several workspaces can be brought successively into view on a single monitor with an advanced interface that allows rapid switching between virtual workspaces. Current interface switching systems—pull-down menus and task bars—have several limitations. Furthermore, task bars are generally available only for switching between applications, not switching between workspaces or desktops.

Others have proposed an alternative means of switching between workspaces, but alternatives have not been evaluated (Watts, 1994). A workspace control diagram, essentially an enhanced task bar, is one promising example that has appeared on some Unix-based operating systems and is also available as a commercial application for Windows.

Two experiments were performed to evaluate different workstation designs for various generalized user tasks (St. John, Manes, Oonk, and Ko, 1999). In Experiment 1, the evaluation included alert perception and display monitoring in a dual-task situation. Four workspaces were used and they were presented on one monitor with virtual workspaces, two monitors with virtual workspaces, and four monitors without virtual workspaces. The switching interface consisted of a hot-key-operated workspace control diagram with indicators for alerts. In Experiment 2, we evaluated and compared multiple display configurations, combining displays and virtual workspaces on a number of common human–computer interaction operations such as finding and accessing workspaces, transferring information, and monitoring.

4.1 EXPERIMENT 1

Experiment 1 evaluated whether multiple monitors with all workspaces visible would be superior to one or two monitors with a workspace control diagram.

4.1.1 Method

We placed participants in a dual-task environment in which the primary task was a tracking task and the secondary task was to monitor gauges for alerts. The tracking task involved keeping a "car cursor" centered on a moving road, while the monitoring task involved detecting alerts on up to three workspaces filled with gauges. Figure 6 shows the workspaces used for these tasks.

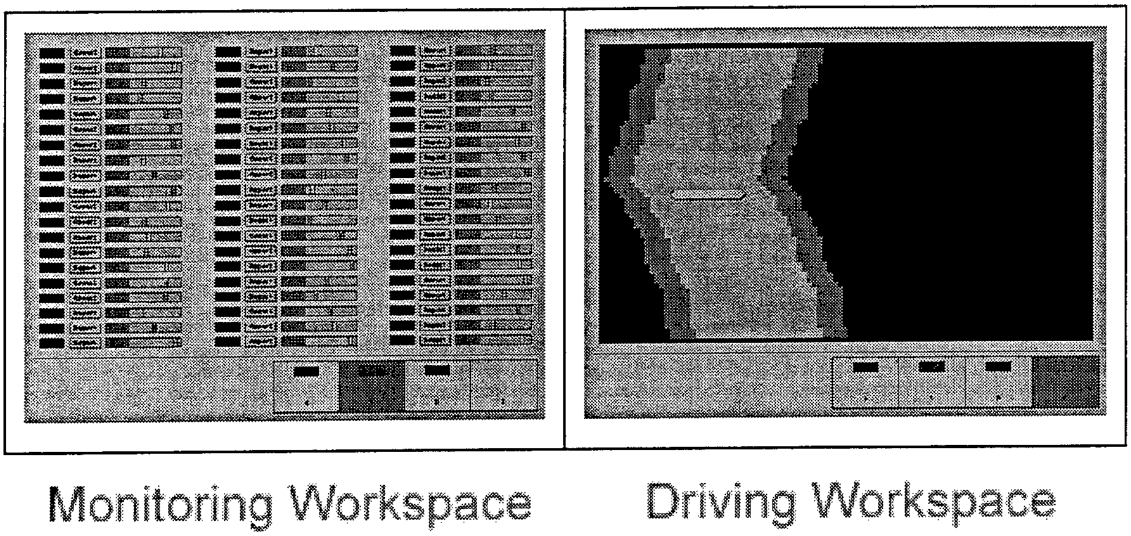


Figure 6. Screen capture of the Monitoring Workspace (three columns of 20 gauges) and the Driving Workspace in Experiment 1. Note the workspace control diagram in the lower right corner of each workspace.

Two types of visual alerts were presented on the Monitoring Workspaces—a *red alert* involving an indicator next to a gauge turning red and flashing, and a *needle alert* involving a needle moving into the "warning region" of a gauge. The red alert was considerably more salient than the needle alert and was detectable by peripheral vision, while the needle alert required direct viewing.

Experiment 1 involved seven display conditions, but this report discusses only the four conditions in figure 7. The independent variables were display condition (driving only, one or two monitors with a workspace control diagram, or four monitors with all workspaces visible) and alert type (red or needle). The dependent measures were tracking error (the root mean square distance between the center of the car cursor and the center of the road, in pixels), detection time (the time to detect an alert, in seconds), and report time (the time to report an alert following its detection, in seconds).

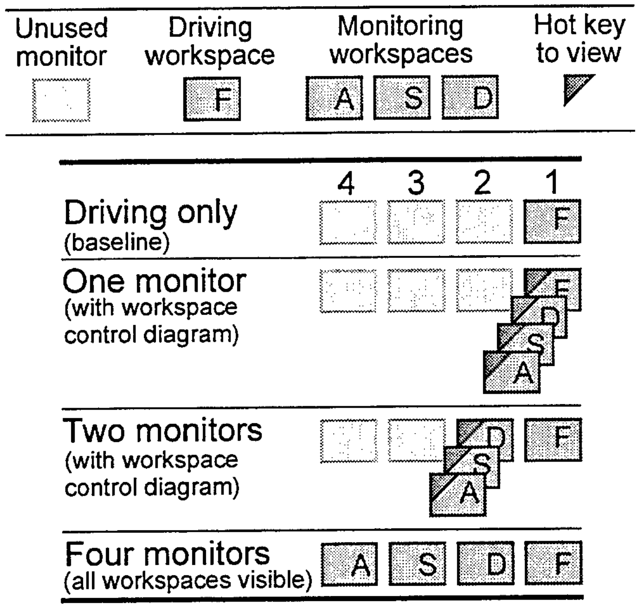


Figure 7. Four of the display conditions tested. The letters A, S, D, and F appeared on the workspace control diagrams, identifying which hot keys to press.

Figure 8 shows the layout of the experiment workstation. The Driving Workspace was always presented on Monitor 1, while the three monitoring workspaces were presented one at a time on Monitor 1 or Monitor 2, or all at once on Monitors 2, 3, and 4 (figure 7). A workspace control diagram was located on the bottom of Monitor 1 for the one-monitor condition, and Monitor 2 for the two-monitor condition. This diagram indicated which hot key to press (either the A, S, D, or F key on the keyboard) to bring a hidden workspace into view. Indicators on the diagram also showed when a red alert was occurring, adding further to the salience of the red alerts (for conditions with workspace control diagrams only).

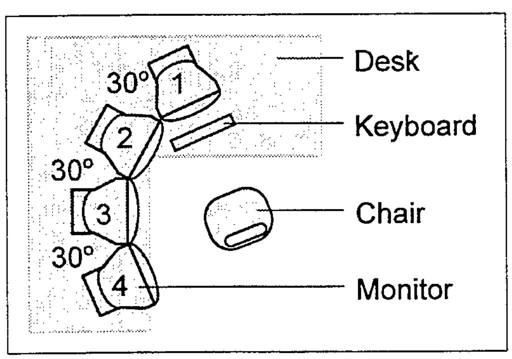


Figure 8. Experiment workstation layout.

Eighteen participants between the ages of 16 to 62 participated in Experiment 1. All participants performed in each of the seven display conditions. Six red alerts and six needle alerts were presented for each condition. Participants steered the car cursor with the left and right arrow keys while scanning the gauge workspaces (using hot keys, if applicable) for alerts. When an alert was detected, participants paused the driving task using the up or down arrow key, moved the cursor to the location of the alert, and clicked on a Report button located next to the alert.

4.1.2 Results

To understand the effect of display condition on performance, we performed a one-way, repeated measures Analyses of Variance (ANOVA), including the three dual-task display conditions, for each dependent measure. There was a main effect of display condition on tracking error, $F(2, 34) = 4.56$, $p = .0176$, and detection time, $F(2, 34) = 3.34$, $p = .0472$, but not on report time, $F(2, 34) = 1.15$, $p = .3279$.

Figure 9 shows that driving performance improved as the monitoring task was distributed over more screens. A Tukey–Kramer post-hoc test indicated that driving performance for the four-monitor condition was significantly better than for the one-monitor condition, $p < .05$. No significant differences were found between the two-monitor and four-monitor conditions or the two-monitor and one-monitor conditions. All dual-task conditions yielded substantially higher tracking errors than the driving-only (baseline) condition.

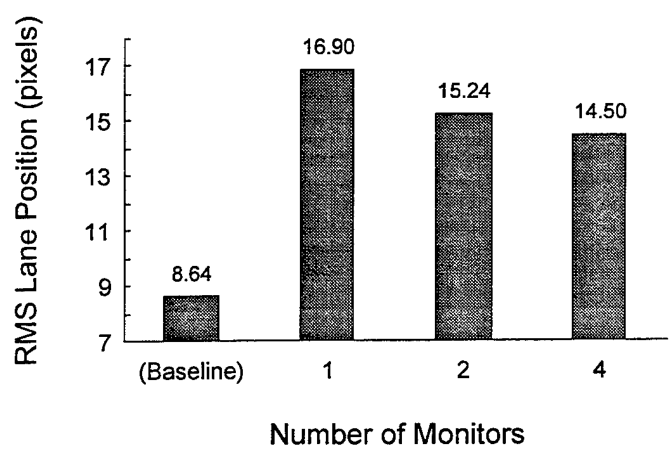


Figure 9. Driving performance with increasing screen area.

Figure 10 shows a slightly different story for alert detection—times were best for the two-monitor condition. A post-hoc analysis, however, revealed that only the improvement in alert detection from the one-monitor condition to the two-monitor condition was significant, $p < .05$. Figure 10 also shows, as expected, that the more salient red alerts were detected considerably faster than the needle alerts, $F = 53.48$, $p < .0001$.

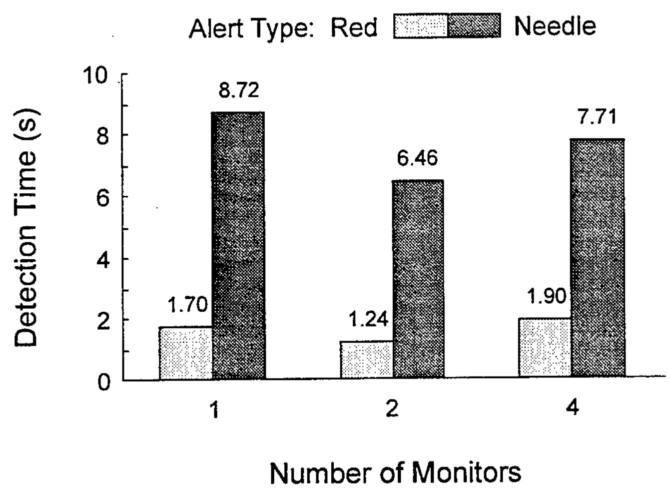


Figure 10. Time to detect an alert for four workspaces as a function of the number of monitors and the alert type.

We observed the worst driving and monitoring performance when the workstation was configured with one monitor and a workspace control diagram, probably because the driving workspace was hidden for long durations during each trial. The two-monitor and four-monitor conditions, however, differed little except that one supported slightly better driving performance and the other provided slightly shorter alert detection times.

4.1.3 Discussion

Experiment 1 found that task performance with only one monitor was degraded compared to performance with two monitors. However, the more interesting finding is that there was little difference between two monitors with a workspace control diagram and the four monitors with all workspaces visible in either driving performance or alert detection. If anything, having only two monitors and a switching interface allowed for better monitoring performance. This is probably because the workspace control diagram was enhanced (red alerts were indicated on the diagrams as well as next to the gauge) and the more peripheral monitors (Monitors 3 and 4) were not used. The minimal cost of implementing such a switching interface might be more cost-effective and space-efficient than purchasing multiple monitors.

4.2 EXPERIMENT 2

We evaluated and compared multiple display configurations combining displays and virtual workspaces on many common human–computer interaction operations such as finding and accessing workspaces, transferring information, and monitoring. In Experiment 1, we investigated accessing workspaces in a dual-task situation involving tracking and alert monitoring. Participants performed better at monitoring for alerts when they used a workspace control diagram to switch between four workspaces presented on two monitors than they did when using only eye and hand movements to access the same number of workspaces, each presented on its own dedicated monitor.

4.2.1 Method

We chose a factory inspection task in which participants monitored 12 assembly lines for mismatches (parts that were placed on the wrong line). Upon finding a mismatch, participants were required to transfer it to the correct assembly line. The 12 workspaces were presented on two, three, or four monitors (figure 11). Because there were always more workspaces than displays and only one workspace could be viewed on a given display at a time, participants needed to refer to a workspace control diagram and press hot keys to switch between the workspaces. A mismatch could belong to any of the following three mismatch types: (1) switch (the transfer required a workspace switch using the diagram or hot keys), (2) monitor (the transfer required a traversal from one monitor to another), or (3) both (the transfer required a workspace switch and a traversal between monitors).

Each participant performed the task using four display configurations. For each configuration, 12 workspaces were used, although the number of workspaces per monitor varied between configurations. The following four display configurations were tested:

1. 2H: two monitors arranged horizontally (six workspaces per monitor)

2. 2V: two monitors arranged vertically (six workspaces per monitor)

3. 3H: three monitors arranged horizontally (four workspaces per monitor)

4. 4HV: four monitors arranged in an upside-down "T" (three workspaces per monitor)

The workspace control diagram contained clusters of rectangular buttons. There was one cluster for each monitor in use, and the positions of the clusters corresponded to the positions of the monitors. Each button on the diagram contained a letter identifying the workspace and a number identifying the hot key to press to bring that workspace into view. Because having a different hot key for each of the 12 workspaces would have been cumbersome, each hot key actually brought a group of workspaces into view (one for each display in use). For example, in the 2H configuration, pressing the 2 key brought Workspaces B and H into view (figure 12), while in the 4HV configuration, pressing the same key brought Workspaces B, E, H, and K into view. Figure 11 shows which hot keys brought which workspaces into view.

4.2.2 Results

Analyses were conducted to determine if any significant differences were evident among the four display configurations (2H, 2V, 3H, and 4HV), two transfer methods (drag and drop and cut and paste), and three mismatch types (switch, monitor, and both). The transfer method was a between-participant factor while display configuration and mismatch type were within-participant factors. Analyses included the dependent measures described in table 2.

Table 2. Four dependent measures used in data analyses.

Measure*	Description
Task time	Average time to complete a trial once the mismatch was presented
Detection time	Average time to notice the mismatch once it was presented
Locate time	Average time to bring the mismatch into view and pause the assembly lines
Transfer time	Average time to transfer the mismatch to the correct inbox after the assembly lines were paused

* All units in seconds.

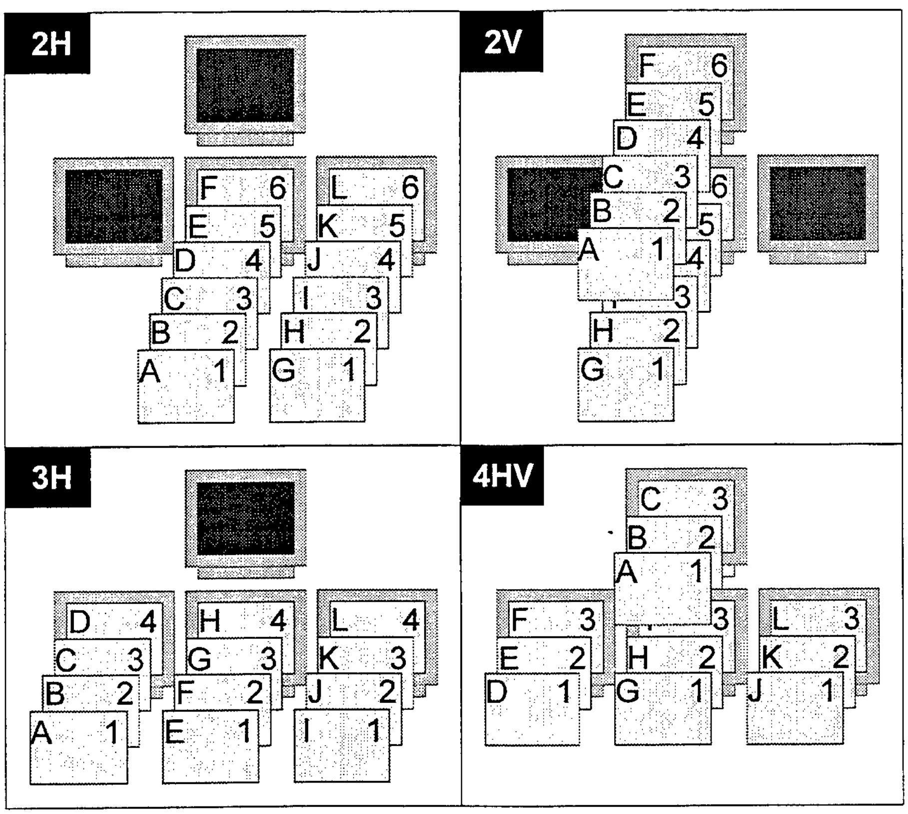


Figure 11. Four display configurations tested in Experiment 2. The same 12 workspaces (A through L) were used for each configuration. The number on each workspace indicates the hot key required to bring it into view.

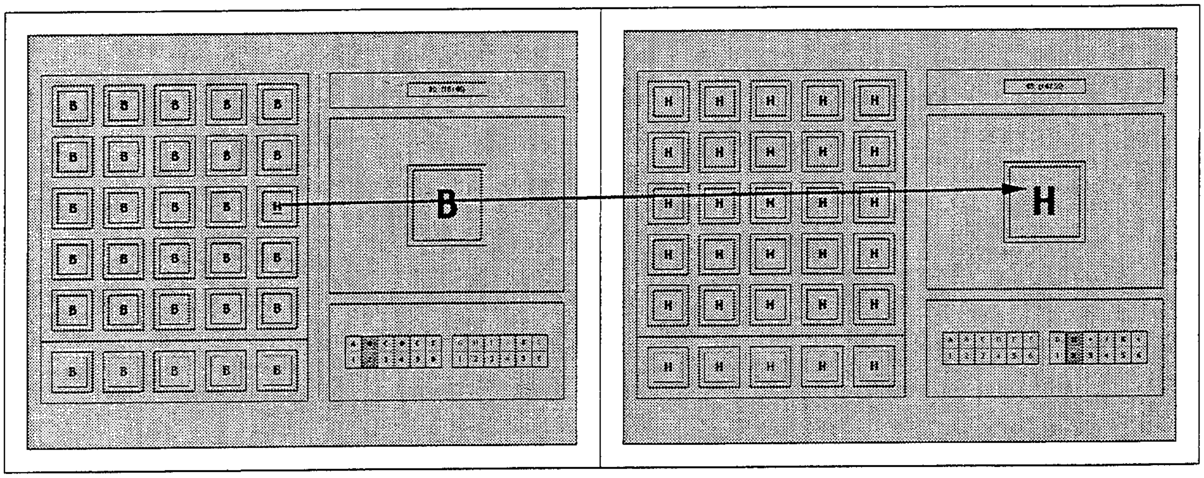


Figure 12. Screen capture from the 2H configuration. In this example, the assembly line on Workspace B (located on the left monitor) contains a mismatch in Row 3, Column 5. The participant must transfer the mismatch to the inbox on Workspace H (located in the right monitor).

Note that task time is a composite measure of locate time and the transfer time. Furthermore, detection time and locate time were often the same because participants frequently pressed the space bar while the mismatch was in view (meaning they detected and located it simultaneously). Finally, no ANOVAs were conducted for transfer errors (transferring a mismatch to the wrong inbox) because such errors were extremely rare. Participants committed only 21 transfer errors in 5280 trials (less than 0.40 percent). Figure 13 shows the mean task times as function of transfer method and display configuration. These times were analyzed with a 2 (transfer method) by 4 (display configuration) ANOVA. There was a main effect of transfer method, $F(1, 22) = 36.51$, $p < .0001$. Participants using the drag and drop method to transfer the mismatch performed the task 3.20 seconds faster than those using the cut and paste method. There was also a main effect of display configuration, $F(3, 66) = 7.56$, $p = .0002$. Separate Tukey–Kramer post-hoc analyses revealed that task times for the 4HV configuration were slower than each of the other three display configurations. There was no interaction between transfer method and display configuration, $F(3, 66) < 1$. The mean detection times were analyzed with a 2 (transfer method) by 4 (display configuration) ANOVA. A main effect of transfer method was found, $F(1, 22) = 5.26$, $p = .0318$.

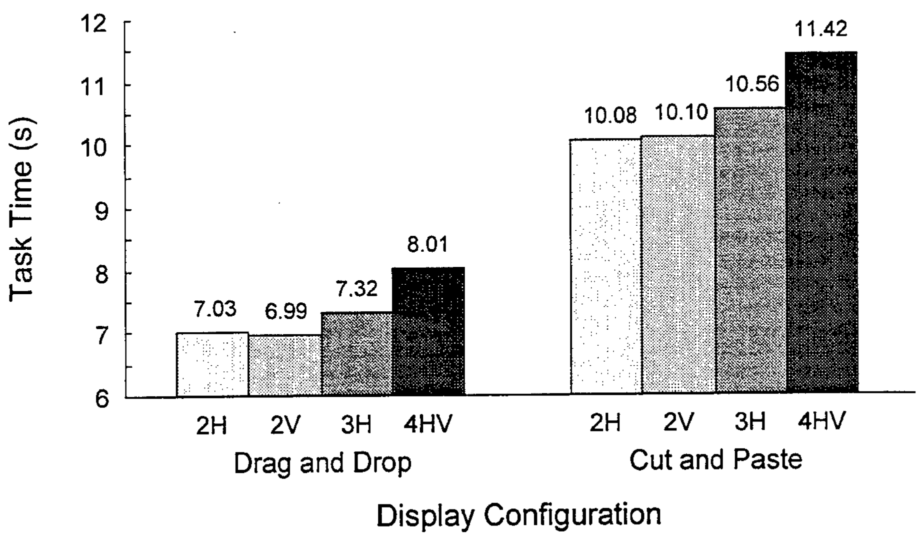


Figure 13. Effect of transfer method and display configuration on task time.

Participants in the drag-and-drop condition detected mismatches 0.58 seconds faster than those in the cut and paste method. There were no other significant effects, *F*s < 2.46, *p*s > .069. A similar ANOVA using locate time revealed no main effects or interactions, *F*s < 4.01, *p*s > .058.

Figure 14 shows the transfer times as a function of mismatch type for drag and drop and cut and paste. The analysis revealed a main effect of mismatch type, $F(2, 44) = 23.05$, $p < .0001$. Transfer times were fastest when the mismatch was transferred between two monitors and no workspace switching was required, and were significantly slower when just switching was needed to bring the correct workspace into view. Transfer times were slowest when a monitor change and switching were needed to transfer the mismatch. There was also an interaction between mismatch type and transfer method, $F(2, 44) = 5.17$, $p = .0096$, indicating that the difference in transfer times between the switch and monitor mismatch types was only found for the cut-and-paste condition.

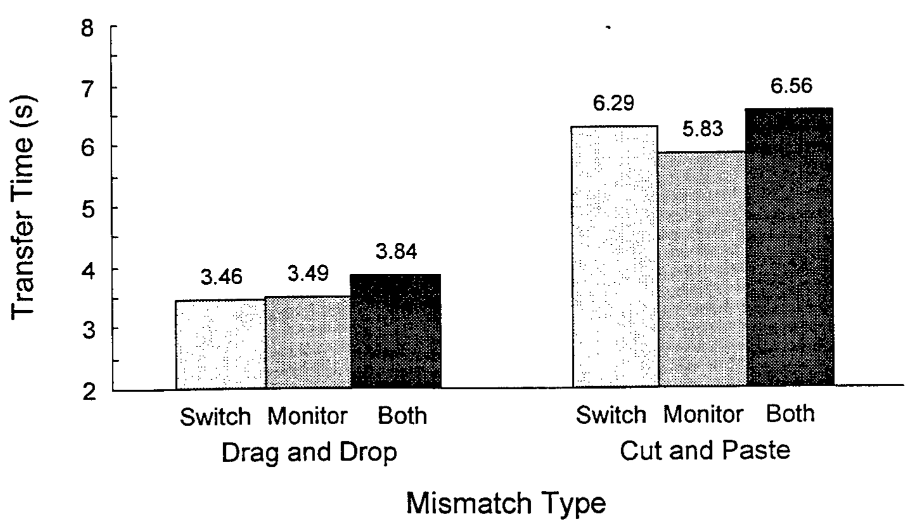


Figure 14. Effect of transfer method and mismatch type on transfer time.

4.2.3 Discussion

Experiment 2 results indicate that as the number of monitors increased, participants took increasingly more time to complete the factory inspection task. Consequent analyses revealed that most slowing was found in the transferring portion of the task (where participants transferred the mismatch to the inbox), not the monitoring portion (where the participant cycled through the assembly lines looking for the mismatch). There are three potential explanations for why the transfer time differed significantly among display configurations. First, in general, as the number of choices increase, the choice reaction time (the time to make a decision between those choices) increases. When transferring the mismatch, participants must determine the destination monitor either from the workspace control diagram or recall it from memory. Hence, as the number of monitors increased, the time to decide between them likely increased. A second explanation for the increase in transfer time with the number of monitors was that participants were more likely to lose track of the cursor as the number of monitors increased. Evidence for this explanation comes from participants' comments and experimenter observations. A third explanation is that mouse movements were necessarily longer on average with more monitors, and mouse movements in the 4HV configuration often involved a horizontal and vertical component (e.g., moving the cursor from the top monitor to the left monitor). In pilot trials, we found that peripheral vision was not sufficient for discovering mismatches. Participants must deliberately focus on each display, slowing the scanning process, and thereby decreasing the potential effectiveness of this strategy.

4.3 MULTIPLE-DISPLAY STUDIES SUMMARY

There are many methods to increase the efficiency of access to multiple workspaces. Each method has its own characteristic advantages and disadvantages. One method is to increase the screen space by increasing the number of monitors or replacing existing monitors with larger ones. Alternatively, increasing the resolution of existing monitors could enhance access to information, although at the expense of font and image sizes. The use of multiple real monitors is becoming more prevalent in the office and some military settings. However, multiple monitors are expensive and require a large physical workspace that is often unavailable, especially in military settings. Furthermore, there is a decreasing payoff in effectiveness for adding more monitors as their placement becomes increasingly peripheral to the user. A less-expensive solution to the information access problem is to use fewer monitors, but add an effective switching interface such as a workspace control diagram to create a large virtual screen area.

Experiment 1 showed that having only one monitor degraded performance compared with two monitors. However, the more interesting finding might be that there was little difference between the two-monitor condition and the four-monitor condition in either driving performance or alert detection. If anything, having only two monitors plus virtual workspaces with a switching interface allowed for better monitoring performance because of the enhanced workspace control diagram (red light alerts were indicated on the diagrams and next to the gauge) and the fact that the more peripheral monitors were not used.

The Experiment 2 results indicate that fewer monitors support better performance for tasks that involve frequent information transfer and monitoring. These findings support the use of the 2H or 2V workstation configurations rather than either 3H or 4HV.

In conclusion, multiple display studies suggest that two monitors with virtual workspaces enhanced by a hot-key-operated workspace control diagram gives optimal performance across various common multi-tasking environments.

5. MULTI-MODALITIES: TOUCH, SPEECH, AND 3-D AUDIO

There are many complementary capabilities to the visual display that can enhance a workstation. This section reviews three of these capabilities: touch screens, speech recognition, and 3-D audio localization.

5.1 TOUCH SCREEN

Other than voice recognition, touch input is probably the most natural human interface to any computing device. It is particularly useful and popular in those applications where the user is relatively unskilled in the operation of computer input devices. Touch screens have been used for many years, mainly in applications such as point of sale, public information kiosks, industrial and process control, military displays, medical displays, and interactive video systems.

We do not recommend touch screen in general Windows tasking. Users make some errors in touch screen interaction. We recommend that the software processing the touch inputs provide feedback to users. In addition, the touch screen interface should be developed in a user intuitive mode such as Variable Action Buttons. The designer should consider the size and location of a touch screen to reduce user fatigue.

The five most common touch screen technologies include capacitive, infrared, resistive, Near Field Imaging (NFI), and SAW. Each technology offers its own unique advantages and disadvantages. A Surface Acoustic Wave (SAW) touch screen was integrated with the FPD in OSAW. In addition to the X and Y coordinates, SAW technology can also provide Z-axis (depth) information. The harder the user presses against the screen, the more energy the finger will absorb, and the greater will be the dip in signal strength. A controller measures the signal strength of the Z-axis. We wanted to know how many levels of pressure sensitivities a user can detect and, if possible, to develop a new touch interface using the Z-axis information. For example, hard touch may replace the double-touch. Today, no software applications are designed to use this feature.

5.1.1 Document Review

We reviewed human factors and HCI literature,* including such topics as touch- screen performance and the interface and operator parameters that influence operator touch performance.

The following design principles and data were taken from the literature.

1. Users prefer direct pointing aspects of the touch screen except for text input, and they tend to want some form of feedback (from software processing) as a form of error reduction.
2. Selecting functions might be faster and more accurate with touch screen than keyboard/mouse technologies.
3. Users report arm and wrist fatigue after extended touch-screen use; thus, screen inclination angle other than vertical should be considered.

---

* Carlow International Incorporated. 1997. Touch Screen Interface Parameters and User Performance. Delivery Order 0002. Space and Naval Warfare (SPAWAR) Systems Center, San Diego, SSC San Diego.

4. Touch-screen response speed is equal to or better than other input devices.
5. Depending upon the task (see design principle 1 above), displayed material, pointing resolution, and user experience, touch screen response accuracy can be less than data tablets, keyboards, mice, joysticks, and track balls.
6. Users tend to learn touch-screen use easily.
7. Touch-screen device performance is comparable to other input devices in all but very-high-resolution tasks.

The following items summarize operator performance observations and effects of interface parameters applicable to SAW devices and recommendations ways to implement the touch-screen interface.

1. Handedness is not an issue.
2. Operators might be able to differentiate between two well-separated levels of pressure or Z-axis pressure levels
3. Consider application of the take-off algorithm (Potter, 1988, 1989) for scoring a touch on a target. A cursor improves performance; an ability to control the cursor characteristics is important.
4. Highlight an object with which the cursor or touch spot is currently in contact to provide an effective cue.
5. Highlight the object currently being touched as the operator drags his/her finger over the object before a take-off response to indicate a selection.
6. A cursor improves performance; however, providing the user a cursor control ability is important. Assuming that some version of the recommended touch mouse interaction modes will be implemented, the visual feedback available should considerably reduce the touch error produced by users. Additionally, Beringer and Peterson (1985) showed that training and practice could substantially reduce the bias error.
7. A stylus might improve touch-response accuracy.

5.1.2 Hard–Soft Pressure Experiment

As mentioned in the Introduction, we wanted to know how many levels of touch sensitivities a user can activate. We found that operators cannot activate more than two levels. In an exploratory experiment in 1999, the SSC San Diego Touch Experiment was conducted to evaluate the use of an operator's touch to dual-activate a given on-screen button. Tactile input devices have been in use for some time. One method used to achieve on-screen activation is through sensing the pressure of an operator's touch. The question arose as to whether the differential of an operator's finger pressures could be used to activate the same on-screen button for two functions. This would be accomplished by having the operator press a button either softly or hard for two respective functions. Individuals have a personal perception as to what constitutes a soft or hard touch. How good is that perception? Can people be trained to decrease individual's various perceptions of hard and soft? These are the questions addressed in this exploratory experiment.

5.1.2.1 Approach. The experiment consisted of individuals responding to on-screen buttons that were labeled as either "Hard" or "Soft." There were no consequences as to the appropriateness of their responses. The computer recorded the pressure of each button activation.

5.1.2.2 Apparatus. An ELO SAW touch screen was used. It provided a 20.1-inch diagonal surface with a 1280 X 1024 resolution and a sensitivity of 8 bits (0 to 255). On the screen were 20 buttons, half labeled "Soft" and half labeled "Hard." The buttons were randomly arranged across the screen area. Each button was activated using one's finger.

5.1.2.3 Participants. Colleagues who are members of SSC San Diego Code D44210 participated as subjects for this experiment. There were 10 participants, 7 males and 3 females with an estimated age range from 21 to 45 years.

5.1.2.5 Procedures. There were two sets of trials in the experiment. The first set were referred to as "Natural" trials since there was no training as to what was considered a soft or hard touch, although they were allowed several familiarization trials. The second set, "Training" trials, started with two trials where the participants were given feedback as to their correct or incorrect pressure activation of a given button.

5.1.2.6 Analysis. The data were formatted using the Microsoft Excel® Program and then read into the SPSS Statistical Software for analysis. The analysis for Paired Samples Statistic was used to calculate the means, standard deviations, and T-Tests. During each trial, a given participant produced 20 data points, 10 for Soft button pushes and 10 for Hard button pushes.

The SPSS Program first averaged the scores within each condition over the 10 participants before calculating the means, standard deviations, and standard error means (table 3).

Table 3. Hard–Soft touch with and without training.

Comparisons	Means	N	Standard Deviation	Standard Mean Error
Pair 1 NH	223.35	10	57.52	18.19
TH	227.88	10	27.43	08.67
Pair 2 NS	79.16	10	48.15	15.23
TS	67.21	10	46.97	14.85
Pair 3 NS	223.35	10	57.52	18.19
TS	79.16	10	48.15	15.23
Pair 4 NS	227.88	10	27.43	08.67
TS	67.21	10	46.97	14.85

Table 4 shows the T-Tests. The analysis shows that there were no significant differences because of training either in activating Hard or Soft buttons. As would be expected, however, there were significant differences (0.001 = 4.781) between the participants' ability to apply the correct pressure for the Hard and Soft buttons respectively, regardless of training.

Table 4. T-Test results.

	Paired Differences					t	df	Sig. (2-tailed)
			Standard Error Mean	95% Confidence Interval of the Difference				
				Lower	Upper			
Pair 1 TH –NH	4.53	63.12	19.96	40.63	49.68	227	9	.826
Pair 2 NS-TS	11.95	39.39	12.46	40.13	16.23	.960	9	.362
Pair 3 NH-NS	114.19	64.38	20.36	98.13	190.25	7.082	9	.000
Pair 4 NH-NS	160..67	49.54	15.67	25.23	196.11	10.256	9	.000

TH = Training/Hard Touch TS = Training/Soft NH = Natural/Hard NS = Natural/Soft

Figures 15 shows individual performances of the participants for the NH condition. The individual performances are more representative of what can be expected than indicated by the above statistics. It should be noted that only one individual did not exceed the cut-off criteria of 128 for a "Natural Hard" condition. This was true for the other conditions except for the "Training Hard" condition.

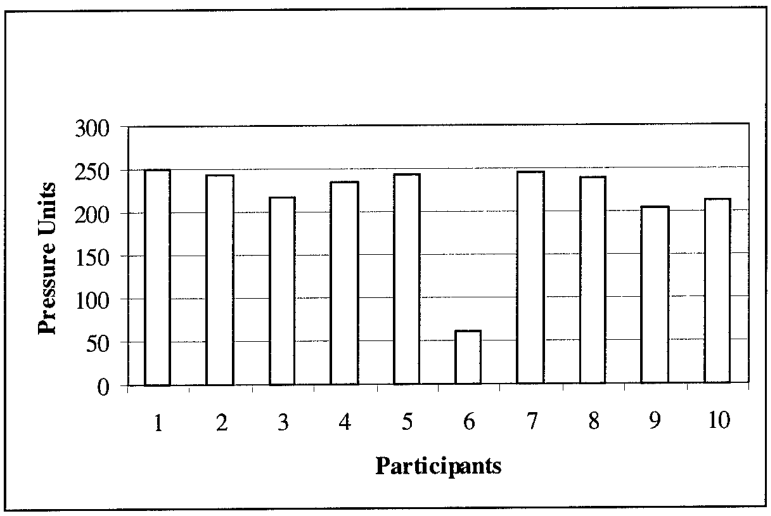


Figure 15. Average "Hard" pressure exerted by individual participants with no training. (Each bar is an average of 100 key activations.)

5.1.2.7 Conclusion. We found that operators cannot activate more than two levels. Training did not result in a significant improvement in performance. Perhaps further training and practice would result in better performance. However, the results indicate that training might not be necessary if some error is initially acceptable. The means between Soft and Hard button pushes were statistically significant without any training.

5.1.3 Touch Screen Summary

We do not recommend touch screen in general Windows tasking. The touch-screen interface should be developed in a user intuitive mode such as Variable Action Button. The designer should consider the size and location of the touch screen to reduce user fatigue.

5.2 SPEECH TECHNOLOGIES

The speech field encompasses topic areas that range from baseline feature extraction of the speech signal through Digital Signal Processing (DSP) to speaker and language identification, speech recognition and synthesis, and natural language discourse systems. In general, business software (word processing, spreadsheets, databases, etc.) is more mature and performs better than speech technologies. Technologies that support interaction between a human and a computer through speech have great promise, but they are still too unreliable and immature for wide use commercially. There are still too many unanswered questions about what makes an effective speech interface, and about the metaphors and paradigms. In other words, there is not yet an accepted concept of operations for how a user speaks to a computer interface, be it the desktop, an application, or an agent. In the military computing environment, even less is known about how to build software with speech technologies.

Commercially, only two speech technologies have been successfully deployed to any significant extent in "off-the-shelf" software. Automatic Speech Recognition (ASR) and synthetic speech generation or Text-To-Speech (TTS) are being marketed in dictation systems. These technologies need more research and development—speech recognition is not 100% accurate, and TTS still sounds mechanical and unnatural.

Natural language and discourse technologies remain in the research realm. ASR and TTS play important roles in these technologies. In a multi-component discourse system, they are the most mature components. Other components include semantic parsing (meaning extraction), context tracking, language modeling and generation, and dialog management. These components rely on hand-tailored systems by teams of linguists and language modelers, and most are still proprietary.

DSP is used in many speech technologies to extract fundamental mathematical features of the speech signal. These features are then used in applications such as speaker and language identification, stress detection, and word spotting. DSP generally involves computing Fourier transforms on the speech signal, and then determining the Cepstral coefficients.

5.2.1 Automatic Speech Recognition

There are several implementation levels of ASR. The simplest and, possibly, the most useful, is "See/Say" functionality.

5.2.1.1 See/Say Function. The user may activate a button or menu that is represented by the user interface object on the display (e.g., the "OK" button, "File" menu). See/Say and macros are relatively easy to implement and can lead to dramatic performance improvements in HCI navigation and control.

5.2.1.2 Speech Macro. The next level of speech functionality is the speech macro. A speech macro enables a user to collect a sequence of linear primitive operations and later start the operation sequence by saying the macro name. This is another relatively simple speech implementation that provides performance enhancement and adds positively to the user's experience.

5.2.1.3 Grammar-Based Speech Recognition. Grammar-based speech recognition refers to finite state grammars that filter specific words in specific orders. Out-of-vocabulary (OOV) words, and vocabulary words that are said out of grammatical order are filtered out by the grammar and are not recognized by the ASR engine. The use of grammars makes it easier for an ASR engine to recognize individual words and word patterns by limiting the range and scope of the potential recognition domain.

5.2.1.4 Natural Language. Natural language implementations range from straightforward semantic parsing of ASR output to wide-ranging, free-form conversation between the human and the natural language system. Most natural language systems remain in the research and development realm, with the notable exception of MagicTalk™ by General Magic, Inc.

The bulk of the DARPA-sponsored natural language research focuses on the commercial domain. There are systems that provide such virtual assistant services as booking airline, hotel, and car reservations. The goal of these programs is to replace the humans currently providing those services with a discourse system that has a speech front-end and a service-related database back-end. The currently available systems are primarily located within university-based research institutions.

5.2.1.5 Applications. The leading COTS speech products provide Windows desktop navigation and application command and control. Retrofitting speech recognition as an input modality to existing COTS applications that were originally designed to support mouse and keyboard input modalities is a problem. Speech is particularly ill-suited to such navigation tasks as menu selection, cursor placement, and window control. It is only partially successful as a discrete command alternative mode as in file opening and saving. Similar conclusions apply to the use of speech to navigate the desktop.

Speech recognition technologies seem to be most successful in application command and control. The leading products support a measure of interoperability between dictation tasks, application commands, and desktop navigation. Thus, one can switch to another application or issue a command without pausing during dictation. This functionality is based on keyword recognition in which specific control words are used as command keys to the speech engine. In well-designed systems, the user can switch seamlessly between the desktop, the application, and text dictation.

Appendix H compares features of the commercial speech recognition systems. All commercial dictation systems are based on large vocabulary and trigram grammars. The vocabulary and the language models are based on either the *Wall Street Journal* model or a proprietary model. The various models use a statistical technique to determine word order and word sequence likelihood. Thus, a language model based on the prose style of a leading newspaper will not perform well in a specialized technical domain such as a military command post. Further complicating the issue of COTS speech recognition adequacy in the military is that recognition performance typically degrades in noisy environments and during use by inexperienced users.

The leading COTS office dictation products are as follows:

- Recognition engines include IBM Via Voice, Dragon Naturally Speaking, and Lernout and Hauspie Recognizer.
- Text-to-speech engines include Microsoft SAPI, IBM Virtual Voice, and Lernout and Hauspie Text-To-Speech.

5.2.2 ASR Application in OSAW

Figure 16 shows one combined application (i.e., the Lightweight Extensible Information Framework (LEIF) and Military Language Processor (MLP) in OSAW). This is an example of software integration featuring the tactical application of LEIF and COTS speech technology in a command center environment. The speech technologies represented here are speaker identification, speech recognition, and text-to-speech. The enabling middleware consist of a Java Speech Package. A LEIF Producer enables speech recognition and synthetic speech generation.

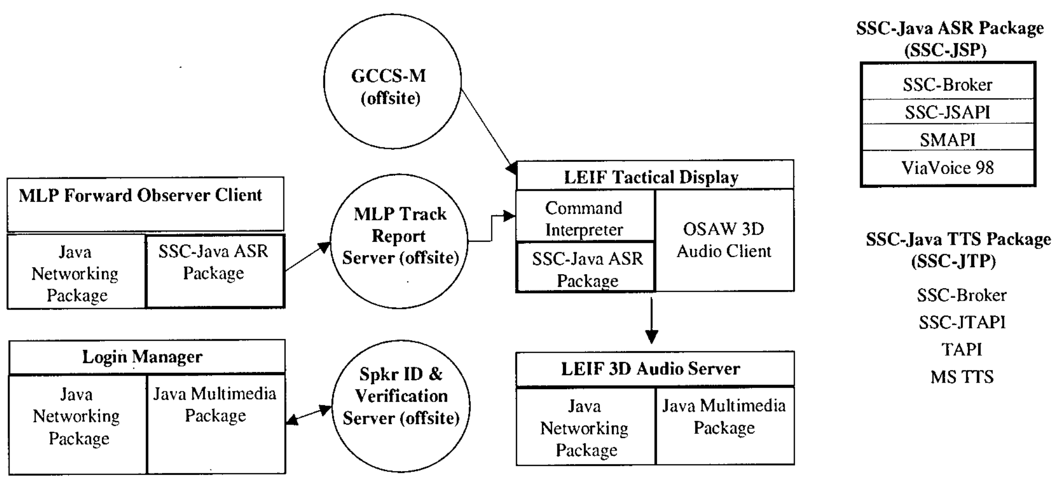


Figure 16. Software integration in OSAW.

The MLP is a semantic parser that extracts information from a naval standard message. Information about tracks, track kinematics, track history, forward observer data, etc., is extracted from the message (which may be a dictated contact report), and is passed to the tactical application for processing. LEIF receives the data and responds accordingly; for instance, by drawing the track on the tactical map. Figure 17 shows how MLP functions.

Speech recognition could be greatly improved, but it is available today. OSAW will provide the means, through research, to provide speech recognition design guidelines for future shipboard workstations.

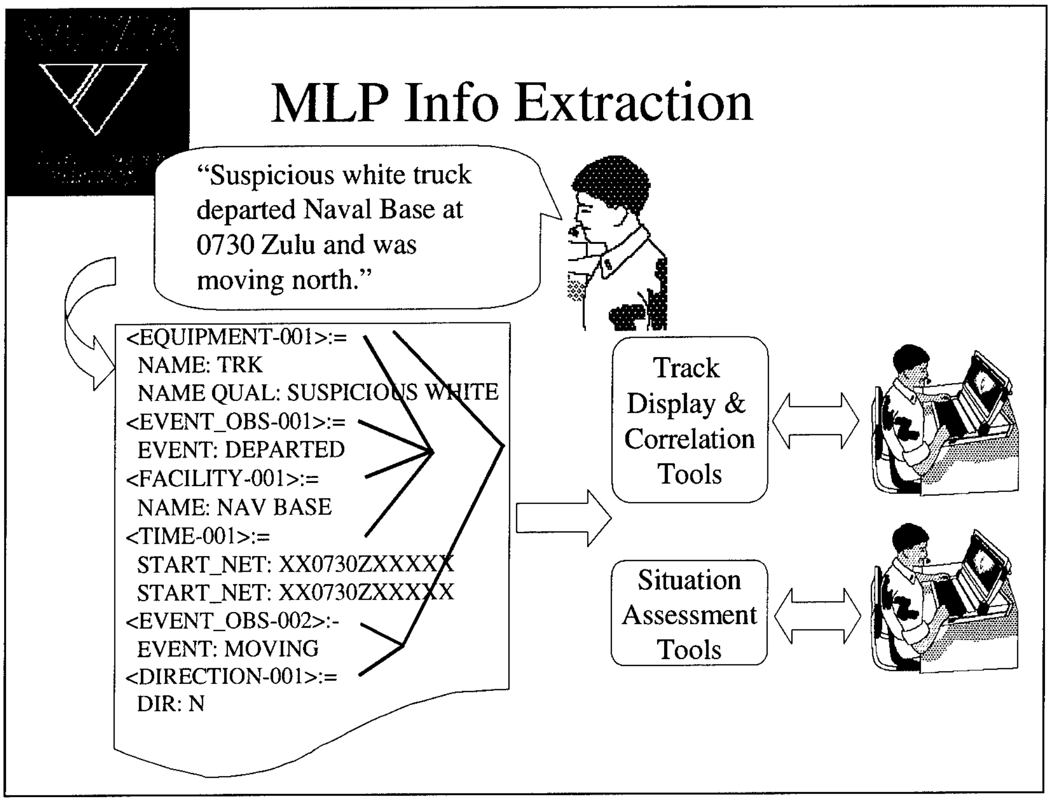


Figure 17. How MLP extracts information.

5.2.3 Speech Technologies Summary

The current state of speech technology is an odd mixture of research projects, COTS dictation products, deployed single-purpose telephony systems, and notional natural language systems. Speech is an immature technological solution looking for a problem. Work in the area has been driven not by a systematic analysis of the requirements, but rather on the idea that if people can speak to each other, they should be able to speak to their machines.

Taking an ad-hoc, non-process-oriented approach to the design and development of an entire technology leads to the same result as when dealing with a system or an application. One ends up with a collection of stand-alone things, some that work reasonably well (dictation, TTS), some that show promise (speaker ID), and others that need more work (NL).

For years, the holy grail of the speech development community was speaker-independent, continuous recognition. Large resources were used to solve the problem, and remarkably effective DSP techniques were optimized. Once optimization was accomplished, the belated question of "what is this good for?" was addressed. We now have dictation products that do a remarkable job of translating human speech to text, which is most appropriately used by individuals with physical handicaps. Neither the average typist nor the computer power user considers dictation an effective way to interact with the Windows desktop or an application.

Essentially, the individual technologies were not originally designed to complement each other or work well together. This is easily seen in the various ways that developers have tried to retrofit ASR to the desktop metaphor and business applications. The desktop metaphor does not work well with speech because speech cannot compete with the efficiency and convenience of the keyboard and the mouse. Similarly, speech is particularly ineffective in executing atomic application features that are better accessed through key shortcuts.

Rather than retrofitting speech recognition to existing keyboard/mouse user interfaces, we must rethink how to design computer interfaces so that speech is one of several equally effective input and output modalities. The keyboard and mouse reign supreme in the desktop metaphor, which in itself does a very poor job of providing an intuitive interface. Thus, rethinking and redesigning the HCI from scratch would be a very productive effort. The designers could learn from the past, apply a modern software engineering process, and design without preventing accommodation of future, unanticipated advances in computing capabilities.

Presently, it is important to remember what the "ins" and "outs" will be at the outset of a software development effort. Designing from the outset while considering speech and other input/output (I/O) modalities is critical to the ultimate success of all future projects.

5.3 SPATIALIZED 3-D AUDIO

Headphone listening is universal throughout the U.S. Navy, with pilots, traffic-controllers, flight-deck personnel, fire-control teams, weapons-console operators, sonar operators, etc., who are required to monitor multiple aural channels while simultaneously sending and receiving voice communications and responding to system-generated auditory alarms and instructions, often in the presence of interfering ambient noise. However, current headphone technology is clearly deficient in the information-processing requirements of these tasks. The effective spatial bandwidth of current U.S. Navy headphone technology is limited to a region between the listener's ears. Consequently, current headphone displays have only two or three auditory channels, far below the typical number of auditory information sources monitored in tactical situations. This problem is dealt with by either selective filtering through a switchboard device, or simply adding multiple headphone sets and/or speaker systems and letting the listener deal with the resulting cacophony. In modern fleet systems, headphone-based displays have become significant information-processing bottlenecks that severely constrain system performance.

5.3.1 Advancing the Technology

Headphone delivered synthetic 3-D audio is an enabling technology for meeting reduced manning requirements while simultaneously maintaining or improving system performance. Current headphone displays are limited because current headphone technology does not deliver the full range of auditory spatial cues required by human listeners to effectively parse a sound field created by multiple simultaneous auditory events. The advantage offered by new 3-D audio synthesis technology is that it provides headphone listeners with auditory spatial cues comparable to those heard under natural listening conditions. In effect, this new technology creates multiple virtual sound sources mimicking physical speaker devices while still taking advantage of the ambient noise-masking effects of headphones. This new technology will significantly improve the ability of headphone listeners to process multiple auditory events, including directional system alerts, in ways that are not possible with current stereo headphones. Using 3-D audio synthesis technology, simultaneous auditory events—as many as seven or eight—can be made more discernable by spatially filtering them so they appear to emanate from different locations. Figure 18 shows a 3-D audio synthesis block diagram. In addition to providing improved discrimination, synthesized 3-D auditory spatial cues can also direct visual attention horizontally and vertically. (Note that the lateralization capability of stereo headphones can only take advantage of interaural differences and therefore cannot provide directional cues for elevation or front-back position.)

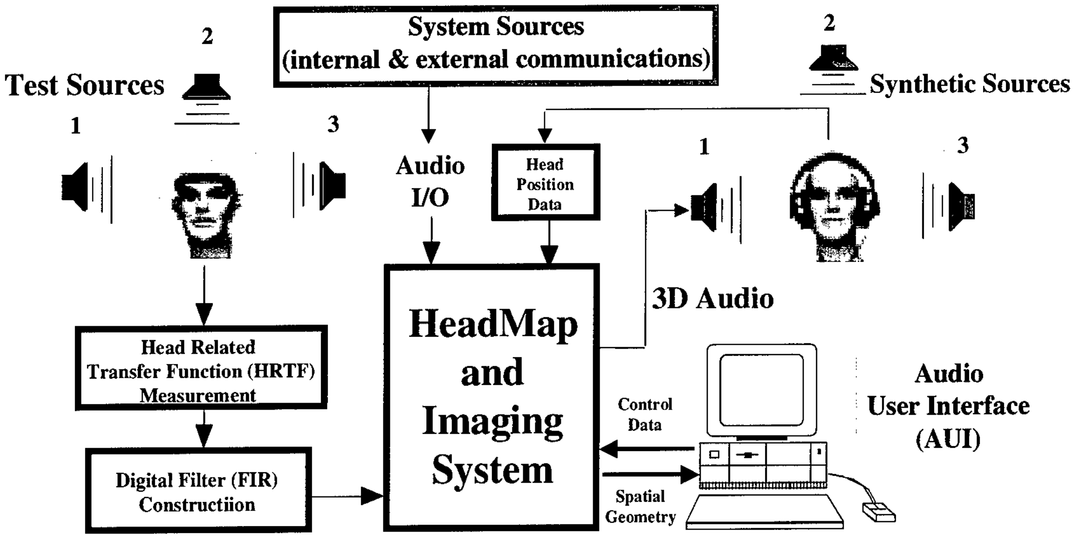


Figure 18. 3-D audio synthesis block diagram.

5.3.2 3-D Audio Localization

Ordinary stereo headphones provide directional cues by manipulating interaural differences (i.e., different arrival times and/or intensities of sounds at each ear). Other cues that are normally provided by the spectral filtering characteristics of the outer ears (pinna) are eliminated. These cues provide information about front/back and up/down positions. The locus of perceived locations of headphone delivered sounds is, therefore, limited to a line between the ears. In contrast, spatialized audio is sound processed to include as much directional information as possible, including synthesized pinna cues. When spatialized audio is delivered over headphones, the listener hears the sound as if it were produced under free-field conditions. Spatialized audio provides headphone listeners with virtual sound sources that appear to be located outside the listener's head. The locus of perceived sound sources is three-dimensional. If head-tracking technology is available, virtual sound sources can be decoupled from the listener's head movements, if required.

5.3.3 3-D Audio Applications

Spatial audio can be useful whenever a listener is presented with multiple auditory streams, requires information about the positions of events outside of the field of vision, or would benefit from increased immersion in an environment. Possible applications of spatial audio processing techniques include the following:

- Complex supervisory control systems such as telecommunications and air traffic control systems
- Civil and military aircraft warning systems
- Teleconferencing and telepresence applications

- Virtual environments
- Computer–user interfaces and auditory displays, especially those intended for use by the visually impaired
- Arts and entertainment, especially video games and music

5.3.4 OSAW 3-D Audio System

There are two major 3-D audio systems available by Lake and AuSIM Engineering. Appendix I compares their main features. The AuSIM, Inc., Gold Series S101 Audio Vectorization System provides a very high-fidelity 3-D audio synthesis for the OSAW. This 3-D audio synthesis system provides superior synthesis fidelity and flexibility. The system uses logically layered, efficient high-level code that runs on industry-standard, commercially priced, general-purpose hardware. Hardware specific code is minimized. The system is fully compatible with most commercially available operating environments, including Win32, SGI, Sun, Mac, etc.

This software-based, industry-standard solution can be either run directly on a user's workstation or implemented as a peripheral server. The system will leverage operating system support for hardware-independent code. The system is also scalable in filter size versus number of sources synthesized. For a fixed processor configuration, filter length can be traded off for an increase in the number for filtered sources. All code is designed for symmetric multiprocessing, enabling overall performance to scale with processor speed and the number of processors. Each Gold Series S101 includes an auralization server that can vectorize eight channels with order-128 filters, an external eight-channel analog/digital interface, a high-fidelity closed headphone set, a headphone amplifier, cabling, client software for Win32, and ultrasonic head-tracking instrumentation. In the current OSAW configuration, the Gold Series S101 is used in a server mode. The Gold Series S101 system includes the following primary components:

1. Core 3-D positional audio rendering software library. Minimally, this library can link directly to any user application and run on any workstation running an operating system supporting Win32 and having a DirectX controllable sound card. This same library scales to use multi-processors and professional digital audio hardware interfaces.
2. Server software wraps the rendering library for use by remote clients through RS-232 control. This component includes a complimentary control client software library for a Win32 host.
3. Client software library supports RS-232 control for any additional customer-specified target host (e.g. SGI, Sun, Mac, etc.).
4. Server extension supports an alternative protocol (i.e., RCP Ethernet, USB, Firewire, etc.) Any server extension component shall include a complimentary control client software library for a Win32 host.
5. Client software library extension supports an alternative control protocol for a customer-specified target host (e.g., SGI, Sun, Mac, etc.).

5.3.5 3-D Audio Localization Summary. To summarize, testing and evaluation of the current 3-D sound synthesis technology at SSC San Diego and elsewhere suggest the following:

- It is highly certain that individualized auditory spatial (HRTF) filters provide high-fidelity directional cues for headphone delivered sounds.

- These synthetic 3-D audio spatial cues significantly improve discrimination between simultaneous sounds.
- These synthetic cues also provide an efficient method of directing visual gaze. (Correlated 3-D spatial cues significantly decrease reaction-time to visual stimuli.)
- It appears probable that synthesis by individualized auditory spatial filters does not introduce any distortions that might interfere with common listening tasks.
- It is also highly certain that non-individualized filters yield significantly poorer listening performance.
- The technology is available whereby individualized HRTF filters can be provided for any listener in an operationally convenient manner.
- However, even non-individualized HRTF filtering yields listening performance that is superior to that achieved by stereo headphones.

Taken together, the above statements indicate that synthetic 3-D sound technology, in conjunction with passive and/or active noise-cancellation headphone technology, has the potential of revolutionizing listening performance in fleet systems by eliminating the effects of distance and ambient noise levels without sacrificing perceptually relevant spatial information. In effect, 3-D audio synthesis technology promises to provide headphone listeners with a virtual anechoic chamber that includes multiple virtual sound sources mimicking physical speaker devices. Such a virtual three-dimensional sound-field can significantly improve listeners' ability to process multiple auditory information sources and maintain a new and better level of situation awareness.

The 3-D audio system will soon support the multiple users in client/server mode. We are also implementing wireless head tracking and audio broadcasting not only to provide user mobility, but to improve the packaging of the rack-mountable audio server.

6. CONCLUSIONS

GOMS task analysis shows that multi-modal HCI has strong potential over conventional workstation design. Creating a new model requires great effort; however, when multiple design iterations are examined, the result will be worth the effort. The current CPM-GOMS model requires the collection of empirical data and adjustment of model parameters. Such a model allowed the design to be optimized in minimum CPM workload, maximum productivity, and best efficiency for a given scenario of tasks.

The use of multiple real monitors is becoming more prevalent in the office and some military settings to support a multi-tasking environment. However, multiple monitors are expensive and require a large physical workspace that is often unavailable, especially in military settings. Furthermore, there is a decreasing payoff in effectiveness for adding more monitors as their placement becomes increasingly peripheral to the user. A less-expensive solution to the information access problem is to use fewer monitors, but add a large virtual screen area with an effective switching interface such as a workspace control diagram.

The multiple display studies suggest that two monitors with virtual workspaces enhanced by a hot-key-operated workspace control diagram gives optimal performance across various common multi-tasking environments.

We do not recommend touch screen in general Windows tasking. The touch-screen interface should be developed in a user intuitive mode such as Variable Action Buttons. The designer should consider the size and location of the touch screen to reduce user fatigue.

Speech technology is not mature enough to apply to U.S. Navy tactical application. Natural language and discourse technologies remain in the research realm. ASR and TTS play important roles in speech technologies. However, it shows promise in application command and control with limited vocabulary and a systematic analysis of functional requirements.

The 3-D audio localization technology can significantly improve operators' ability to process multiple auditory information sources and maintain a new and better level of situation awareness. The 3-D audio systems should be improved in the following areas: (1) client/server mode for multiple users, (2) digital audio routing to improve the communication, and (3) wireless head-tracking and audio broadcasting not only to provide user mobility, but to improve the packaging of rack mountable audio server.

We are continuously improving OSAW to support the future of the Q-70 design and acquisition program through the current Q-70 Technology Insertion program of SPAWAR PD-13 and NAVSEA PEO (EXW) PMS 440.

7. REFERENCES

Alwang, G. 1999. "Speech Recognition," *PC Magagine* (Dec 1), pp. 167–188.

Astheimer, P. 1993. "What You See is What You Hear: Acoustics Applied in Virtual Worlds," *IEEE Symposium on Virtual Reality*, San Jose, October, 1993.

Beringer, D. B. and J. G. Peterson. 1985. "Underlying Behavioral Parameters of the Operation of Touch-Input Devices: Biases, Models and Feedback," *Human Factors*, vol. 27, no. 4, pp. 445–458.

Card, S. K. and A. Henderson. 1987. "A Multiple, Virtual-Workspace Interface to Support User Task." In *Proceedings of ACM CHI+GI'87 Conference on Human Factors in Computing Systems and Graphics Interface* (pp. 53–59). ACM Press, New York, NY.

Card, S. K., T. P. Moran, and A. Newell. 1983. *The Psychology of Human-Computer Interaction.* Lawrence Erlbaum Associates, Hillsdale, NJ.

John, B. E. 1990. "Extensions of GOMS Analyses to Expert Performance Requiring Perception of Dynamic Visual and Auditory Information." In *CHI'90 Conference Proceedings* (pp. 107–115), 1–5 April, Seattle, WA.

Kendall, G. S., W. L. Martins, and S. L. Decker. 1989. *Spatial Reverberation: Discussion and Demonstration, Current Directions in Computer Music Research.* MIT Press: Cambridge, MA.

Kieras, David. 1993. *Using the Keystroke-Level Model to Estimate Execution Times*. University of Michigan, Ann Arbor, MI.

Lockheed Martin. 1998. "Display Technology Investigation for the AN/UYQ-70(V) Advanced Display System." Document Number 7373733, Rev. A (Oct). Lockheed Martin Tactical Defense Systems, St. Paul, MN.

Montaniz, F. and R. Mack. 1991. "A Comparison of Touch Interface Techniques for a Graphical Windowing Software Environment." In *Proceedings of the Human Factors Society 35th Annual Meeting* (pp. 290-294). Human Factors Society, Santa Monica, CA.

Obermayer, R.W., J. M. Linville, and F. P. Calantropio. 1999. *A Model for Analysis of the User Interface (MAUI).*

Potter, R., M. Berman, and B. Schneiderman. 1989. "An Experimental Evaluation of Three Touch Screen Strategies within a Hypertext Database," *International Journal of Human-Computer Interaction*, vol. 1, no. 1, pp. 41–52.

Potter, R., L. Weldon, and B. Schneiderman. 1988. "Improving the Accuracy of Touch Screens: an Experimental Evaluation of Three Strategies." *Proceedings of the Conference on Human Factors in Computing Systems* CHI '88 (pp. 27–32). ACM SIGCHI, New York, NY.

Sola, K.E. 1998. "The Military Display Dilemma," *Information Display* (Dec), pp. 28–31.

St. John, M., D. Manes, H. Oonk, and H. Ko. 1999. "Workspace Control Diagrams and Head-Mounted Displays as Alternatives to Multiple Monitors in Information-Rich Environments." In *Proceedings of the Human Factors and Ergonomics Society 43rd Annual Meeting* (pp. 438–442).

Watts, J. 1994. "A Cognitive Analysis of Navigation in the Computer Medium." Technical Report CSEL-1994-02. Cognitive Systems Engineering Laboratory, Ohio State University.

APPENDIX A

STRIKE COORDINATION WINDOW TASK

Table A-1. Strike Coordination Window Task.

Sequ. Nr.	Event	System Task	Operator Task
1	Alert	Get operator's attention	Recognize Alert
1a			Select Alert
1b			Invoke "Act On"
2	Mission Assignment Window opens		Manually? NO
2a			Select "Auto Assign Missions"
2b		Auto assign missions	
2c			Review Results
2d			Accepts Results
2e			Select "Create Taskings"
2f	Mission Tasking Window opens		
3	Mission Search Criteria Window opens	Provide "default search criteria"	Accept Criteria? YES
3a			Select Search
3b	Mission Search Results Window opens	Provide list of applicable missions	Review List of Missions
3c			Select Mission of Interest
3d			Select "Amplify"
3e	Mission Definition Page Window opens	Provide amplifying mission information	Review Mission Data
3f			Select "Close"
3g			Repeat 3c-3f as required
3h	MDP Window closes / Mission Search Results Window still open		Select Desired Mission
3i			Select "Apply"
3j		Pair Mission/Aimpoint to Target	
3k		Post pairing in Mission Assignments window	
3l			Repeat 3h-3k as required
3m			Select "Close"
3n	Mission Search Results Window closes		

Table A-1. Strike Coordination Window Task. (continued)

Sequ. Nr.	Event	System Task	Operator Task
3o			Select "Close" on Mission Search Criteria Window
3p	Close Mission Search Criteria Window / Missions Assignments Window still open		Select "Create Taskings"
3q	Mission Tasking Window opens		
4	Assign Platforms to Missions		Assign Manually? NO
4a			Select Mission(s) for Platform assignment
4b			Select "Auto Platform"
4c		Assign platform to missions using platform algorithm	
4d	Platform/Mission pairings are posted to Mission Tasking Window		Review pairings
4e			Accept Pairings? YES
4f			Repeat 4-4e as required
5	Mission Tasking Window still open		Create Coordinated Strike? YES
5a			Select "Create Coordinated Strike"
5b	Open Create Coordinated Strike Window		Enter Desired TOT (dd hhmmZ mmm yy)
5c			Enter time window around desired TOT (hh:mm)
5d			Select "OK'

Table A-1. Strike Coordination Window Task. (continued)

Sequ. Nr.	Event	System Task	Operator Task
5e	Close Create Coordinated Strike Window	Group missions that fall within TOT window/ assign C/S #	
5f	Update Mission Taskings with coordinated strike		Create Coordinated Strike? YES
5g			Repeat 5a-5f as required
6	Mission Taskings Window still open		Generate Tasking? YES
6a			Select Mission? C/S
6b			Select "Generate Tasking Message"
6c		Auto-create LSP or Indigo message, as required.	
6d	Open OTG Message Window		Review Message Content
6e			Make Changes? NO
6f			Select "Send Tasking Message"
6g	Close OTG Message Window / Mission Taskings Window still open	Xmit LSP / Indigo	Generate another tasking? YES
6h			Repeat 6a-6f as required
6i			Generate another Tasking? NO
6j	Mission Taskings Window still open		Select "Close"
6k	Close Mission Tasking Window		

APPENDIX B
WINDOWS USED FOR WINDOW TASK

MISSION ASSIGNMENTS #001 (Day 01)

Target Name	Target BE	Target Location		Weapon Type	Salvo Size	Time on Target	Mission ID	Aimpoint ID
Mirim Airfield	1234AZ2157	30-53-32N	111-43-32E	LAC-D	002	23 0234Z NOV 96	001-001-54344	AB
aaaaaaaaaaaaaaaaaaaaaaaa	xxxxxxxxxx	nn-nn-nna	nnn-nn-nna	aaaaa	nnn	dd hhmmZ mmm yy	nnn-nnn-nnnnn	a
aaaaaaaaaaaaaaaaaaaaaaaa	xxxxxxxxxx	nn-nn-nna	nnn-nn-nna	aaaaa	nnn	dd hhmmZ mmm yy	nnn-nnn-nnnnn	a
								a
aaaaaaaaaaaaaaaaaaaaaaaa	xxxxxxxxxx	nn-nn-nna	nnn-nn-nna	aaaaa	nnn	dd hhmmZ mmm yy	nnn-nnn-nnnnn	a
aaaaaaaaaaaaaaaaaaaaaaaa	xxxxxxxxxx	nn-nn-nna	nnn-nn-nna	aaaaa	nnn	dd hhmmZ mmm yy	nnn-nnn-nnnnn	a
aaaaaaaaaaaaaaaaaaaaaaaa	xxxxxxxxxx	nn-nn-nna	nnn-nn-nna	aaaaa	nnn	dd hhmmZ mmm yy	nnn-nnn-nnnnn	a
								a
aaaaaaaaaaaaaaaaaaaaaaaa	xxxxxxxxxx	nn-nn-nna	nnn-nn-nna	aaaaa	nnn	dd hhmmZ mmm yy	nnn-nnn-nnnnn	a
aaaaaaaaaaaaaaaaaaaaaaaa	xxxxxxxxxx	nn-nn-nna	nnn-nn-nna	aaaaa	nnn	dd hhmmZ mmm yy	nnn-nnn-nnnnn	a
aaaaaaaaaaaaaaaaaaaaaaaa	xxxxxxxxxx	nn-nn-nna	nnn-nn-nna	aaaaa	nnn	dd hhmmZ mmm yy	nnn-nnn-nnnnn	a
								a

[Auto-Assign Missions] [Manually Assign Missions] [Create Taskings] [Close]

MISSION TASKINGS #001

C/S	Ref #	Mission ID	Verification #	Weapon Type	Salvo Size	Time on Target	Platform Assigned
nnnnn	nnn	nnn-nnn-nnnnn	nnnnn	aaaaa	nnn	dd hhmmZ mmm yy	aaaaaaaaaaaaaaa
nnnnn	nnn	nnn-nnn-nnnnn	nnnnn	aaaaa	nnn	dd hhmmZ mmm yy	aaaaaaaaaaaaaaa
nnnnn	nnn	nnn-nnn-nnnnn	nnnnn	aaaaa	nnn	dd hhmmZ mmm yy	aaaaaaaaaaaaaaa
nnnnn	nnn	nnn-nnn-nnnnn	nnnnn	aaaaa	nnn	dd hhmmZ mmm yy	aaaaaaaaaaaaaaa
nnnnn	nnn	nnn-nnn-nnnnn	nnnnn	aaaaa	nnn	dd hhmmZ mmm yy	aaaaaaaaaaaaaaa
nnnnn	nnn	nnn-nnn-nnnnn	nnnnn	aaaaa	nnn	dd hhmmZ mmm yy	aaaaaaaaaaaaaaa
nnnnn	nnn	nnn-nnn-nnnnn	nnnnn	aaaaa	nnn	dd hhmmZ mmm yy	aaaaaaaaaaaaaaa
nnnnn	nnn	nnn-nnn-nnnnn	nnnnn	aaaaa	nnn	dd hhmmZ mmm yy	aaaaaaaaaaaaaaa
nnnnn	nnn	nnn-nnn-nnnnn	nnnnn	aaaaa	nnn	dd hhmmZ mmm yy	aaaaaaaaaaaaaaa

[Auto-Platform M^3]

[Manually Perform Platform M^3] [Create Coordinated Strike] [Generate Tasking Message] [Close]

MISSION SEARCH CRITERIA

Search for Mission folders based on:

☒ Target Name : aaaaaaaaaaaaaaaaaaaaaaaaaa

☐ Target BE : XXXXXXXXXX

☒ Target Location : nn-nn-nna nnn-nn-nna

Search | **Cancel** | **Close**

MISSION SEARCH RESULTS

Mission Search Results for: Target Name, Target BE, Target Location

Using MDP's windows to display this mission data

- Show Search results window .
- Show MF ID List window .
- Mission Definition page will give amplifying info re: a mission
- Show mission textually

{Note: Operator can display missions textually or graphically.)

Apply | **Amplify** | **Close**

PERFORM PLATFORM M^3

Perform Platform M^3 for Mission ID: nnn-nnn-nnnnn

Perform Platform M^3 | Cancel

PLATFORM M^3 SEARCH RESULTS : MSSN# nnn-nnn-nnnnn

Platform TN	Platform Name	DTG	Latitude	Longitude
aaaaaaaa	aaaaaaaaaaaaaaaaaaaa	dd hhmmZ mmm yy	nn-nn-nna	nnn-nn-nna
aaaaaaaa	aaaaaaaaaaaaaaaaaaaa	dd hhmmZ mmm yy	nn-nn-nna	nnn-nn-nna
aaaaaaaa	aaaaaaaaaaaaaaaaaaaa	dd hhmmZ mmm yy	nn-nn-nna	nnn-nn-nna
aaaaaaaa	aaaaaaaaaaaaaaaaaaaa	dd hhmmZ mmm yy	nn-nn-nna	nnn-nn-nna
aaaaaaaa	aaaaaaaaaaaaaaaaaaaa	dd hhmmZ mmm yy	nn-nn-nna	nnn-nn-nna
aaaaaaaa	aaaaaaaaaaaaaaaaaaaa	dd hhmmZ mmm yy	nn-nn-nna	nnn-nn-nna

Apply | Cancel | Close

CREATE COORDINATED STRIKE

Time on Target:

Window (+- TOT): **minutes**

OK | **Close**

APPENDIX C

STRIKE COORDINATION EXECUTION TASK

Table C-1. Strike Coordination Execution Task.

Sequ. Nr.	Event	System Task	Operator Task
1	Prepare for Coordinated Strike Execution Monitoring		Open "Monitor Strike Execution" Window via OSV or menu selection
2		Open "Monitor C/S Execution Window	Select Coordinated Strike for monitoring via option menu
3			Select C/S control mode via Option menu
4			Select the items for display
5			Set "show recommendations" and method
6			Select "OK"
7		Open C/S 9001 control display	
8		Display Missile Activity	Maintain situational awareness
9	Alert Received	Get Operator attention/ display urgent action alert	Select alert and select "Act On"
10		Display failure and recommend action	Gain situational awareness
11			Accept system recommendation
12	Recover from failure		Select missile 2143; drag and drop on 70 AA
13		Open question dialog to confirm missile order	Read question; select YES
14	Command Missile Flex	Send msg to missile	
15		Display new routing; update mission timeline	Maintain operational awareness
16	Provide post-strike analysis and recommendations	Open post-strike analysis / recommendations window	Read information; absorb information; select "Close"
17		Determine if Post-strike report is required, open question dialog	Read information; determine course of action; select YES
18		Create / Xmit post-strike report	

Table C-1. Strike Coordination Execution Task. (continued)

Sequ. Nr.	Event	System Task	Operator Task
19		Determine if DDG 51 is to be tasked for ready spare, open question dialog	Read information; determine course of action; select UES
			Maintain awareness for need to perform execution task sequence again.

APPENDIX D

WINDOWS USED FOR EXECUTION TASK

MONITOR COORDINATED STRIKE EXECUTION

File | Help

Select Coordinated Strike: 9001

Coordinated Strike Control Mode: Positive

DISPLAY :

Route Control Measures: ☐ Missiles: ☐

Missile Messages: ☐ Routes: ☐

Post-Strike Analysis / Recommendations: ☐ Aimpoints: ☐

Show Launch Control Recommendations: ☐

Text: ◇ Graphics: ◇ Both: ◇

Confirm all Missile Orders: ☐

OK | Reset | Defaults | Cancel

Positive-- system recommends actions; operator approves/changes
Negation- system initiates actions operator may override
Automatic- system acts automatically without operator input

POST-STRIKE ANALYSIS / RECOMMENDATIONS

ANALYSIS:	RECOMMENDATIONS:
C/S 9001: Expended 7 Missiles Met TOT Overall effectiveness: 0.77 70 AA: BDI: 0.92 Tasked: 0.75 71 AA: BDI: 0.75 Tasked: 0.65 71 AB: BDI: 0.60 Tasked: 0.65 60 AB: BDI: 0.80 Tasked: 0.75	1. Submit Post-strike Report: 70 AA 71 AA: 60 AB: Mission Complete / Successful 2. Task DDG 51 to fire ready-spare for 71 AB.

CLOSE HELP

QUESTION

? CREATE POST-STRIKE REPORT FOR C/S 9001

YES NO HELP

QUESTION

? TASK DDG 51 TO FIRE READY-SPARE, MISSION HY 1, 60 AB?

YES NO HELP

APPENDIX E

MODEL ASSUMPTIONS AND PARAMETERS

The analyses were preliminary estimates based on the cautions presented in the literature and therefore must be checked empirically. Initial parameter estimates and assumptions are presented below and were used only for model checkout and developing example outputs.

Home time only if hand is not on mouse (keyboard)

Estimated probability of a pointer being lost = 0.05

Estimated probability of voice recognition error = 0.20

Maximum CPM parallel activity is assumed

Time for a cognitive cycle = 50 msec

Time for eye movement = 30 msec

Time for visual perception = 100 msec

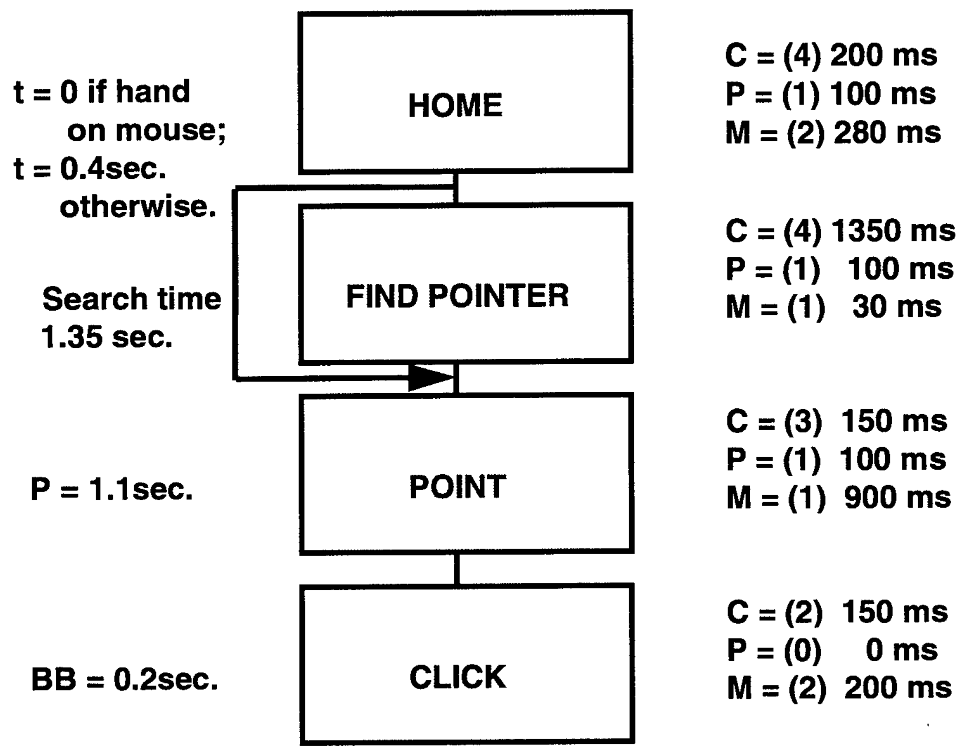
Hand movement per KLM (could add Fitts' law estimate) and CPM lit

Utterance time = 130 msec. per syllable (170 msec. unpracticed)

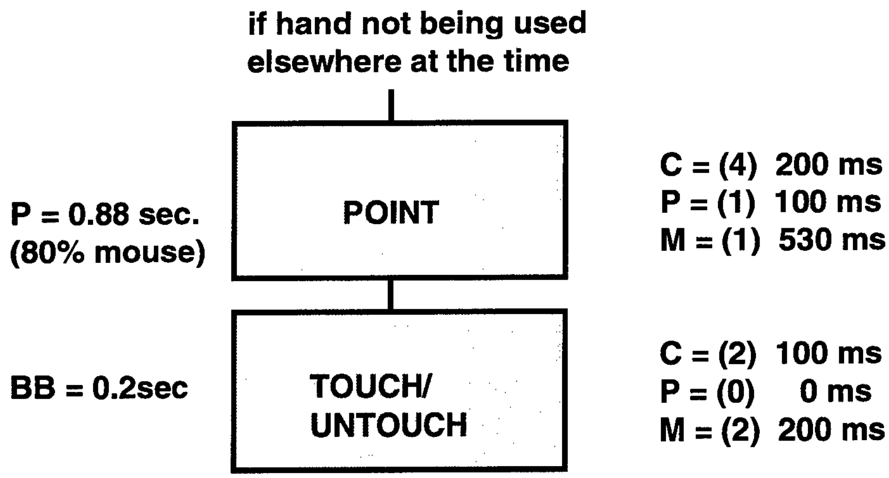
APPENDIX F
MODEL BLOCK DIAGRAMS

Note: The C, P, and M in the block diagrams stand for Cognitive, Perceptual, and Motor respectively.

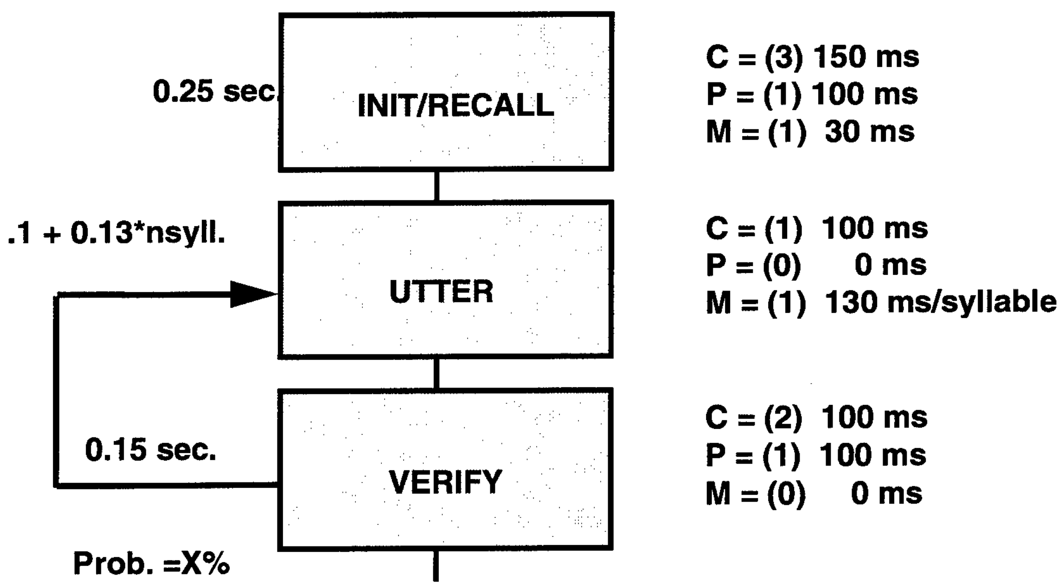
1. Mouse Mode



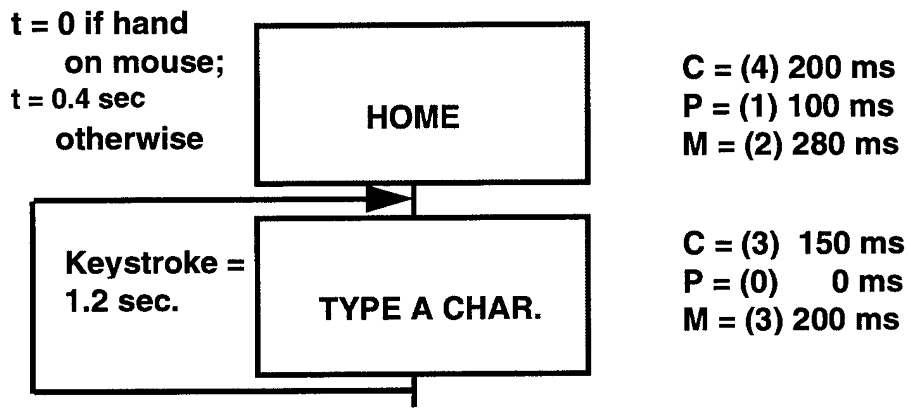
2. Touch Mode



3. Speech Mode



4. Keyboard Mode



APPENDIX G
EXAMPLE OUTPUTS

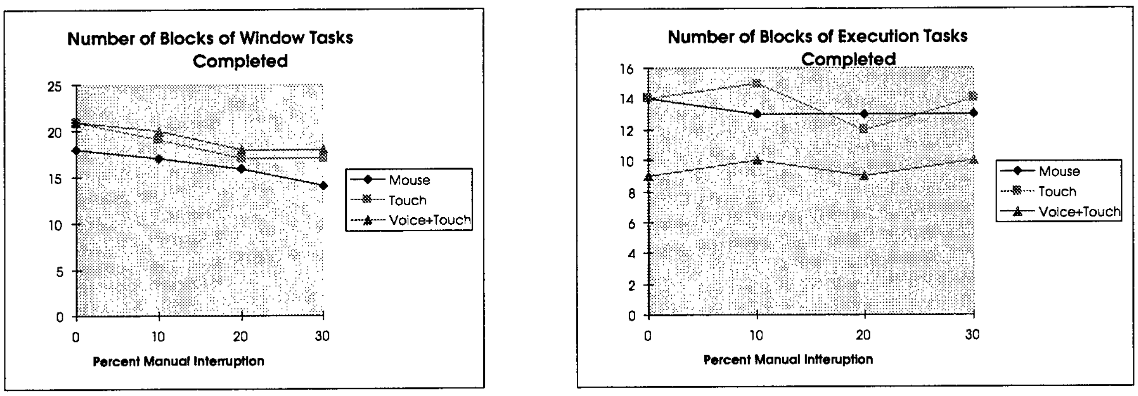


Figure G-1. Example output: number of blocks of window and execution tasks completed (external manual interruptions).

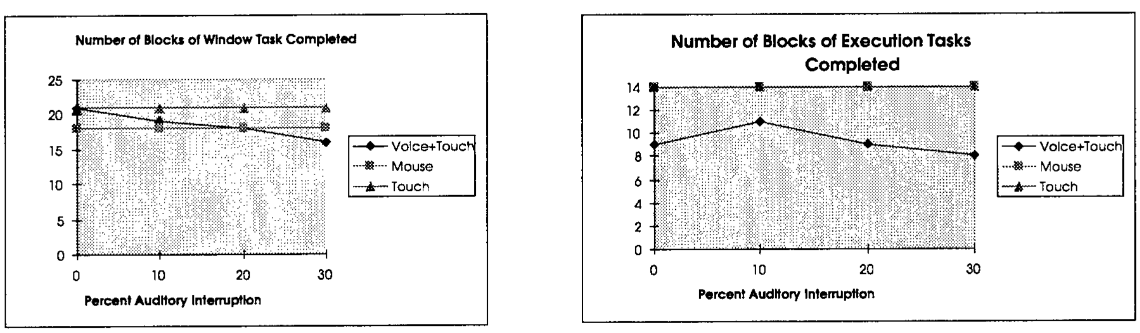


Figure G.2. Example output: number of blocks of window and execution tasks completed (external auditory interruptions).

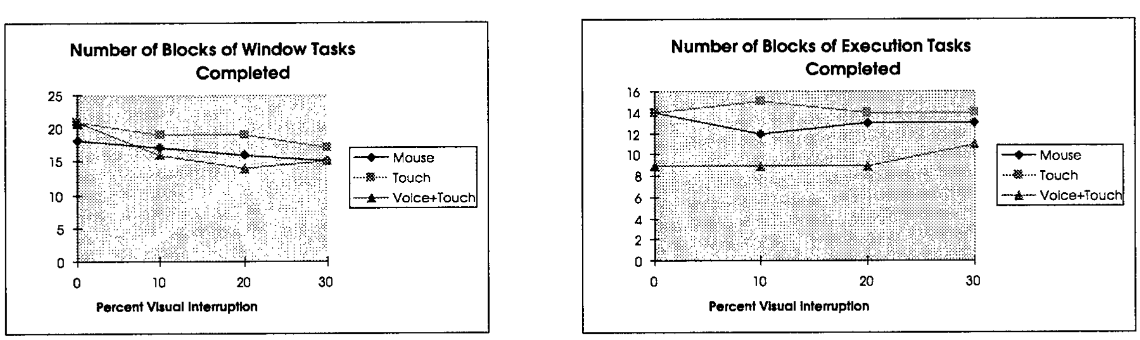


Figure G-3. Example output: Number of blocks of window and execution tasks completed (external visual interruptions).

Mouse Links	External	Mouse	Keyboard	Screen 1	Screen 2
External	0	10	0		
Mouse	10	434	18		
Keyboard	0	18	16		
Screen 1					
Screen 2					

Touch Links	External	Mouse	Keyboard	Screen 1	Screen 2
External	0		0	9	1
Mouse					
Keyboard	0		19	19	0
Screen 1	10		19	364	31
Screen 2	0		0	32	60

Voice+Touch Links	External	Mouse	Keyboard	Screen 1	Screen 2
External	0		2	8	0
Mouse					
Keyboard	0		20	14	1
Screen 1	5		17	62	3
Screen 2	0		1	3	3

Figure G-4. Example output: link analysis of hand movements.

Mouse Links	Screen 1	Screen 2
Screen 1	375	37
Screen 2	37	41
Mdist	400	

Figure G-5. Example output: link analysis of mouse movements.

APPENDIX H
COMMERCIAL SPEECH RECOGNITION SYSTEMS

Dragon Systems introduced the first general-purpose continuous-speech recognition program for the PC in June 1997; IBM Corporation followed soon after. The performance of the speech recognition accuracy has been improved and boosted to as high as 98 percent. In 1997, when we started developing the speech recognition system for OSAW, the IBM ViaVoice development tool was only available for the Windows environment.

We reviewed five general-purpose continuous-speech recognition programs for the PC: Nuance Commnunications Nuance 6, Dragon NaturallySpeaking, IBM ViaVoice, L&H Voice Xpress Plus, and Philips FreeSpeech. The major features of the speech recognition software are listed and compared (Alwang, 1999). In addition, the Nuance 6 features are also listed in table 17. Nuance 6 only supports multiple platforms such as NT, Sparc Solaris, DEC UNIX, etc. and networked client/server architecture providing flexible deployments options.

Table H-1. Speech recognition system comparison.

	Nuance 6	ViaVoice Pro Millennium Edition	L&H Voice Express Professional 4	Dragon Naturally Speaking Professional 4	FreeSpeech 2000
Company	Nuance Communications www.nuance.com	IBM www.ibm.com/viavoice	Lernout & Hauspie www.lhs.com	Dragon Systems www.dragonsys.com	Philips www.speech.philips.com
Accuracy after (%)	97	98	94	96	93
Throughput (words/min)		31	27	35	24
Development Tools	Yes	Yes		Yes	Yes
Unix Version	Yes	No	No	No	No
Base vocabulary/ Expandable Size		64k/2000k	34k/64k	160k/240k	60k/670k
Support Multiple Users	Yes	Yes	Yes	Yes	Yes
Text to speech		Yes	Yes	Yes	Yes

Table H-1. Speech recognition system comparison. (continued)

	Nuance 6	ViaVoice Pro Millennium Edition	L&H Voice Express Professional 4	Dragon Naturally Speaking Professional 4	FreeSpeech 2000
Command macros		Yes	Yes	Yes.	No
Client/Server	Yes	No	No	No	No
Training (min)		30	60	60	15

APPENDIX I

COMMERCIAL 3-D AUDIO SYSTEMS

There are two major 3-D audio systems available by Lake and AuSIM Engineering Solutions. AuSIM was founded in 1998 to provide positional 3-D audio simulation solutions to mission-critical applications. In 1996, Crystal River Engineering (CRE) was acquired by Aureal Incorporated, who has developed a 3-D audio chipset named A3D. This major undertaking required all of the acquired CRE resources and, thus, the customers with mission-critical applications were left with only legacy CRE products. With encouragement from Aureal, AuSIM was launched to maintain and advance the highest level of positional 3-D audio technology, exclusively for high-end simulation and academic research. Since 1991 Lake Technology Limited in Australia has been developing 3-D audio systems for real-time acoustic simulation. Lake's digital technology allows for realistic simulation for room acoustics and manipulation of the virtual sound environment through a computer. The research products are widely used by the research and academic organization. Table 18 compares the specification of 3-D audio systems. Both systems are based on HRTF technology to localize sound sources. APIs are available to develop the customized 3-D audio applications.

Table I-1. 3-D Audio System specification comparison.

Factors		AuSim www.ausim3d.com	Lake www.lake.com
Channel details	Input source streams	Practical limit is 64 channels. Eight is standard. 44.1, 48.0 & 96 kHz SampleRates	Max 16 total I/O on CP4 Much more for Huron (maximum dependant on configuration). 32 I/O or more is possible. Digital or analog input, 44.1 or 48 KHz sample rates.
	Output binaural streams	The practical limit is 32 binaural pairs.	As above for physical I/O connections. Maximum of four binaural output streams with CP4 DSP power, many more with Huron, again depending on configuration.
Localization	Max. number of channels	Any number can be localized. There is a trade-off between fidelity and the number of simultaneously rendered sources.	Unlimited number of sound sources at any one time, with closest eight rendered at any one time.
	Dynamic Range	24bit, > 120 dB	24 bit, Digital >110 dB
	Latency	< 1 msec	<1 msec
	Max. Delay	> 1 msec	Please clarify terminology! Input to output analog converter delay <1ms (inherent in all analog converters)
	Update Rate	> 60 Hz	>60 Hz

Table I-1. 3-D Audio System specification comparison. (continued)

Factors		AuSim *www.ausim3d.com*	Lake *www.lake.com*
Environmental simulation	Room response format	Room acoustics are dynamically modeled.	B-Format or any combination of W, X, Y and Z impulse responses.
	Room Simulation	Unlimited number of reflectors or diffractors.	Yes. Exact number of rooms which can be modeled is dependant on the DSP power available.
	Door Simulation	Model any sound barrier or combination of barriers in free space.	Yes. Leakage of sound from one room to another is modeled, along with the amount that the door is open. Dependant on DSP power available.
HRTF datasets	Filter Length	Up to 16384 taps Standard HRTFs are nominally 128 taps	Optimized and compressed into Lake's proprietry format. HRTF data supplied with system.
	Sampling Rates	44.1 & 48 kHz	44.1 or 48 KHz.
	Sample word size	16-bit integer, 32-bit floating point	24-bit integer internal processing.
	Spatial grid size	No limit, any rectangular, 3-D nonlinear datasets are supported. User may load own datasets.	Supplied in Lake's proprietary HRTF format.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-01-0188*

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
01-07-2000	Technical	October 1995 to September 1999

4. TITLE AND SUBTITLE
OPEN SYSTEMS ADVANCED WORKSTATION TRANSITION REPORT

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
0603707N

6. AUTHOR(S)
H. Ko

5d. PROJECT NUMBER

5e. TASK NUMBER
DN308298

5f. WORK UNIT NUMBER
CE31

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
SSC San Diego
San Diego, CA 92152-5001

Pacific Science & Engineering Group
6310 Greenwich Drive, Suite 200
San Diego, CA 92122-5918

8. PERFORMING ORGANIZATION REPORT NUMBER
TR 1822

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Office of Naval Research
800 N. Quincy Avenue
Arlington, VA 22217-5160

Naval Sea Systems Command
2531 Jefferson Davis Highway
Arlington, VA 22242-5160

10. SPONSOR/MONITOR'S ACRONYM(S)
ONR/NAVSEA

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
U.S. Navy Command and Control systems require complex task support from shipboard workstations that receive information from various sources and display information multiple modalities of operators. The specific multi-modalities involve touch and voice inputs with visual and 3-D auditory outputs. This report describes the development of the Open Systems Advanced Workstation (OSAW) and presents guidelines for using multi-modal technologies.

15. SUBJECT TERMS
Mission Area: Command, Control, and Communications
decision support   decision-making   human factors
decision aids   man-machine systems

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			H. Ko
U	U	U	UU	88	**19b. TELEPHONE NUMBER** *(Include area code)* (619) 553-8013

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

INITIAL DISTRIBUTION

D0012	Patent Counsel	(1)
D0271	Archive/Stock	(6)
D0274	Library	(2)
D027	M. E. Cathcart	(1)
D0271	D. Richter	(1)
D44	J. Martin	(2)
D441	O. Larson	(1)
D441	G. Osga	(1)
D44210	H. Ko	(20)
D44210	R. Smillie	(1)
D44213	D. Lulue	(1)
D44215	J. Kaiwi	(1)

Defense Technical Information Center
Fort Belvoir, VA 22060–6218 (4)

SSC San Diego Liaison Office
Arlington, VA 22202–4804

Center for Naval Analyses
Alexandria, VA 22302–0268

Office of Naval Research
Attn: NARDIC (Code 362)
Arlington, VA 22217–5660

Government-Industry Data Exchange
Program Operations Center
Corona, CA 91718–8000

Office of Naval Research
Arlington, VA 22217–5660 (2)

Naval Sea Systems Command PMS 440
Arlington, VA 22242 (10)

Space and Naval Warfare Systems Command
San Diego, CA 92110–3127 (2)

Pacific Science and Engineering Group
San Diego, CA 92122 (2)