THE USE OF MODEL-BASED, WINDOW DISPLAY INTERFACES IN REAL TIME SUPERVISORY CONTROL SYSTEMS

A THESIS

Presented to

The Faculty of the Division of Graduate Studies

By

Donna Lynn Saisi

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in Industrial Engineering

Georgia Institute of Technology

May, 1986

THE USE OF MODEL-BASED, WINDOW DISPLAY INTERFACES IN

REAL TIME SUPERVISORY CONTROL SYSTEMS

Approved:

Dr. Christine Mitchell, Advisor

Dr. T. Govindaraj

Dr. Rich. ..enneman

Date approved by Advisor 21 May 1986

ACKNOWLEDGEMENTS

First of all, I would like to thank my advisor, Chris Mitchell, whose insight, enthusiasm and hard work made the GT-MSOCC research a valuable and rewarding learning experience. I want to thank my committee members, Dick Henneman and T. Govindaraj, for their advise and suggestions throughout this project. I also thank John Hammer for the literature he provided on computer windows.

I wish to acknowledge Russ Heikes for his time and patience. His statistical expertise was an invaluable resource. I also appreciate the help from Richard Henderson who provided literature, wrote programs and answered questions concerning SAS statistical software.

The GT-MSOCC code was produced through the efforts of a number of people who I would like to acknowledge: Michelle and Helmut Forren, Chris Mitchell, Richard Robison, and Rakesh Bisaria.

I wish to thank Janet Fath for helping me organize the introduction to this paper, a particularly troublesome chapter. Also, a very special thanks is given to Janet Fath and Dick Henneman for their support and friendship.

Finally, I am indebted to Richard Robison for the many hours and lost sleep spent working on this paper. His friendship, emotional support, and uncountable acts of kindness made the completion of this thesis possible.

This research was supported by NASA Goddard Space Flight Center Contract Number NAS5-28575.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

THE USE OF MODEL-BASED, WINDOW DISPLAY INTERFACES IN REAL TIME SUPERVISORY CONTROL SYSTEMS

Donna Lynn Saisi

238 Pages

Directed by Dr. Christine Mitchell

Windowing technology may be a valuable design technique for presenting information to operators of real time, data intensive supervisory control systems. Using a windowing system, multiple sources that reflect different aspects of system state can be displayed simultaneously on a single screen. To evaluate the effectiveness of a window-based interface, two user interfaces to a simulation of a NASA satellite communications system were designed. One interface consisted of displays that were typical of those used in command-and-control systems. The second interface was based on an operator function model of the supervisory controller of the simulated system. The operator function model determined the contents and placement of computer windows in the user interface. The model also determined the needed set of windows to perform each operator control function. The development of the window interface is discussed as well as results from the experiment that compared the two interfaces.

Eleven measures that reflected operator performance were analyzed. Subjects using the window interface operated the system significantly better on nine of the measures. Performance was also less variable with the window interface across session and subject.

CHAPTER I

INTRODUCTION

The decreasing cost of computer hardware (e.g., memory and microprocessors) has made interactive computer graphics an accessible resource for many computer applications (Foley and Van Dam, 1982). One application of interactive computer graphics is in the area of process control. Foley and Van Dam cite specific process control applications of interactive graphics including arcade games and flight simulators, as well as real-world applications such as status displays for refineries, power plants and computer networks. In addition, interactive graphic displays can be used by military commanders, flight controllers and also spacecraft controllers who monitor satellite data transmission and initiate corrective procedures when problems arise (Foley and Van Dam, 1982).

A powerful new technology in the area of interactive graphics is the development of window management systems that control multiple overlapping windows. A window is an area on a video display terminal in which a page or a partial view of a page is displayed. Foley and Van Dam (1982) comment that "each window is in essence a variable-size virtual screen that reflects the progress of some activity" (p.16). They state that an advanced window management system liberates the user from sequentially accessing and processing data.

One area for which windowing technology may be beneficial is for interfaces to automated process control systems. Traditional process control rooms contain panels of single-sensor, single-indicator displays (e.g., a temperature gauge) that individually reflect specific physical aspects of the system (Rasmussen, 1984). Rasmussen observes that a major trend affecting designers of modern human-machine interfaces is the rapid development of computer-based information technology. Rather than forcing the operator to search arrays of hardware-oriented data, Rasmussen suggests matching computer information processing to the decision processes of the human. One alternative to conventional interfaces is an interface of windows containing preprocessed, task-relevant data. Windows comprising such an interface are controlled by a model which Rasmussen states is a description of operator decision tasks and the necessary information required to conduct these tasks. When operators are provided with task-specific information, preliminary data search, elementary calculation and information integration is reduced.

An interface based on a model of the system operator's information needs may facilitate operation of an automated process control system. A windowing system incorporated into this design may contribute to the easy access of information by the human operator. In this thesis, a model-based, multiple window display for presenting data to operators of a real-time, interactive supervisory control system is developed and evaluated. Before describing this application of windowing technology, however, some recent literature on computer windows will be reviewed. Several different types of windows are defined in the literature as well as applications that have incorporated these different windows. Some research also exists that has explored potential advantages of windowing technology.

Characteristics and Applications of Windows

Little research currently exists that presents general guidelines for the use of computer windows. The most prevalent types of literature describe either features of windowing systems or describe specific applications that have incorporated windowing technology. Some features of windowing systems are defined below, and their applications are described in subsequent paragraphs.

Windows have been implemented on a number of computer systems and have been defined in numerous ways. Windows can be distinguished by the six following characteristics:

1. Hardware. Windows have been implemented on text video terminals, on graphics terminals, and on bitmapped raster-graphics computers.

2. Contents of Windows. Depending on the computer system, the terminal and the windowing software, the contents of windows may be alphanumeric, graphic or some combination of these. The contents of windows may or may not incorporate color.

3. Window Arrangement. CRT windows may have one, two or 2 1/2 dimensions (Figure 1.1). One dimensional windows are areas separated by horizontal lines drawn across the width of the screen. Different processes run in each one dimensional window. Two dimensional windows, unlike one dimensional windows, can differ in width as well as height. Two

dimensional windows may be juxtapositioned or separated by screen space. Finally, windows may overlap one another; Card, Pavel and Farrel (1984) define these as 2 1/2 dimensional windows.

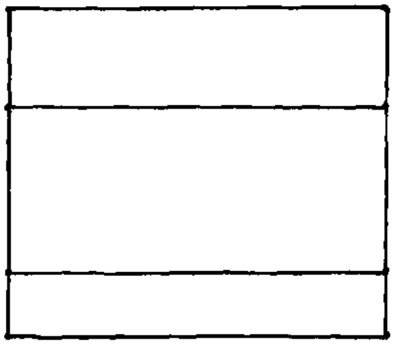
4. Processing in Windows. Some windowing environments contain static windows and do not allow processing of information within windows. Others allow data processing in a single window or in several windows. If data is processed in multiple windows, the windowing system may afford either sequential processing or multiprocessing.

5. Manipulation of Window Position. Windows in some systems have fixed locations. Other windowing environments allow windows to be located at or transported to any position on the screen. If windows do not have predetermined coordinates, window positioning may either be system-defined or user-defined.

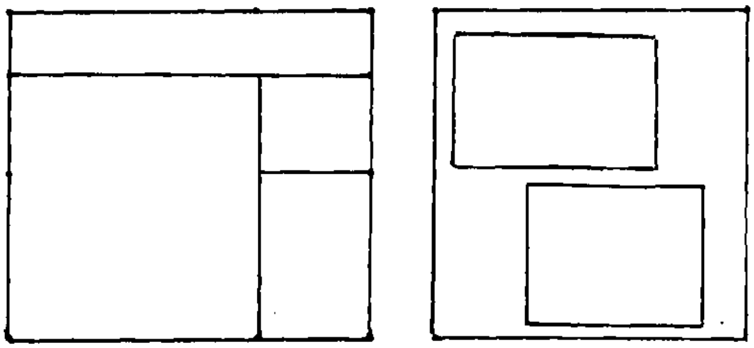
6. Space conservation. Card et al. (1984) describe several methods to allocate screen space effectively by presenting windows at different levels of detail. Bifocal windows compress windows that are not currently of interest at the side of the screen. An optical fish-eye window compresses information so that is appears like the image in a convex mirror. Logical fish-eye windows display in greater detail some of the information contained in a window. Zooming windows increase in size, and either the contents (usually graphics) enlarge with the window, or else the amount of exposed data is increased. Finally, to conserve screen space, windows can be represented as icons, or very small pictures, that may be selected and expanded into windows.

Windows can take on many forms as they differ on the characteristics described above. Table 1.1 provides a number of current window applications and their associated characteristics. These existing systems will be described in the following paragraphs.

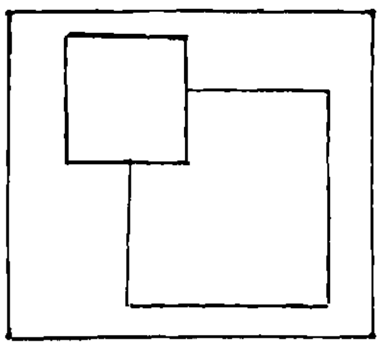
The first application in Table 1.1 is boxing analysis, which is a method for organizing data on a full screen display into two dimensional windows (Steveler and Wasserman, 1984). This approach groups proximate, alphanumeric data on a text video terminal by framing sets



One dimensional windows



Two dimensional windows: juxtapositioned and separated



2 1/2 dimensional windows

Figure 1.1 Types of Window Arrangement.

Table 1.1
A List of Six Windowing Applications and
Their Characteristics

WINDOW SYSTEM	TERMINAL	CONTENTS/ COLOR	DIMEN- SION	PROCESSING	WINDOW POSITION	SPACE CON- SERVATION
Boxing Analysis	Text Video	Alphanumeric/ No	2	None	Fixed	None
FLAIR	Graphics	Alphanumeric, Graphics/Yes	2	Sequential	System Defined	None
TRIP	Bitmapped Graphics	Alphanumeric, Graphics/No	2	Multi- processing	System Defined	None
Star	Bitmapped Graphics	Alphanumeric, Graphics/No	2 1/2	Multi- processing	User Defined	Icons
Smalltalk	Bitmapped Graphics	Alphanumeric, Graphics/No	2 1/2	Multi- processing	User Defined	Zooming Contents, Zooming Windows
Maryland Windows	Text Video, Graphics	Alphanumeric	2 1/2	Multi- processing	User Defined	Zooming Windows

of items that are completely surrounded with blank spaces. Boxing analysis is used on static data pages (e.g., a patient's medical record), and not on dynamic computer processes.

FLAIR (Wong and Reid, 1982) provides another application of windowing technology. FLAIR is a graphics design tool that incorporates five juxtapositioned, two dimensional windows that update sequentially on a color, graphics monitor. Windows in FLAIR differ from boxing analysis in that each FLAIR window performs a unique function, rather than serving as a static storage box for a data set. Windows on the FLAIR system allow the user to enter commands, view the resulting construction of icons, perform arithmetic calculations, receive error messages, and be reminded of possible commands on a single screen.

A third windowing system is TRIP (Gould and Finzer, 1982). This system is a teaching aid for learning to solve algebraic motion problems that relate time, rate and distance. Whereas FLAIR has only one window active at any given time, TRIP has multiprocessing windows. The TRIP system positions bitmapped, animated icons on the screen to represent motion problems pictorially and dynamically.

The next application is Xerox's Star Professional Workstation (Purvy, Farrell and Klose, 1983), a system containing iconic representations of documents, filedrawers, folders and in/out baskets. Icons can be selected with a mouse, and contents of the windows that they represent can be used or manipulated. For example, file icons may be enlarged to full-sized windows on the screen to be read or edited. The contents of windows in this system may be alphanumeric or bitmapped graphic, and these windows may overlap.

The last two windowing applications depicted in Table 1.1 are general purpose windowing systems used to develop specific applications such as TRIP and the Xerox's Star Professional Workstation. These examples are representative of a number of advanced, general purpose windowing systems that have been created (Meyrowitz and Moser, 1981; Stallman, Weinred and Moon, 1983; Teitelman, 1974; Williams, 1983). The development of bitmapped terminals has allowed multiple processing on various areas of a single screen (Pike, 1983). This ability has led to the creation of windowing systems in which a single display screen is divided into overlapping areas that update asynchronously. The Smalltalk system was the first windowing environment to contain overlapping, multiprocessing, alphanumeric and bitmapped graphic windows (Goldberg and Kay, 1976; Goldberg and Robson, 1983).

Windowing systems have been created that run not only on bitmapped displays, but also on standard 24 x 80 character alphanumeric and graphics terminals. One such system is the Maryland Window System (Weiser, Torek, Trigg and Lyle, 1983). The Maryland Window System, which runs on Berkeley Unix versions 4.1 and 4.2, is a windowing environment for manipulating multiple, 2 1/2 dimensional, alphanumeric windows. Areas on the screen can be framed and labeled, and these windows can be moved, covered or placed on top of other windows. Multiple processes can execute within various windows on a single screen. The Maryland Window system and other advanced windowing systems provide a compact, inexpensive, multiple processing work environment.

As shown in this section, windows can take many forms and have been used for a variety of very different applications. One windowing application that has not yet been researched is the use of windows in a supervisory control environment. The next section examines potential benefits of using windows in human-computer interfaces for supervisory control systems.

Potential Utility of Windows in Supervisory Control Systems

One possible use of an advanced windowing environment is as an interface to a supervisory control system. Windowing technology may be a valuable design technique for presenting information to operators of real-time, data-intensive, supervisory control systems. User workstations for such systems often consist of several monitors, each of which is used to display as many as several hundred different display pages. Windowing technology is a technique that can be used to condense information from several display pages onto a single screen, thereby increasing the information content of a display and reducing the difficulty of accessing information.

A supervisory control system is a system that operates in an automatic or semi-automatic mode. During normal operations, the system functions without human operator intervention. A supervisory controller is a person responsible for monitoring the system for malfunctions, determining the source of any problem and intervening to correct the situation. The supervisory controller may also be responsible for taking over control of the system when the automatic controller is unable to perform its operations.

Traditionally, operators monitored system state by reading one-sensor, one-indicator display devices. The development of high-speed, digital computers has enabled data reflecting a supervisory control system's state to be displayed on computer terminals. Sheridan (1976, 1984) defines supervisory control as the situation where a human interacts with a computer to access information and enter commands, and the computer implements the commands to control the process. Some examples of systems with computer-based interfaces are NASA satellite communications systems, airplane cockpits, nuclear power plant control rooms and telephone network management systems (Mitchell and Miller, 1986).

The possibilities for computer-based display content and format are infinite. One possible interface is a single or small number of CRT screens containing windows of task-specific, rather than hardware-specific, information. This type of window interface allows the operator to simultaneously access multiple views of the system and computer preprocessing alleviates the operator from tedious data search and low level data processing.

Little research has been conducted to provide general guidelines for producing a window-based interface. Card, Pavel and Farrell (1984) are engaged in one of the first studies that examine features of windowing systems and their possible applications. Some of this research is applicable to the domain of supervisory control system interfaces. Seven functional advantages of windows are defined by Card et al. (1984). These advantages are discussed in detail below.

1. Windows provide an increased amount of information. A display containing overlapping windows makes more information available or readily accessible than a two dimensional, full-screen display allows. Using 2 1/2 dimensional windows, more areas can be placed on the screen than actually fit. Partially overlapped windows allow the operator to use some information and know where the remaining information is located. Thus, windows can provide an increased amount of information on a single screen.

2. Windows provide an easy way to access multiple sources. In the area of supervisory control, one problem with conventional displays is that although each screen contains a wealth of information, very little of it is relevant at any one time for the specific task at hand. Using a windowing system, the operator can access useful portions of many displays simultaneously.

3. Windows provide a way to integrate multiple sources. Information integration is facilitated when multiple sources of data are displayed simultaneously. When the operator is forced to erase one piece of information to access related information, the previous data may be forgotten before it is integrated with currently displayed data. Human short term memory is a limited and transient data store, and data must be continuously rehearsed to be retained in short term memory (Loftus and Loftus, 1976). Miller's classic study indicates that humans can retain only about seven distinct items at one time (Miller, 1956). Woods (1984) notes that serial presentation of data where the human engages in across-display processing can degrade user information extraction as compared to parallel presentation where data is displayed simultaneously. If all relevant information is displayed at once, the operator is not forced to memorize several items while alternating among multiple display pages.

4. Multiple independent programs can run within separate windows. A supervisory controller may be responsible for several separate processes that must be monitored continu-

ously. Multiple dynamic windows indicating top level views of the system serve to facilitate system monitoring. On one screen, several windows each tracking a different process can update independently. Should a malfunction occur within one process, the top level window representing the process would indicate to the operator that the area required further investigation.

5. Windows can provide a reminding or helping function. Particularly useful in dynamic systems, a help window can define currently available or appropriate commands. A help window may aid a new operator in learning the syntax of system control commands. A reminding window can be used to provide the trained operator a history of past user inputs, or system events and alarms. This type of reminding window indicates to the operator what tasks have been completed and what needs to be accomplished.

6. Windows can be used to provide multiple context. Depending on the window, commands or keys can have different interpretations. For example, when the cursor is in one window a keystroke can be interpreted as typewritten input. In another window the same keystroke can move the cursor or select an item.

7. Windows can show multiple representations of the same process. One strategy for maintaining adequate system performance is to monitor successively detailed views of the system while eliminating correctly functioning areas from consideration and focusing on problematic areas. Dynamic multiprocessing windows allow the user to view both a top level and a detailed system representation at once. The operator can repair system malfunctions using a detailed system representation in one window while viewing the effect of the repairs on overall system performance in a window that reflects high level system functioning.

These seven functional advantages defined by Card et al. (1984) provide reasons why windows may be beneficial in a supervisory control environment. Before incorporating a window-based interface, however, potential drawbacks from inappropriately using windowing technology should be considered. Important issues to consider before applying windowing technology to specific applications are discussed in the next section.

Design Issues for Supervisory Control Windows

Precautions should be taken when incorporating a windowing system in supervisory control displays. Windowing technology enables the set of windows that is in use simultaneously to exceed the CRT's restricted screen space. If a task requires the operator to relate a large number of windows, the windows may overlap. If too many windows are in use, the majority of the user's time is spent overlaying one window on top of the others (Card et al., 1984). This situation is like working on a cluttered desk, where more time is spent shuffling through papers than in problem solving. One possible solution is to divide a task into a series of smaller tasks, each of which requires fewer windows.

Other important design issues are presented by Murray, Hakkinen and Mackraz (1984) for incorporating a windowing system in an office workstation. Proposed requirements are that the system should facilitate simple and quick moves between windows, and that the system should allow easy shifts from a window to a full screen view of a process. These are important considerations, since workers will be reluctant to use a system that is awkward and time consuming.

If a user has the capability to manipulate windows, it should be easy to move between windows, to call and erase windows and to relocate windows. An alternative to user controlled windows is to have the system aid in managing windows. A study which supports the concept of system controlled windows was performed by Bury, Davies and Darnell (1985). These authors compared performance with and without the use of a window system and found that subjects in the windowed environment took longer to complete the required tasks. When total time to perform the task was partitioned into screen-arrangement time and task-solving time, however, task-solving time was actually less in the windowed environment. This suggests that the benefit of problem solving using windows may be outweighed by the long time it takes to arrange windows in a usable format.

If the operator does not have the ability to call, erase and relocate windows, then it is important for windows to appear in an appropriate location when they are needed and to disappear when they are no longer needed. The interface designer must first determine the current operator task. Next, the information needed to carry out the task must be determined. Finally, windows containing task-oriented information should be arranged on the screen in a way that minimizes obstruction of required windows. If some windows must overlap, identifying features of the obstructed windows (e.g., window labels) should be visible.

A major difficulty with this approach is the determination of the current operator task, and thus, the determination of appropriate windows to display. There may be a number of tasks defined at the same priority level demanding operator attention. Alternatively, the operator may want to quickly finish a low priority task before starting a higher priority task, and may not want low priority windows to be erased or covered. One alternative to having windows presented automatically is to have the operator enter high level commands indicating what control task is currently of interest; then the set of windows required to perform the task could be presented. The system can erase or obscure windows when the operator indicates that a new control task is of greater interest. Alternatively, the operator may be permitted to erase one window or an entire set of windows.

Another important issue is determining the contents of windows. Computers are faster and more accurate than humans at low level information processing (e.g., elementary mathematical calculations), so this activity should be allocated to the computer. When the interface provides preprocessed data related to the current operator function, the human is free to commence higher level problem solving activities. A model relating operator control functions to information required to perform these functions is needed to determine the contents of windows.

In the chapters that follow, an experimental supervisory control environment that was used to evaluate a window-based interface is described. The window interface is controlled by a model of the operator's functions and related information needs. First, the system under study and a conventional user interface designed by NASA are described. Second, an operator function model of the system is summarized, and an interface based on the model is developed. Third, an experiment comparing the window-based interface to the conventional system interface is described. Finally, the results and their implications for display design are presented.

CHAPTER II

THE EXPERIMENTAL ENVIRONMENT: MULTISATELLITE OPERATIONS CONTROL CENTER (MSOCC)

A satellite communications system at NASA Goddard Space Flight Center was selected as a representative supervisory control system for this study. The supervisory control task of interest involves configuring and monitoring computer and communication equipment that supports command and control of NASA near-earth orbiting satellites. Before describing the details of this supervisory control task, an overview of the NASA satellite system is presented.

The main function of NASA-Goddard Space Flight Center (GSFC) is the design, launch and control of near-earth orbiting satellites. These satellites are unmanned missions that orbit the earth gathering data about weather, atmosphere, sun and earth. The satellites periodically transmit their scientific data or "telemetry" to an earth groundstation that in turn forwards the data to GSFC. Each contact with a spacecraft is called a "pass". During a pass, the spacecraft sends data down to GSFC, and mission controllers at GSFC send back commands as well as check the overall health and safety of the spacecraft.

NASA mission controllers manage and control each spacecraft. These spacecraft-specific controllers work in Mission Operations Rooms (MORs) that are staffed continually to receive telemetry, to monitor the spacecraft's status, and to issue new spacecraft commands. A configuration of communication lines and computers is needed to allow MORs to communicate with orbiting satellites. Although MORs are spacecraft-specific, most computer and communications hardware supporting real-time satellite commanding and data capture are shared resources. The Multisatellite Operations Control Center (MSOCC) is the system that coordinates the use of the shared computer and communications equipment. MSOCC schedules the equipment for satellite passes, configures and deconfigures MOR command and control equipment, and forwards telemetry and satellite health and safety data on to other NASA divisions.

GT-MSOCC

One subdivision of the overall MSOCC system is responsible for configuring shared computer and communications equipment, as well as monitoring the status of computer and data processing equipment currently in use. GT-MSOCC is a somewhat simplified simulation of this MSOCC subsystem. GT-MSOCC was developed at the Georgia Institute of Technology. It is a real-time, interactive, discrete event simulation of the MSOCC system that coordinates computer and communications equipment. The simulation is written in C and runs in BRL Unix (4.2 BSD) on a VAX 11/780 computer system.

GT-MSOCC is an automated system. Scheduling spacecraft contacts, assigning equipment, configuring communications equipment, and deconfiguring communications equipment are all performed automatically. The GT-MSOCC operator is primarily a monitor, retained in the system to detect and compensate for system problems. The GT-MSOCC operator intervenes when 1) the automated system is unable to configure or deconfigure a scheduled pass, 2) there are problems with equipment currently being used to support a spacecraft pass, or 3) an unscheduled spacecraft contact is requested. In order to detect and compensate for these system problems, the operator accesses information about equipment use, availability and performance. The user interface provides the operator with information needed to perform the GT-MSOCC operator control functions.

Conventional Operator Interface

The interface for the MSOCC supervisory control system is a set of full page screens containing data that reflect physical aspects of system functioning. The interface provides the operator access to all measurable system information. This approach tries to safeguard against omitting some piece of data that might be needed to detect or compensate for a critical system state. Information is contained on over one hundred displays, and the operator has several monitors on which to view display pages. The interface is designed so that each display page is assigned to a certain monitor and can only be accessed on that monitor. If information is required from two display pages assigned to the same monitor, the operator must erase one display to access another.

This method of data presentation assumes that the human is able to select, process and integrate information from multiple sources quickly and accurately. This type of low level information processing, however, may be difficult for humans due to short term memory limitations. Kantowitz and Sorkin (1983) state that human memory is a limited resource. Only a few independent items or chunks, i.e., numbers or words, can be stored simultaneously in working memory. To retain items in working memory, they must be rehearsed. Kantowitz and Sorkin cite studies indicating that items are forgotten when rehearsal is eliminated by intervening events or when interference among different items within the same category occurs. When an operator works with an interface like that described above, switching among different display pages and searching for new items may eliminate rehearsal of items already being stored in working memory. Since operators must retain similar data items simultaneously (e.g., names of system components or numbers of comparable lengths), interference among items may occur.

Wickens (1983) asserts that the normally defined limit of working memory capacity is probably an overestimate of memory limits within the context of human-machine interaction. As noted by Wickens, Moray (1980) defines a "running memory" task as one where a continual sequence of items is presented, and the operator is not required to remember the entire string. Moray found that operators who are engaged in a running memory task, typical in operation of a dynamic process control system, have a memory span of considerably less than 7 ± 2 chunks.

GT-MSOCC displays are designed from an interface proposed by NASA for the MSOCC system (NASA, 1983). The GT-MSOCC user workstation (shown in Figure 2.1) consists of three CRT's and a single keyboard on which the operator requests information and executes commands. The center monitor displays a dedicated equipment configuration and status page; the left and right monitors support over one hundred full page display screens that reflect hardware usage and performance.

Figure 2.2 shows the configuration and status page that is displayed on the center monitor of the GT-MSOCC operator workstation.

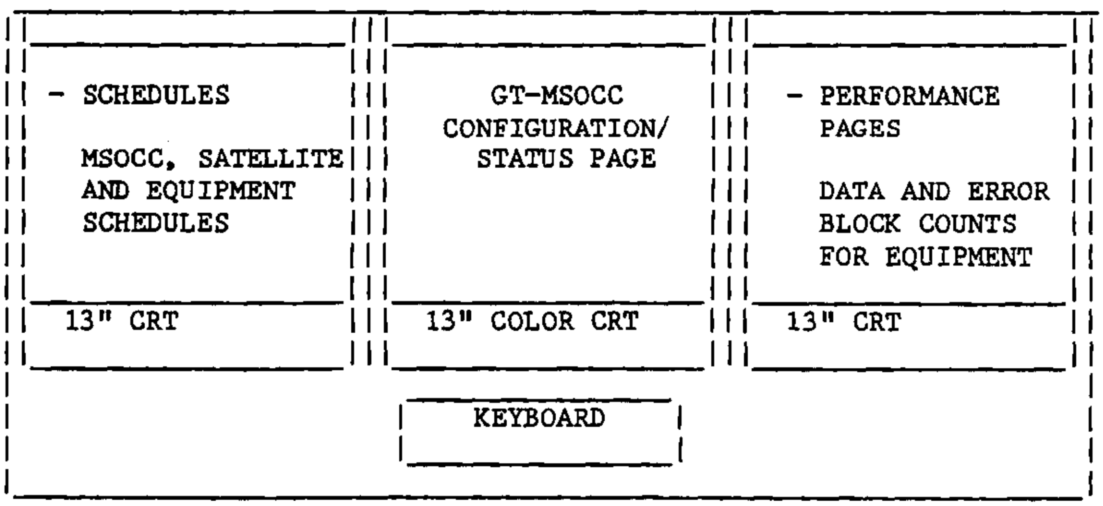


Figure 2.1 The Conventional GT-MSOCC Three Monitor Operator Workstation.

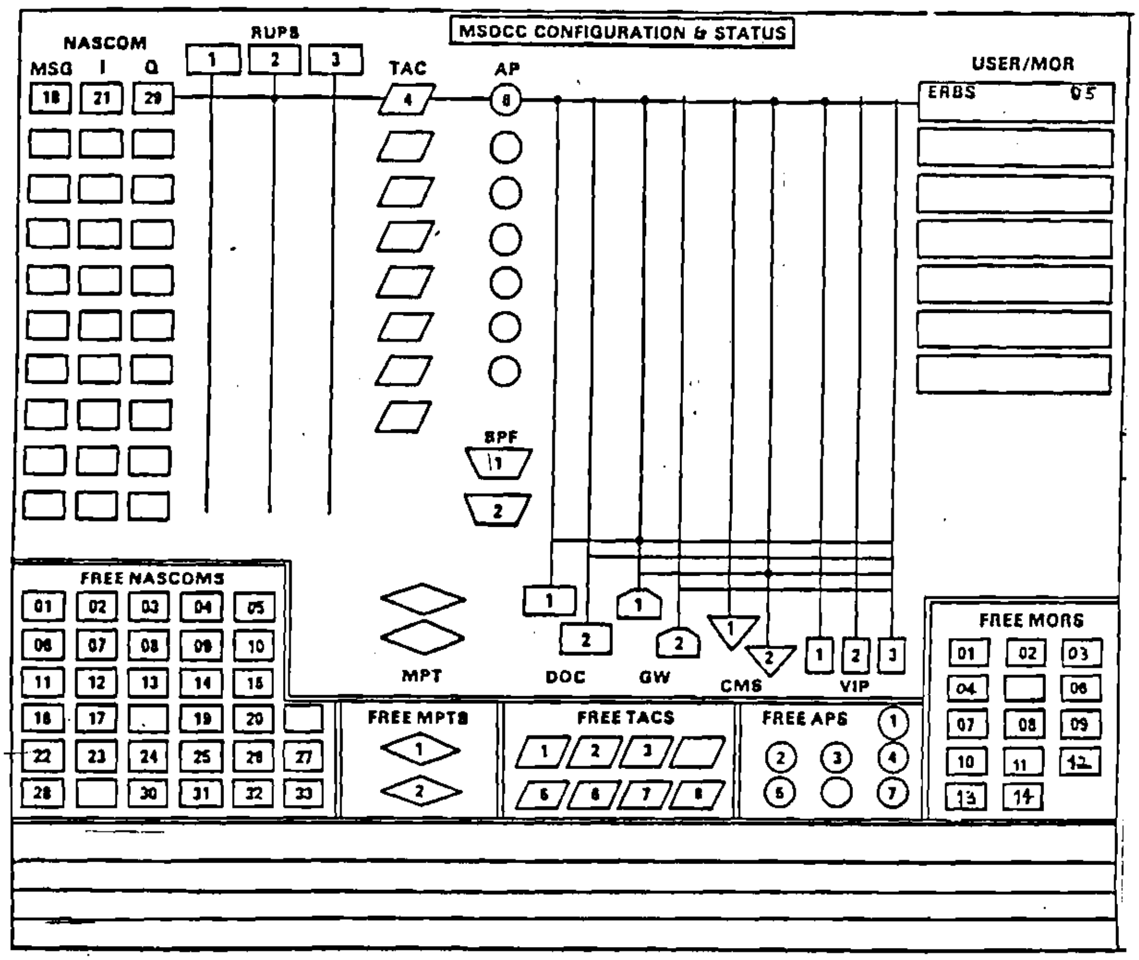


Figure 2.2 Configuration and Status Display Page.

It is a dedicated graphics page that displays the hardware status of all equipment under GT-MSOCC operator control. In addition, the equipment configurations supporting each current spacecraft contact are shown on this page. The upper portion of the screen shows equipment configurations and status of currently transmitting satellites

(Figure 2.2). The lower portion of the screen shows the status of equipment that is not in use. Both parts of the screens are color coded, and each piece of equipment is represented with an icon. A red icon indicates that a component has failed, a blue icons represents idle equipment, and hardware components that are in use and are operational are coded in green. The sample GT-MSOCC configuration and status page in Figure 2.2 shows that ERBE is the only spacecraft currently being supported. Reading from left to right the equipment supporting this mission consists of NAS 18, 21, and 29, RUP2, TAC4, AP6, DOC1, GW1, CMS2, VIP1 and MOR5.

The operator uses the configuration and status page for a number of purposes. The configuration and status page is used to detect a component that has had a hardware failure and is in use supporting a spacecraft. Operators may need to know what equipment is supporting a mission; this information is useful for identifying the cause of software problems. The configuration and status display also indicates what equipment is idle and operational, which is necessary information for identifying replacement components. Finally, information about what equipment is failed and offline for maintenance is valuable for strategic planning.

In addition to the center GT-MSOCC configure and status page, the GT-MSOCC operator has two other display monitors available. The operator can access many pages of information on both the right and left monitors. The right screen provides updating data block and error block counts for each current mission (Figure 2.3). The left screen displays schedule information (Figure 2.4). First the performance pages on the right monitor will be described. Then in subsequent paragraphs the left monitor's schedule display pages are presented.

The operator uses the right monitor to ensure the integrity of data currently being transmitted through the GT-MSOCC system. Components that are operational and are represented by green blocks on the status and configuration page may still degrade the data in some way. The displays on the right monitor are used to detect a decreased rate of data transmission or an increased rate of error block propagation through the system. Operators infer the rate of data transmission by observing successive updates of total block counts received at various pieces of equipment. To detect data transmission problems, operators generally determine the rate of data flow as information completes its path through the system. If the operator detects a low data transmission rate at a terminal point of the system, he or she traces back along the equipment configuration to determine the source of the problem. There are up to five system terminal points that must be monitored; data transmission at these components is displayed on two different pages on the right monitor.

Figure 2.3 shows data transmission at the MOR, one of the system terminal points. On this page three updates are displayed simultaneously. For the satellite DSEI the oldest block count (BC) was 539 blocks, then 1043 appeared and the most recent update was 1535. The oldest block count is overwritten by the new update, so the next

			TELEMETRY STATUS/QUALITY				083/00:24:43		
NAME	SITE	TIME DOWN	TYPE	DEST	MSID	TBR	BC	TBP	FLAGS
DSEI	MAD	083/00:28:00	RT	MAD	0077	01228	001535	00002	00013
DSEI	MAD	083/00:28:00	RT	MAD	0077	00431	000539	00000	00004
DSEI	MAD	083/00:28:00	RT	MAD	0077	00834	001043	00001	00008
PM	EAST	083/00:25:00	RT	EAST	0069	10276	012845	00039	00195
PM	EAST	083/00:25:00	RT	EAST	0069	10789	013486	00042	00214
PM	EAST	083/00:25:00	RT	EAST	0069	11435	014293	00047	00236
AE-D	ORR	083/00:30:00	PB	ORR	0079	00415	000519	00000	00004

Figure 2.3 Display Page of Data and Error Block Counts at the MOR Terminal Point.

update will appear in the center row for DSEI and on the top row for PM.

When equipment is diagnosed as faulty, the operator tries to replace it with a another component. The left monitor gives alphanumeric and graphic schedules that indicate the scheduled use and future availability of equipment. Schedule information provided on the left screen includes an overall GT-MSOCC spacecraft pass schedule (Figure 2.4), schedules for each satellite (Figure 2.5) and schedules for each of the individual pieces of equipment (Figure 2.6). Graphics schedules for classes of equipment are also provided (Figure 2.7). The decision as to which schedule to use is dictated by the type of equipment of interest, the purpose for which information is required, and the personal preference of the operator. The same information is usually available on a number of different

110/18:25:47
DOC1 ONL

DAY 116-117 MSOCC SUPPORT SCHEDULE 116/00:00 TO 117/23:59

STRT	AOS	LOS	END	USER	ORBIT	STA	TYPE	LINES	EQUIPMENT
1910	1930	2010	2017	ISEE-1	03490	GWM	RT	03 08 14	RUP1 TAC1 AP3 DOC1 MOR5
1940	2000	2020	2025	ISEE-3	01578	MAD	RT	18 27	TAC4 AP2 DOC1 MOR3
1940			2100	CSC-BASL			SW		AP4 SW1
2000			2130	CSC-SPIF			SW		SPF2 MOR8
2050	2110	2157	2202	DE-1	01831	AGO	RT	01 05 14	RUP3 TAC1 AP5 GW2 DOC1 MOR7
2110	2130	2300	2305	ISEE-3	01578	EAST	RT	19 21	RUP2 TAC2 AP6 DOC1 MOR3
2205	2215	2235	2237	ATS-1	00290	WEST	RT	04 08 12	TAC3 AP1 DOC1 MOR2
2230	2240	2300	2305	ATS-5	00184	AGO	RT	22 28 32	RUP2 TAC2 AP2 DOC1 MOR1
2300			0000	CSC-GW			TST		AP1 AP7 GW1 GW2 DOC1 MOR8 LCR1
2310	2320	2340	2342	ATS-3	00225	GDS	RT	23 33	TAC2 AP2 DOC1 MOR1
2312	2332	0117	0122	DE-1	01832	ORR	RT	01 11	RUP3 TAC5 AP5 DOC1 MOR7
2330			0020	IMP-8			PB		RUP2 TAC8 AP6 DOC1 MOR4
0000			0100	MNT-DEC			PM		TAC1
0020	0030	0120	0122	ISEE-1	03491	HAW	RT	05 09 17	TAC7 AP3 DOC1 MOR5
0040			0150	COBE			SIM		RUP1 TAC1 TAC3 TAC2 AP4 DOC1 MR10
0100			0400	MNT-DEC			RM		AP6
0110			0230	ERBS			OFL		AP7 DOC2 MOR4

Figure 2.4 A Sample MSOCC Schedule Page.

065/18:27:06
DOC1 ONL
DAY 067-070 ISEE-3 SUPPORT SCHEDULE 066/00:00 TO 072/23:59

STRT	AOS	LOS	END	DAY	ORBIT	STA	TYP	LINES	EQUIPMENT
1420	1430	1455	1457	067	01541	ORR	RT	20 24 28	RUP2 TAC6 AP2 DOC1 MOR3
1805	1815	0450	0452			MAD	RT	18 22 24	RUP2 TAC4 AP2 DOC1 MOR3
0130			0240	068			OFL		AP4 DOC2 MOR3
0504	0514	0728	0732		01542	GDS	RT	03 14	RUP1 TAC2 AP2 DOC1 MOR3
0732			0835				PB		RUP1 TAC2 AP2 DOC1 MOR3
0930	0940	1000	1005		01542	ORR	RT	20 22 24	RUP2 TAC2 AP6 DOC1 MOR3
1005			1045				PB		RUP2 TAC2 AP6 DOC1 MOR3
1110	1120	1147	1152		01542	ORR	RT	18 20 30	RUP2 TAC8 AP6 DOC1 MOR9
1308	1318	1348	1352		01542	ORR	RT	18 22 25	TAC4 AP4 DOC1 MOR3
1430			1530				OFL		AP5 DOC2 MOR3
1639	1649	1750	1754		01543	ORR	RT	02 03 04	RUP3 TAC1 AP5 GW1 CMS2 DOC1 MOR3
1807	1817	0454	0458		01543	EAST	RT	20 25 30	RUP2 TAC2 AP2 DOC1 MOR3
0510	0520	0749	0753	069	01544	GDS	RT	20 21 30	RUP2 TAC2 AP6 DOC1 MOR3
0844	0854	1017	1021		01544	ULA	RT	01 07 13	RUP1 TAC7 AP2 DOC1 MOR3
1420	1430	1605	1607		01544	ORR	RT	04 07	RUP3 TAC6 AP4 DOC1 MOR3
1808	1818	1902	1904		01545	MAD	RT	18 19 33	RUP2 TAC2 AP4 DOC1 MOR3
0000	0010	0123	0125	070	01545	ORR	RT	18 25 30	RUP2 TAC6 AP2 DOC1 MOR3
0125			0210				PB		RUP2 TAC6 AP2 DOC1 MOR3

Figure 2.5 A Support Schedule for the ISEE-3 Satellite.

		365/14:05:29
		DOC1 ONL
DAY 001-003	AP3 SUPPORT SCHEDULE	001/00:00 TO 007/23:59

STRT	AOS	LOS	END	USER	ORBIT	STA	TYP	LINES	EQUIPMENT
1910	1920	2015	2017	ISEE-1	04360	GWM	RT	02 06 11	RUP1 TAC1 AP3 DOC1 MOR7
2020			2105	CSC-GW			SW		AP3 SW1
2110	2120	2259	2301	ISEE-3	01742	EAST	RT	18 20 33	RUP2 TAC5 AP3 DOC1 MOR3
2301			2350	ISEE-3			PB		RUP2 TAC5 AP3 DOC1 MOR3
0110	0120	0312	0314	IMP-8	00560	WEST	RT	01 02 03	RUP1 TAC1 AP3 DOC1 MOR1
0400			0500	MNT-POC			PM		RUP1 TAC5 AP3 MOR9
0510			0603	ATS-1	00322	WEST	RT	04 11 12	TAC7 AP3 DOC1 MOR1
0625			0715	COBE			SIM		RUP2 TAC4 TAC6 TAC3 AP3 DOC1 MR10
0715	0725	0836	0838	DE-1	01890	ORR	RT	15 16 17	RUP3 TAC5 AP3 DOC1 MOR7
0850			0940	CSC-GW			TST		AP3 AP7 GW1 GW2 DOC1 MOR8 LCR1
1320			1410	IMP-8			PB		RUP1 TAC7 AP3 DOC1 MOR1
1410			1610	CSC-BASL			SW		AP3 MR10
1730			1830	MNT-OPS			PM		AP3 MOR9
1850	1900	2030	2032	ISEE-1	04361	GDS	RT	03 16	TAC7 AP3 DOC1 MOR5
2020			0330	MNT-DEC			RM		AP3
0350			0440	ATS-3			OFL		AP3 DOC2 MOR4
0500	0510	0710	0712	ERBS	00006	WEST	RT		RUP1 TAC3 AP3 GW1 CMS1 DOC1

Figure 2.5 A Support Schedule for the ISEE-3 Satellite.

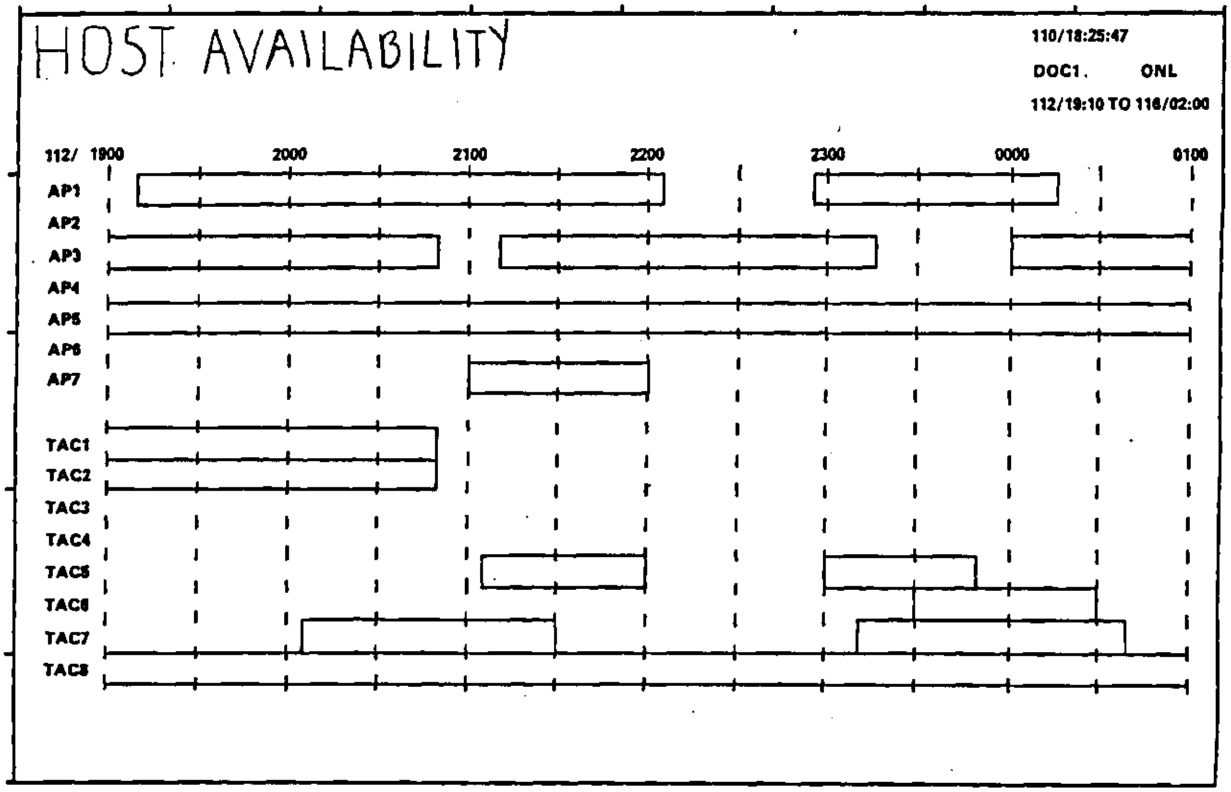


Figure 2.7 A Sample Graphic Schedule Display Page.

schedules.

To retrieve any disply page of information the operator issues the command "DISPLAY" followed by the name of the desired display. For example, the command "DISPLAY MSOCC SCHED" calls the overall system schedule. A complete list of all information retrieval commands for the conventional interface is provided at the end of Appendix B.

Designers of a conventional interface such as the one described in this chapter are familiar with physical aspects of the system, but may be unfamiliar with the functions of operators who will eventually use the displays. All information is displayed whether it is needed or not, and often redundant information is provided on multiple display pages. Since displays are not developed based on knowledge about specific operator functions, displays are created that can be used by different types of system operators who have very different responsibilities. However, these displays make rapid information retrieval difficult, especially in time-critical situations. Using an interface like the conventional GT-MSOCC interface described above, the supervisory controller must integrate many pieces of information from a number of display screens to accomplish the current operator task.

An alternate approach to designing a supervisory control interface is introduced in the next chapter. The proposed interface is based on an operator function model that determines GT-MSOCC operator control functions and related information needed to perform these functions. The interface controlled by this model presents the operator with succinct information related to the current operator task. Information is aggregated at a level that facilitates rapid decision making and is presented on a single screen when it is needed.

CHAPTER III

A MODEL-BASED WINDOW USER INTERFACE

GT-MSOCC was developed to resolve a number of human-computer interface design issues for supporting control systems. One of the design issues of interest was a comparison of windows versus full-screen displays, which is the topic of this thesis. The previous chapter defined the GT-MSOCC system and described the conventional, full screen interface for the system. A new interface is described in this chapter. The proposed interface is based on a model of the GT-MSOCC operator's information needs (Mitchell, 1985). From the model, a two monitor workstation was designed. One screen supports dynamic icons; the other supports a windowing environment. Although the implementation of the windowing environment is the focus of this chapter, the dynamic icons that were implemented as part of the interface are also discussed.

Table 3.1 gives an overview of the proposed GT-MSOCC interface. The GT-MSOCC interface is a model-driven system comprised of dynamic icons and an alphanumeric 2 1/2 dimensional window environment. Various operator functions as defined by the model are accomplished using information contained either in the icons or windows. The relationship between the operator function model, the window

Table 3.1
An Overview of the GT-MSOCC Interface
Based on an Operator Function Model

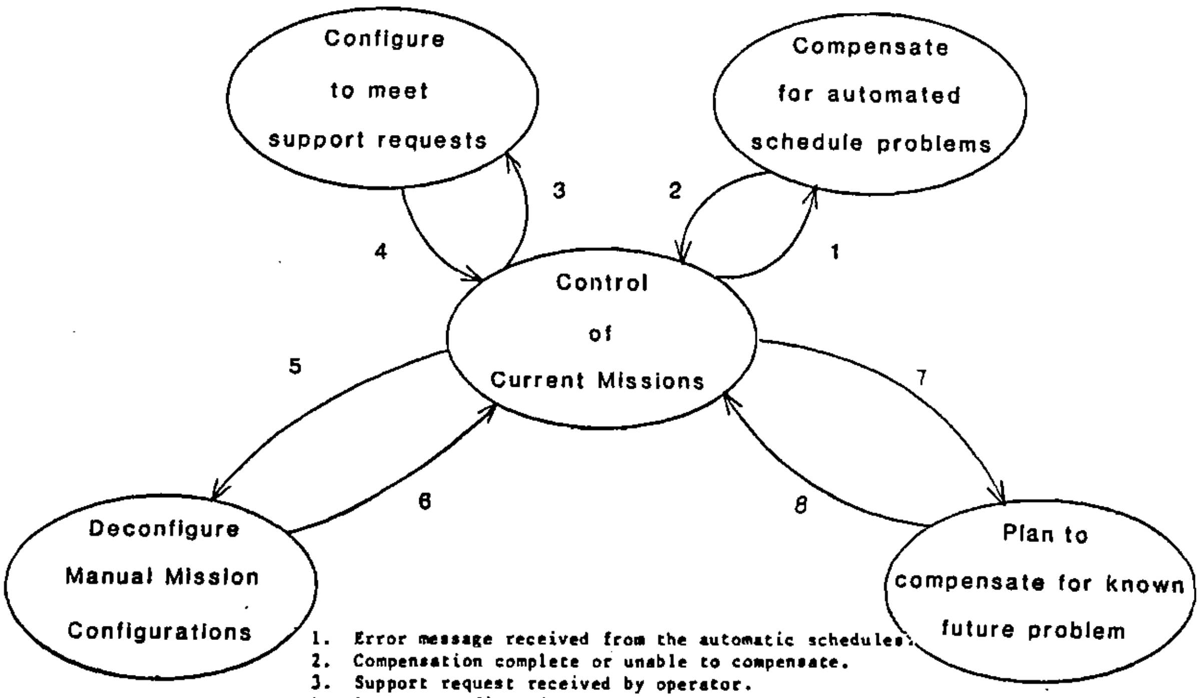
GT-MSOCC MODEL-BASED INTERFACE	
OPERATOR CONTROL FUNCTIONS SUPPORTED BY DYNAMIC ICONS	OPERATOR CONTROL FUNCTIONS SUPPORTED BY A WINDOW ENVIRONMENT
- Monitoring - Fault Detection	- Fault Compensation - Compensation for Schedule Conflicts - Configuration to Meet Support Requests - Strategic Planning

environment, and specific commands and sets of windows is the focus of this chapter.

The GT-MSOCC Operator Function Model

This chapter illustrates the design of an interface that uses windows in order to enhance supervisory control performance in an automated system. As mentioned earlier, the interface was developed using an operator function model of GT-MSOCC constructed by Mitchell (1985). The model defines major GT-MSOCC operator functions, subfunctions comprising these main functions, and information and commands required to complete the operator tasks.

Figure 3.1 depicts a top level view of the model and introduces the five major GT-MSOCC operator control functions. The model represents operator states, or tasks, as nodes in the network. The



1. Error message received from the automatic scheduler.
2. Compensation complete or unable to compensate.
3. Support request received by operator.
4. Request configured or unable to meet request.
5. Message received that a manually configured mission is completed.
6. Deconfiguration completed.
7. Operator summons schedule and/or mission template pages when no other triggering event takes place.

Figure 3.1 Major GT-MSOCC Supervisory Control Functions.

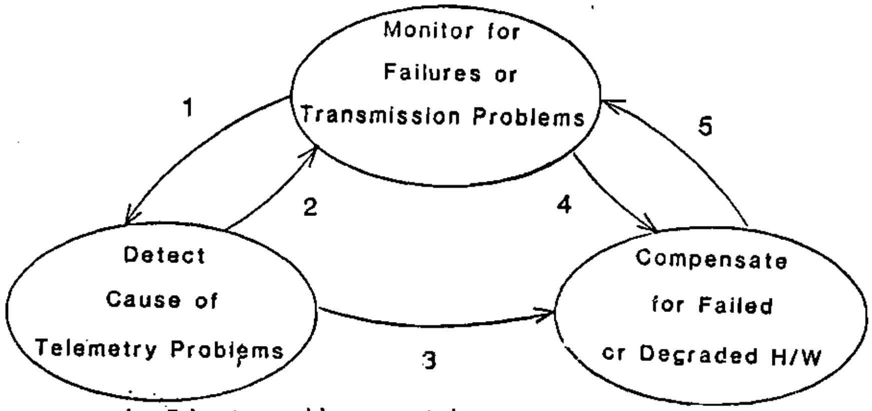
arcs represent system events that cause the operator to complete or interrupt one task and begin another. The first major operator function is control of current missions. This function is depicted in the center node. The GT-MSOCC operator ensures that all equipment currently in use is functioning and that data integrity is being preserved. Control of current missions is maintained except when system malfunctions divert operator attention. A message usually notifies the operator when a system problem requires human intervention.

For example, an error message from the automated scheduler alerts the operator when equipment scheduled for a contact cannot be automatically configured. The operator then attempts to compensate for the automated scheduling problem by identifying replacements for scheduled equipment that has become unavailable. Compensation for automated schedule problems is the second major operator function.

An operator may also receive a message requesting unscheduled support for an emergency contact, a system demonstration or software testing. If sufficient hardware is available, the operator configures equipment to meet the support request. This is the third major operator function. Once an equipment string has been manually configured or altered it must be deconfigured by the operator upon completion of data transmission. Manual deconfiguration is the fourth control function.

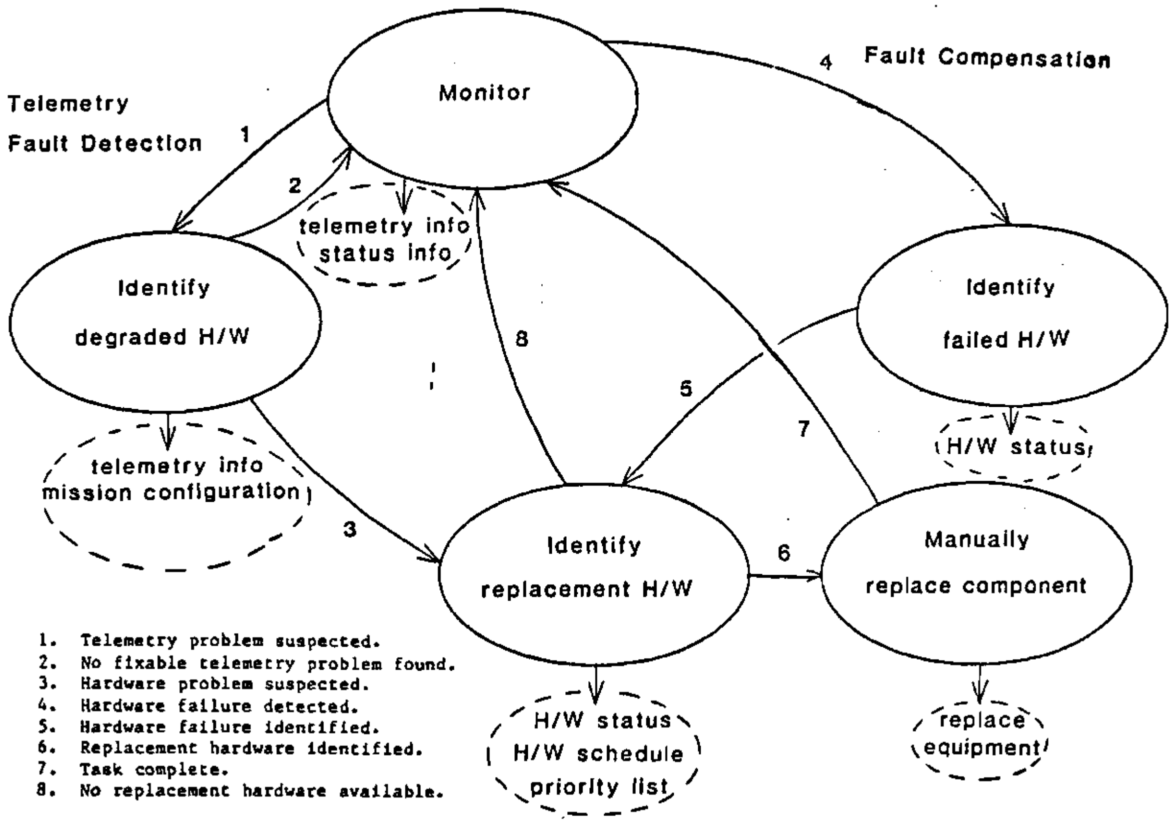
The final operator control function is planning for potential future problems. For example, a failed hardware component that is scheduled for use in the near future is likely to cause an automated schedule problem. When the system appears to be stable, operators may engage in strategic planning for such events.

Figures 3.2a to 3.2f define the subfunctions and activities that comprise the five major GT-MSOCC operator functions defined above



1. Telemetry problem suspected.
2. No fixable telemetry problem found.
3. H/W component identified as a cause of telemetry problem.
4. H/W failure message received by the operator.
5. H/W failure fixed or compensation deemed impossible.

Figure 3.2a The Subfunctions Comprising the Control of Current Missions Function.



1. Telemetry problem suspected.
2. No fixable telemetry problem found.
3. Hardware problem suspected.
4. Hardware failure detected.
5. Hardware failure identified.
6. Replacement hardware identified.
7. Task complete.
8. No replacement hardware available.

Figure 3.2b Specific Tasks Comprising the Subfunctions for the Control of Current Missions.

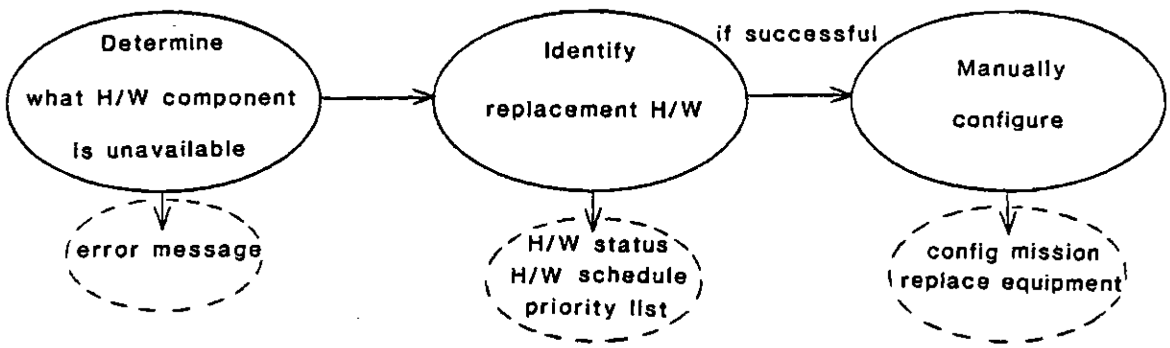


Figure 3.2c Subfunctions Comprising the Function: Compensate for Automated Schedule Problems.

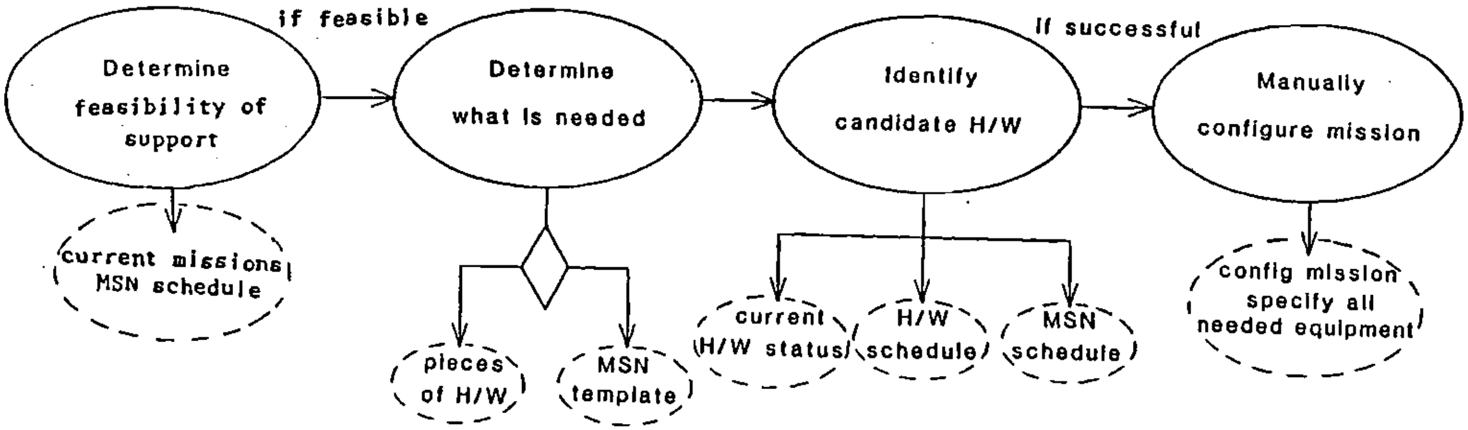


Figure 3.2d Subfunctions Comprising the Function: Configure to Meet Support Requests.

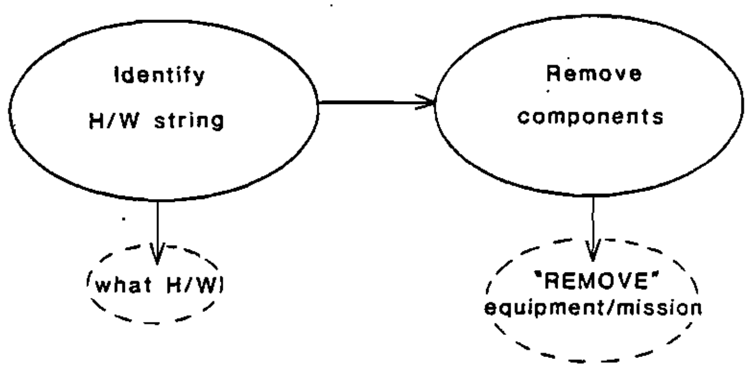


Figure 3.2e Subfunctions Comprising the Function: Deconfigure Manual Mission Configurations.

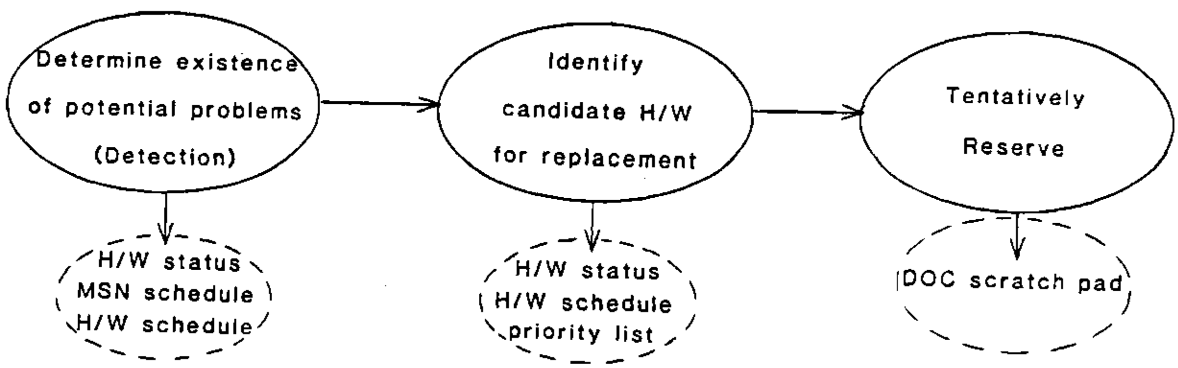


Figure 3.2f Subfunctions Comprising the Function: Plan to Compensate for Known Future Problems.

(taken from Mitchell, 1985). These figures are discussed in more detail below in the context of a proposed operator workstation.

The Proposed GT-MSOCC Workstation

Using the GT-MSOCC operator function model briefly explained above, a two CRT GT-MSOCC workstation is proposed. Each CRT provides information to support functions that were defined by the operator function model. The right screen supports the operator functions of monitoring and fault detection. The left screen supports fault compensation as well as several other operator functions.

The Right Monitor

The first operator function is the control of current missions. This activity is comprised of three subfunctions: monitoring, fault detection, and fault compensation (Figure 3.2a). The primary GT-MSOCC operator responsibility is to ensure that data transmitted from spacecraft are captured and that data quality is preserved. Thus, the operator continually monitors current missions and, when a problem is suspected, searches the equipment configuration to locate the probable cause. Since the operator spends a significant amount of time monitoring the system for failures and identifying faulty equipment, the right screen of the proposed GT-MSOCC operator workstation is dedicated to providing information that enables the operator to carry out these tasks easily. The primary feature of this display is a qualitative, dynamic icon that integrates important system features into a high level error detection device.

The use of this qualitative, high level, dynamic representation is consistent with the design principles currently being proposed by both the supervisory control (Goodstein, 1984; Mitchell and Miller, 1986; Rasmussen, 1984; Wickens, 1983) and the human-computer interaction research communities (Foley and Van Dam, 1982). Rasmussen and his colleagues are the foremost proponents of these ideas for effective supervisory control.

Since the operator is limited to considering only a few data items at a time due to inherent human information processing limitations, Rasmussen (1981) asserts that information for system monitoring should initially be provided to the operator at a level of detail that reflects high level concepts. To monitor overall system performance, Rasmussen (1984) suggests that information be provided such that the operator can immediately detect deviations from the target (or expected) state.

In a more general way, the human-computer interaction research community concurs. Foley and Van Dam (1982) state that by using pictorial representations, "we are largely liberated from the tedium and frustration of looking for patterns and trends by scanning many pages of linear text on line printer listings or alphanumeric terminals" (p. 5). Dynamically varying graphical representations may be an even better means of communicating information than static pictures (Foley and Van Dam, 1982). This observation suggests that the operator can process information needed to monitor the system most quickly by using a dynamic, pictorial representation of the system. This representation should reflects high level system features by graphically indicating differences between actual and expected states.

Goodstein (1984) notes one such dynamic, iconic interface to a nuclear reactor where normal system state is represented as a polygon and deviations from normal are represented as indentations and bulges on the polygon. Twelve critical parameters are labelled around the polygon, and certain patterns of distortion map to specific system malfunctions. An evaluation of this icon resulted in good operator performance in terms of detecting and diagnosing deviations from normal.

In the interface to GT-MSOCC, the right screen contains dynamic graphics that integrate system features so that accurate monitoring and rapid fault detection is facilitated. The right screen has two purposes. It continually provides high level information about currently supported missions, and when requested provides more detailed information about hardware status and data quality and flow rate at individual components comprising a satellite's equipment string.

Dynamic Icons for System Monitoring. Monitoring current missions is a frequent operator activity, since the operator continuously monitors the system unless a problem with current data transmission is suspected or another major GT-MSOCC control function preempts system monitoring. Since monitoring current missions is the prevalent operator task, half of the right monitor is dedicated to iconic representations of the most significant features of each current satellite contact. A dynamic spigot icon was chosen to provide a qualitative representation of data block flow rate, error block count, and total accumulated data.

Figure 3.3 depicts the spigot icon for the satellite ERBE. Liquid flowing into the icon's bucket represents information flow rate as data reaches one of the terminal points in the equipment string supporting a spacecraft contact. Red dots collecting at the bottom of the bucket represent the amount of bad data (error blocks) that have been transmitted through the system.

The spigot icon is dynamic. At any given time the spigot icon for each satellite currently being supported represents the worst

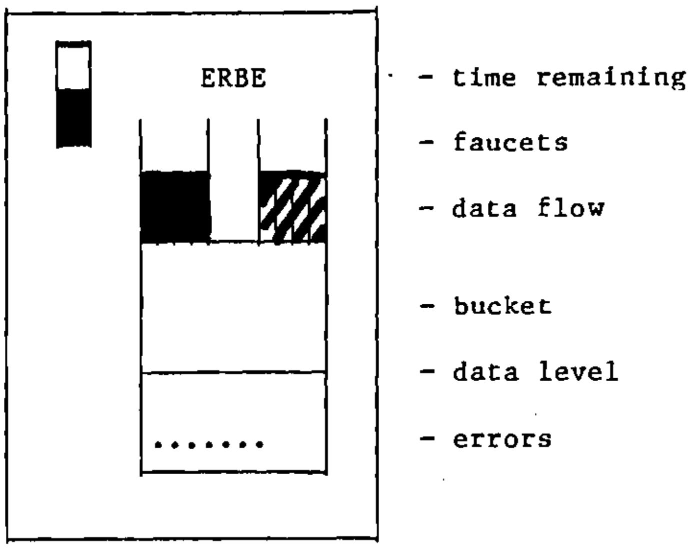


Figure 3.3 A Spigot Icon for the ERBE Satellite. A spigot icon is provided for each current spacecraft contact.

points of system functioning, i.e., those most likely to require operator attention. The spigot icon qualitatively represents the smallest flow rate of any system terminal point. In a similar way, the red error blocks at the bottom of the bucket depict the maximum number, proportionally, of detected errors at any terminal point. A terminal point is defined as any place in the equipment configuration where data are not transmitted to another piece of equipment that is under GT-MSOCC operator control. Possible terminal point include the following equipment: an MOR, RUP, VIP, GW and CMS.

The spigot icon depicts data transmission quite differently than the conventional displays that were presented in the previous chapter. The primary GT-MSOCC operator task is to ensure that data are flowing to all system components and that both the flow rate and data quality are acceptable. The conventional interface presents the total number of data blocks processed at each piece of equipment and the number of error blocks each component has produced. These displays are indirect representations of the quantities of interest, since the operator monitors rate of data transmission, not the total amount of data accumulated.

Unlike conventional displays, the spigot icon integrates the separate status of individual terminal points into one error detection device. Its purpose is to alert the operator to potential problems with data transmission.

Dynamic Icons for Fault Detection. A spigot icon displaying decreased flow rate or increased error counts alerts the GT-MSOCC operator to potential problems with a satellite's computer and communication equipment. To locate the source of the problem, the operator initiates fault detection by requesting additional information about the equipment supporting a specific mission.

On the left half of the screen the operator can request a detailed, dynamic picture of the individual equipment string supporting the mission. This detailed icon provides the information required for fault detection. There are two ways that equipment can fail: either hardware or software may be faulty. A different icon exists to reflect each type of failures. The first type of failure is a hardware failure, in which a component is completely inoperable and requires offline maintenance. An equipment status icon provides a detailed representation of an equipment configuration that color codes a failed piece of equipment in red and operational components in green. Figure 3.4 provides an example.

A software problem is the second type of failure that can occur. A software problem is more subtle than a hardware failure and is not easy for the automatic system to detect since the component is functioning but is in some way degrading the data. There are two major classes of symptoms for software problems. Either data are not flowing at a fast enough rate or the quality of data blocks is being compromised. The data flow icon, shown in Figure 3.5, represents data transmission and error block counts at each piece of equipment

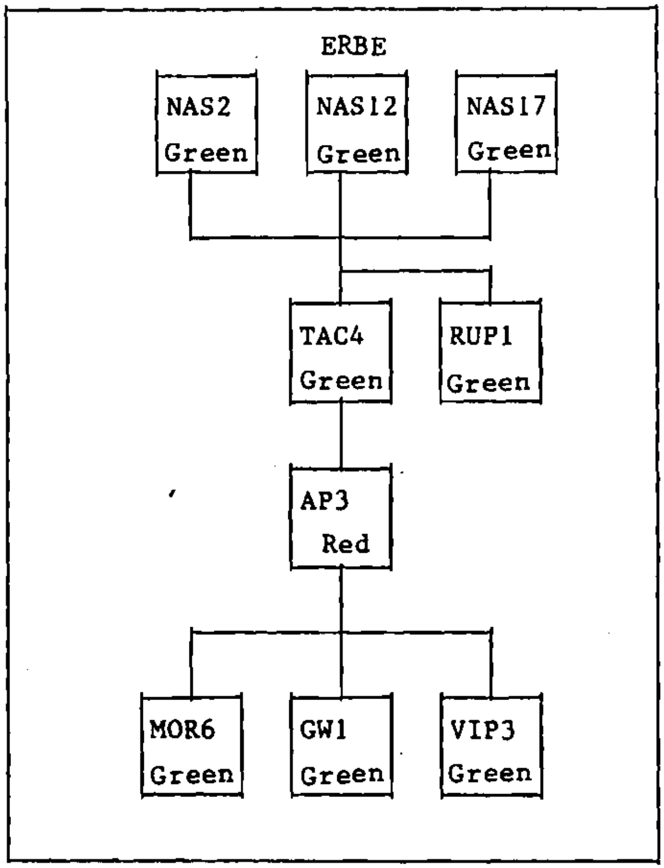


Figure 3.4 A Status Icon for the ERBE Satellite. The component AP3 has a hardware failure and is is coded in red.

supporting a spacecraft contact.

If a problem is due to a hardware failure, the equipment status icon is more useful. The equipment status icon is drawn on the screen more quickly than the data flow icon and makes it immediately obvious to the operator which component is faulty (the component is

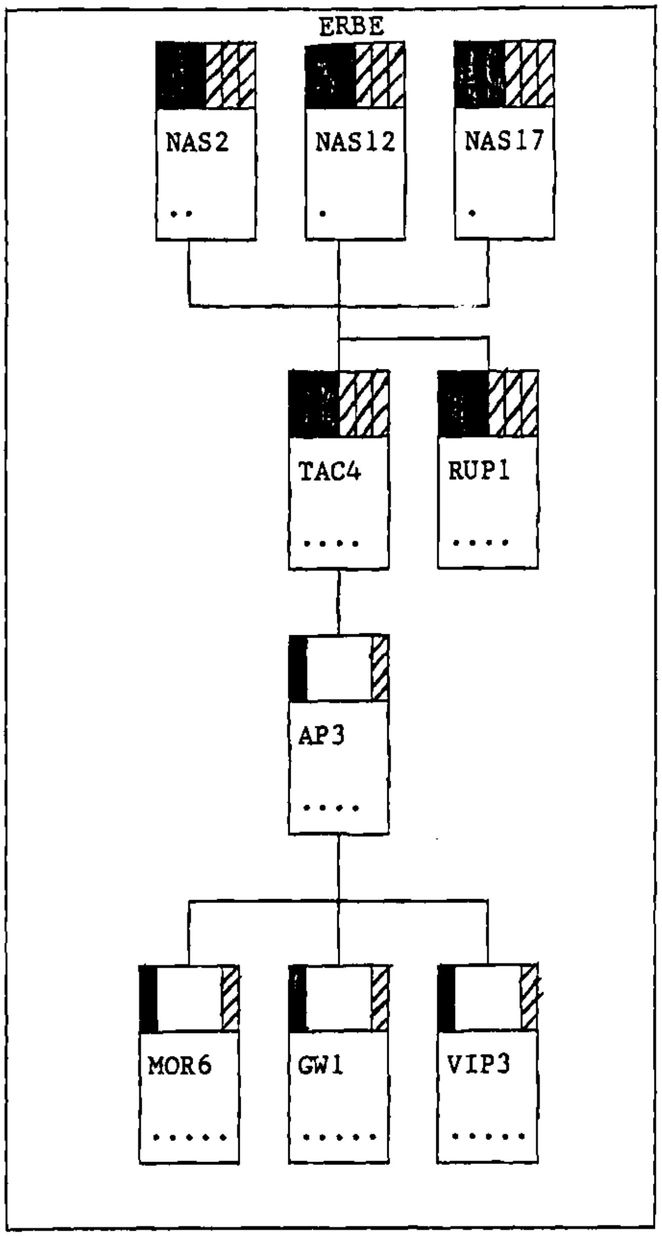


Figure 3.5 A Flow Icon for the ERBE Satellite. AP3 is causing decreased flow through the system.

red). The equipment status icon also indicates that the system problem is due to a hardware failure. If a software failure is causing the problem, the faulty component can be detected by means of the data flow icon.

The operator can request either of these two detailed icons directly or can issue a command that chooses between them. For example to view a more detailed representation of the ERBE satellite, the operator could issue the command "DISPLAY ERBE FLOW" to access the data flow icon, "DISPLAY ERBE STATUS" to access the equipment status icon, or "DISPLAY MORE ERBE" to have the system choose either the flow or status icon. The "more" command has some intelligence and reveals the representation that is most valuable for fault detection. The system provides the equipment status icon only if one or more components supporting the mission of interest has had a hardware failure. Otherwise, this command produces the data flow icon on which the operator can detect the more subtle software failures.

Figure 3.6 depicts a sample of the right graphics monitor. Three missions are currently being supported, and the operator has requested a detailed view of the satellite AE-QL. On the right monitor, spigot icons alert the operator to potential system malfunctions. A detailed representation of an equipment configuration provides the mechanism for fault identification.

Once the operator identifies a component as the probable cause of data flow problems using the right monitor, he or she commences fault compensation on the left monitor. This is the third and final

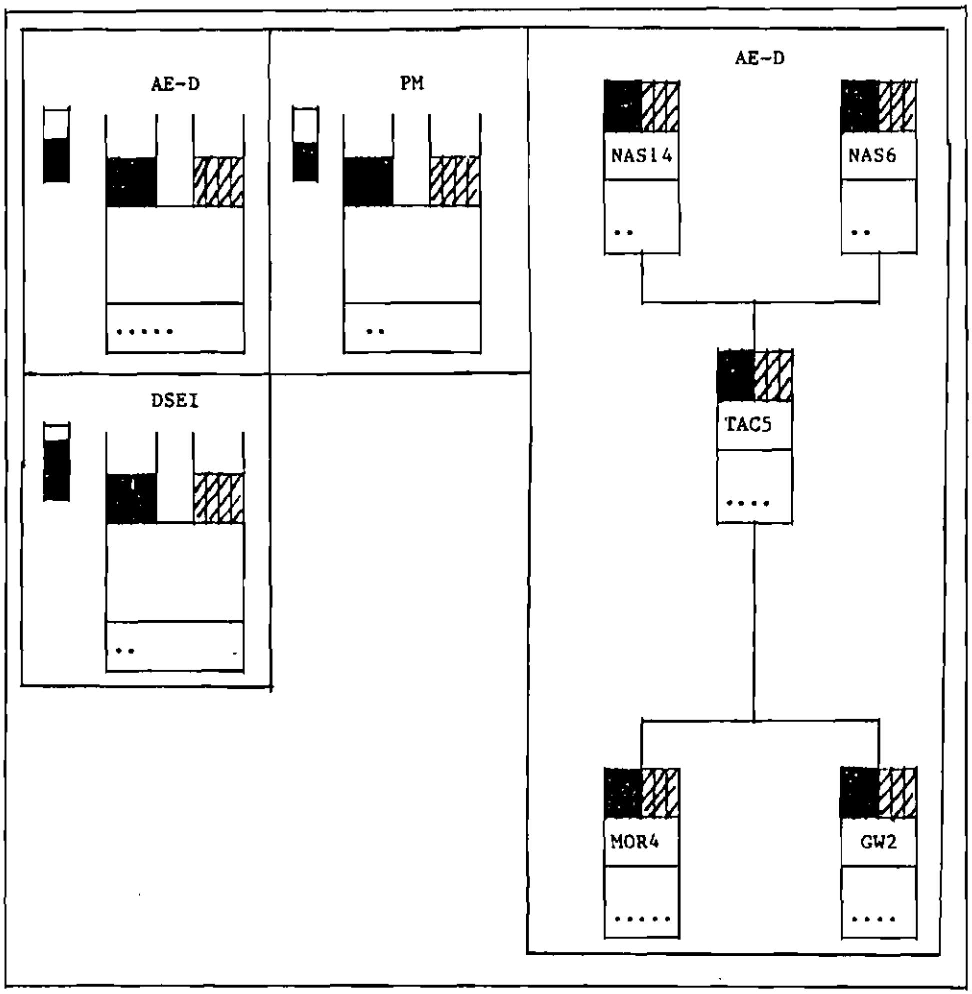


Figure 3.6 A Sample of the Right Graphics Monitor.

subfunction of the control of current missions function (Figures 3.2a and 3.2b).

The Left Monitor

At this point, the operator uses the left monitor to obtain necessary information to carry out the new control task. This screen provides an alphanumeric windowing environment in which the user enters commands and information requests, and receives messages and windows of information. Information within these windows assists the operator in compensating for system faults. Computer windows on this monitor are also available to aid the operator in performing other system tasks.

There are three permanent windows on the left monitor. There is a window containing the current time, a command window, and a message window. In the time window, Greenwich mean time (the standard world-wide time) is displayed and is updated every ten seconds. Within the command window, the operator enters requests to access information required for problem solving and enters control commands. In the message window, the operator receives system messages and alarms, many of which serve to alert the operator that a new control function may be necessary. This window contains only the most recent system message. To refer to a previous message, an operator can request an event log window containing a five minute history of system messages.

Also provided on the left monitor are sets of windows to aid the operator in accomplishing major system functions as defined by the operator function model described in a previous section. Each set of windows provides the necessary information to accomplish a major operator function. Three high level commands may be issued to request these windows:

```
HELP REPLACE equipment_name
HELP CONFIGURE mission_name
HELP CONFIGURE mission_name time_duration
```

The HELP REPLACE command aids the operator in fault compensation once a failure has been detected. Fault compensation is defined by the operator function model as an important subfunction in the control of current missions (Figure 3.2b). The first HELP CONFIGURE command aids the operator in compensation for automated schedule problems (Figure 3.2c), where the system scheduler is unable to configure a scheduled satellite contact because one or more components are unavailable. The second HELP CONFIGURE command aids the operator in responding to requests for support that is not scheduled. The information needed to configure an unschedule mission is given by the operator function model in Figure 3.2d. The following sections describe the windows provided by these "help" commands.

Windows to Help Replace a Component. The alphanumeric window CRT is where the operator requests and receives information needed for fault compensation. The HELP REPLACE command aids the operator in compensating for faulty equipment. After issuing the HELP REPLACE command, a window containing all suitable candidate replacement equipment for the faulty component is displayed. Given the component that needs to be replaced, the system displays a window containing all components that are operational, currently idle (or not loaded to full capacity), and available during the required time duration. To complete fault compensation, the operator replaces the malfunctioning component with one of the operational and available components provided in the aiding window.

Windows to Help Configure a Scheduled Pass. When the system is unable to configure a scheduled spacecraft contact because one or more pieces of equipment is unavailable, it is necessary for the operator to intervene. To compensate for an automated schedule problem (the second major GT-MSOCC operator function defined by the model) the operator first determines which scheduled hardware component is unavailable and then identifies candidate hardware to replace unavailable equipment (Figure 3.2c).

A HELP CONFIGURE command produces a set of windows that provides the operator with this information. The operator specifies a mission name, and if that mission is scheduled but not configured, the system provides aiding windows. One window gives a template containing all equipment scheduled for the satellite pass with the unavailable piece(s) marked with an asterisk. The same command also produces windows containing suitable replacements for each unavailable component. Figure 3.7 shows the resulting windows following an operator request for help in supporting a mission that could not be configured automatically. Within the figure, the template window indicates that AP3 and GW2 were scheduled for the ERBE satellite contact, but are not currently available. In addition to the mission template window, Figure 3.7 also displays windows listing potential replacement equipment for the two unavailable components. After calling information windows, the operator manually configures the equipment string sub-

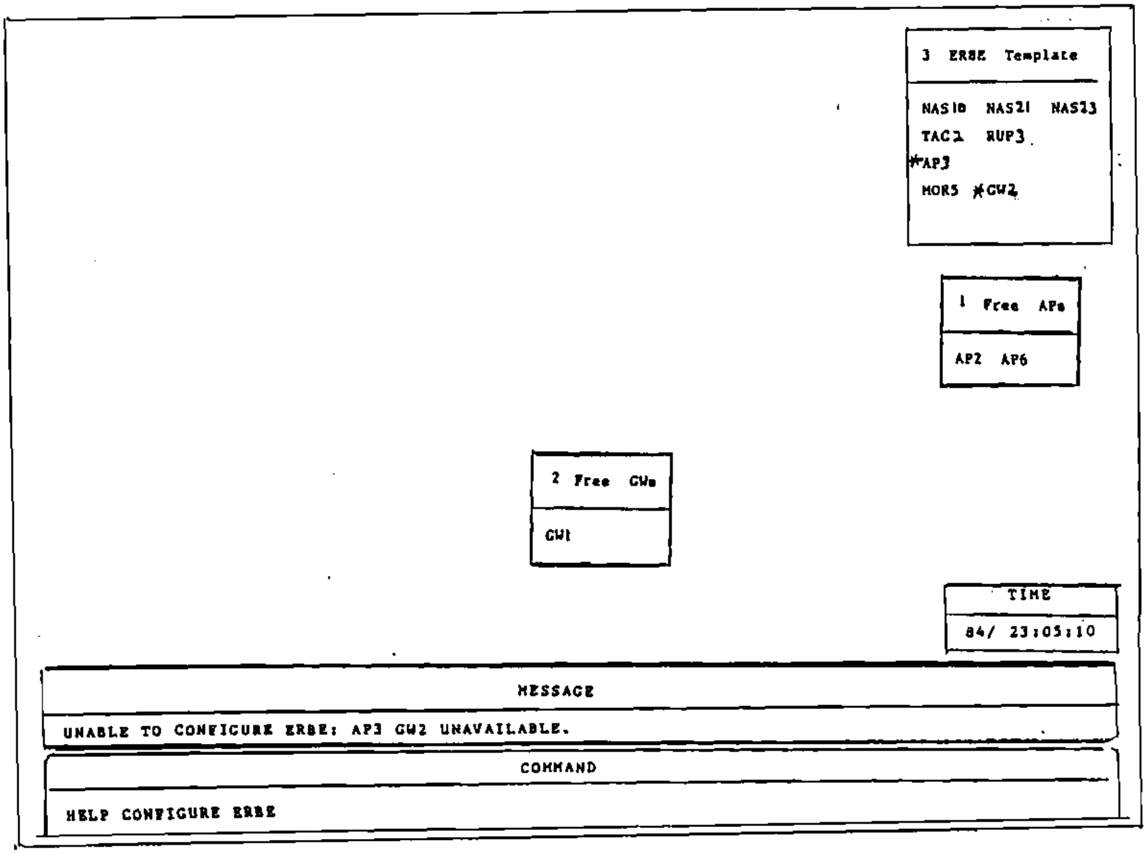


Figure 3.7 A Sample Response to the "HELP CONFIGURE ERBE".Command.

stituting suitable equipment for unavailable components (Figure 3.2c).

Windows to Help Configure an Unscheduled Pass. The next major GT-MSOCC operator function (Figure 3.2d) is to configure to meet support requests. The GT-MSOCC operator may receive a query concerning the feasibility of support for an additional spacecraft contact over a given time duration. A message to that effect and an audio alarm serve to notify the operator that an equipment configuration is required for unscheduled support.

Figure 3.2d presents the tasks comprising this function as defined by the operator function model. Before configuring to meet a support request, the operator must determine whether the GT-MSOCC system can maintain another mission and must ensure that the mission is not already scheduled within the requested time duration. If the contact is feasible, the operator determines what hardware is needed and identifies candidate hardware. Finally, if possible, the operator manually configures the mission. Using the specifications of the operator function model, a HELP CONFIGURE command adaptively calls the windows of information required by the operator to undertake the control function. This HELP CONFIGURE command differs from the one described above in that a time duration as well as a mission name must be specified by the operator. The command adapts to current system state and only displays relevant windows. For example, if a pass is not feasible for any reason (e.g., configuring to support another mission would exceed GT-MSOCC capacity of five concurrent missions), the operator would only receive a message to that effect, and would not see any additional windows. On the other hand, if a pass is feasible, the same HELP CONFIGURE command produces a set of windows to aid the operator in configuring equipment for the unscheduled support.

In this case, the command produces a template window indicating all equipment that the satellite requires, and also several windows that contain all candidate hardware. These windows contain equipment of the needed types that are free for the required time duration. Figure 3.8 provides a sample group of windows produced by the command to HELP CONFIGURE a 10 minute ERBE satellite pass when support for such a contact is feasible.

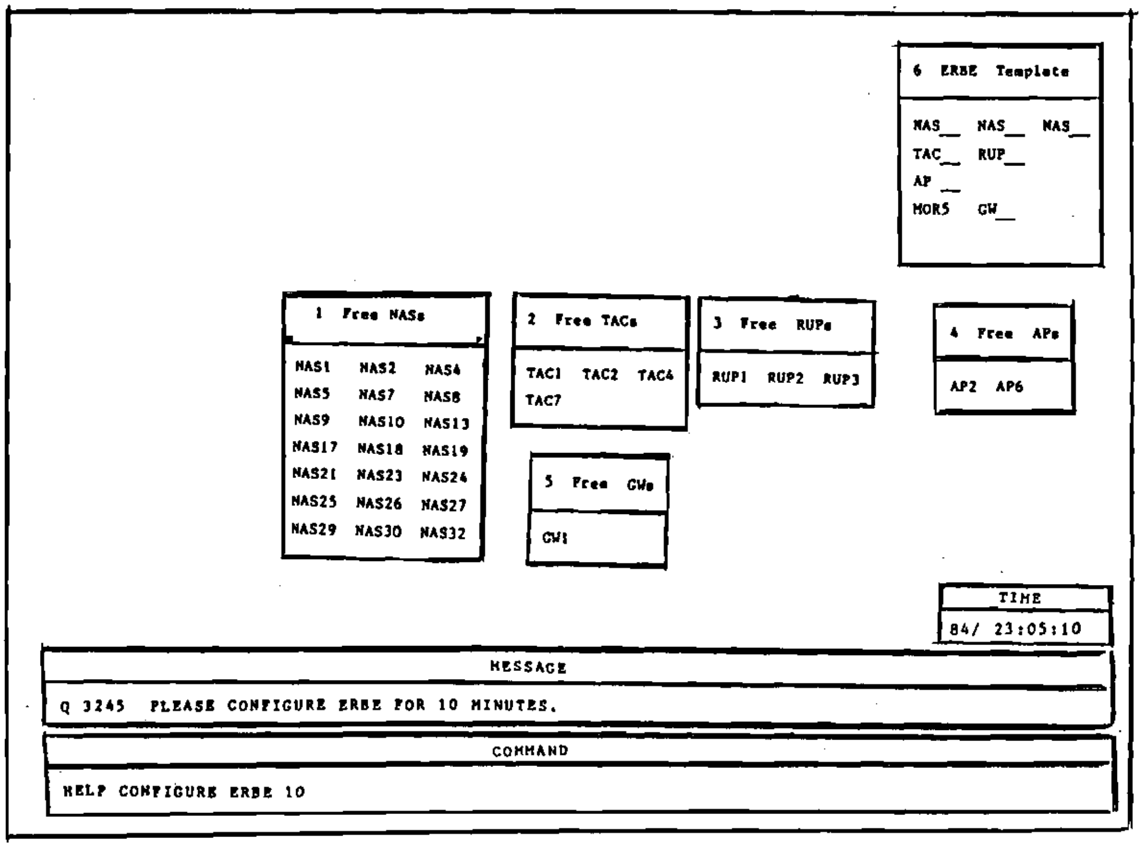


Figure 3.8 A Sample Response to a "HELP CONFIGURE ERBE 10" Command.

When one or more types of equipment have no available members, the operator receives a message to this effect and receives windows indicating the equipments' usage over the required time duration.

According to the GT-MSOCC operator model, there are two ways to complete this major operator function (configure to meet support requests). If the pass is not feasible or sufficient equipment is not available, the operator gives a negative response to the request for unscheduled support and makes no further attempt to configure the pass. If sufficient equipment is available the operator responds positively and commences to configure the mission.

Equipment Deconfiguration. No windows are available to aid the operator in manually deconfiguring equipment, because this function is so simple. As shown in Figure 3.2e, the operator first identifies the mission to deconfigure. In the GT-MSOCC system the operator receives a message to this effect. To complete this function the operator manually deconfigures the equipment string. Procedures to deconfigure equipment are the same using this interface as using the conventional interface that was defined in the previous chapter.

Windows to Help Plan for Known Future Problems. The final GT-MSOCC operator function is plan to compensate for potential future problems (Figure 3.2f). Unlike the other major operator functions, a single high level command is not available to provide the operator with the necessary preprocessed information to carry out this task. The operator must rely on schedules and equipment status information in order to determine the existence of potential scheduling problems

1 GT-MSOCC Schedule			
msn	up	down	equipment
DE	*	2:42	NAS28,23,21 RUP3 TAC6 AP3,5 MS1 VIP3 MOR13
WS-D	*2:38	2:43	NAS3,17 TAC3 AP5 VIP2 MOR9
AE-QL	2:41	2:47	NAS2,13 TAC2 AP1 CMS2 VIP1 MOR4
ISE	2:47	2:56	NAS2,30,12 RUP1 TAC4 AP3 CMS2 VIP3 MOR8
GEO	2:47	2:51	NAS26,22,15 RUP3 TAC2 AP1 CMS1 VIP2 MOR2
GSAT	2:54	3:02	NAS1,27,33 RUP2 TAC6 AP6 GW1 VIP2 MOR7
LNSAT	2:55	3:01	NAS6,12,31 RUP2 TAC5 AP2 GW2 VIP3 MOR3

Figure 3.9a A Sample Overall GT-MSOCC Schedule. This schedule indicates that DE is transmitting data and that mission WS-D is scheduled but not yet configured.

3 DE Schedule		
up	down	equipment
*	2:42	NAS28,23,21 RUP3 TAC6 AP3,5 CMS1 VIP3 MOR13
2:58	3:01	NAS26,24,7 RUP2 TAC8 AP4,3 CMS1 VIP3 MOR13
3:18	3:23	NAS9,25,8 RUP1 TAC6 AP6,7 CMS2 VIP3 MOR13
3:38	3:41	NAS30,11,13 RUP1 TAC6 AP6,7 CMS1 VIP3 MOR13
4:18	4:22	NAS11,24,17 RUP3 TAC8 AP7,6 CMS2 VIP2 MOR13
6:18	6:24	NAS26,5,21 RUP2 TAC1 AP7,3 CMS1 VIP2 MOR13

Figure 3.9b A Schedule for the Satellite DE.

2 AP5 Schedule		
msn	up	down
DE	*	2:42
WS-D	2:55	3:01
VENTR	3:01	3:07
WS-D	3:45	3:51
DSEI	4:01	4:09
VENTR	4:11	4:18

Figure 3.9c A Schedule for the Component AP5.

1 Status	
TAC1	IDLE
TAC2	IDLE
TAC3	BUSY
TAC4	IDLE-FAILED
TAC5	IDLE
TAC6	IDLE
TAC7	IDLE
TAC8	BUSY

Figure 3.10 A Status Window for the TAC Computers. A status window is available for each class of equipment.

and to prepare to compensate for these situations. Figures 3.9 and 3.10 give examples of schedule and status windows, respectively. To retrieve these display windows the command issued is "DISPLAY" followed by the name of the window of interest. For example, the command "DISPLAY MSOCC SCHED" accesses the overall system schedule. A complete list of the commands and the command syntax for the window interface is included at the end of Appendix C.

The schedules are similar to those in the conventional displays (Figures 2.4, 2.5 and 2.6) defined in the previous chapter, but unnecessary information has been deleted, based on the operator's information needs as determined by the operator function model. For example, the columns in Figure 2.4 listed Acquisition of Signal (AOS), Loss of Signal (LOS), orbit number, ground station, and data type which is important information for scientists who maintain the health and safety of the satellites. However, this information is irrelevant to the GT-MSOCC operator and is repeated on all alphanumeric schedules contained in the conventional interface. These columns were eliminated from the window schedules (Figure 3.9a). Also, to conserve screen space, schedules in the window interface provide a less extensive view of the future.

Aiding windows designed specifically for planning were not included in this interface design. Operators were encouraged to solve problems as they occurred.

Window Placement. Windows are arranged on the left screen so that the set of windows called by a "help" command and that are used to carry out a given task obstruct one another as little as possible. For example, the HELP REPLACE command provides a free-equipment retrieve these display windows the command issued is "DISPLAY" followed by the name of the window of interest. For example, the command "DISPLAY MSOCC SCHED" accesses the overall system schedule. A complete list of the commands and the command syntax for the window interface is included at the end of Appendix C.

The schedules are similar to those in the conventional displays (Figures 2.4, 2.5 and 2.6) defined in the previous chapter, but unnecessary information has been deleted, based on the operator's information needs as determined by the operator function model. For example, the columns in Figure 2.4 listed Acquisition of Signal (AOS), Loss of Signal (LOS), orbit number, ground station, and data type which is important information for scientists who maintain the health and safety of the satellites. However, this information is irrelevant to the GT-MSOCC operator and is repeated on all alphanumeric schedules contained in the conventional interface. These columns were eliminated from the window schedules (Figure 3.9a). Also, to conserve screen space, schedules in the window interface provide a less extensive view of the future.

Aiding windows designed specifically for planning were not included in this interface design. Operators were encouraged to solve problems as they occurred.

Window Placement. Windows are arranged on the left screen so that the set of windows called by a "help" command that are used to carry out a given task obstruct one another as little as possible. For example, the HELP REPLACE command provides free-equipment windows and a template window that do not overlap and do not obstruct the permanent message window, command window or time window. Each type of window has a dedicated location on the screen. Search time for information should decrease, if the operator knows where to expect windows. Woods (1984) states that providing a fixed format is a useful technique for data retrieval, since assigning classes of data to specific screen locations helps the user to link spatial location with data type. Figure 3.11 provides the format for different types of windows on the GT-MSOCC window monitor.

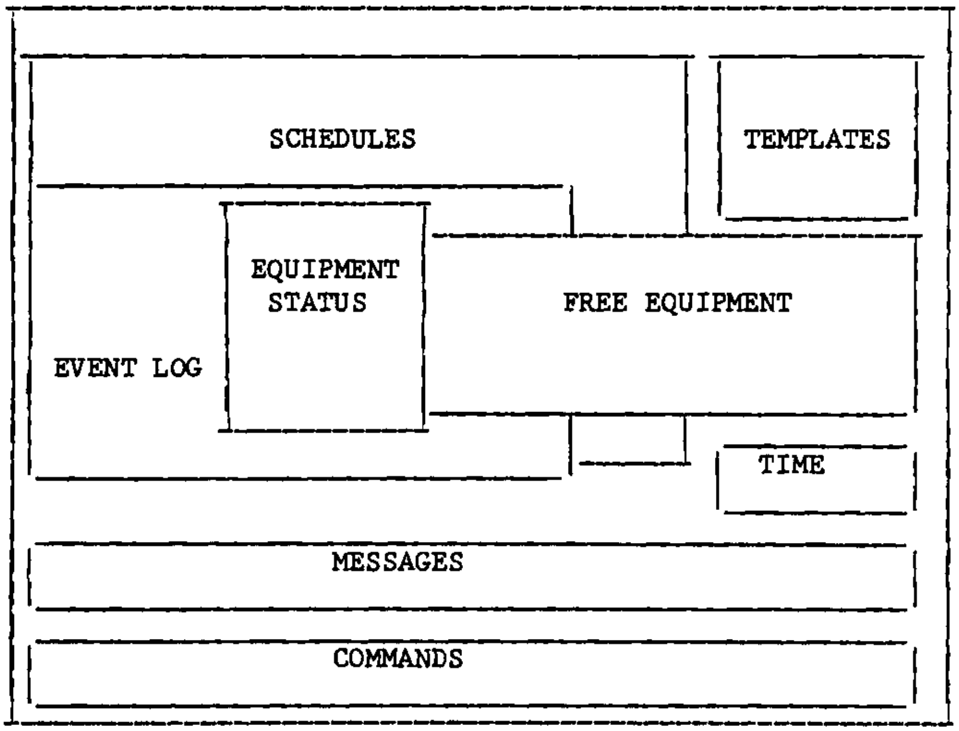


Figure 3.11 Dedicated Locations for Different Types of Windows.

Since the interface determines the placement of windows, the operator does not have to take time to format windows on the screen. Deleting windows, however, is an operator responsibility. Each window is numbered, and the operator may erase a single window (e.g., "ERASE 6"), a group of windows (e.g., "ERASE 1 2 3.8 5") or all of the window (e.g., "ERASE ALL"). The command window, the message window and the time windows are permanent and cannot be erased.

Summary of the Proposed GT-MSOCC Workstation

In summary, the proposed GT-MSOCC workstation consists of two CRT's. The right screen is dedicated to system monitoring and fault detection, since these are defined by the model to be primary operator functions. This screen uses dynamic icons to depict the most relevant features for monitoring and fault detection. The left screen contains alphanumeric windows designed to aid the operator in fault compensation and other operator functions. The contents, appearance and placement of windows are dictated by a detailed model that defines the information necessary to accomplish major GT-MSOCC operator functions.

The proposed interface uses the same control commands as the conventional interface introduced in the previous chapter. The only difference between the interfaces is the information retrieval commands and displayed information. The conventional interface is comprised of full-screen information displays that maintain a consistent format and are dynamic only with respect to changing values of variables displayed within a page. Operators must determine which pages are useful to perform a task and must select and integrate information from multiple screens. The proposed experimental interface contains qualitative icons and alphanumeric windows whose appearance and contents adapt not only to changes in system events, but also to the changing information requirements of the GT-MSOCC operator. This interface gives the operator high level commands that propose a more detailed iconic description of equipment status or a set or alphanumeric windows likely to be most valuable for accomplishing the current operator function.

Chapter II presented the conventional NASA interface to the GT-MSOCC system. In this chapter a model of the GT-MSOCC operator was briefly described and a icon/window interface based on the model was explained. The following chapter describes an experiment that compared the two GT-MSOCC interfaces and includes a presentation and discussion of the resulting statistical analyses.

CHAPTER IV

AN EXPERIMENT

The main purpose of the experiment described in this chapter is to evaluate a supervisory control system interface that was based on an operator function model of the system. The interface incorporates dynamic, qualitative icons and an alphanumeric window environment. The last chapter demonstrated how an operator function model could act as a control structure for the appearance and contents of display windows. In this chapter, an experiment to compare the model-controlled window/icon interface with a conventional full-screen interface to the system is described.

Method

Subjects

Twenty students from Georgia Institute of Technology, sixteen males and four females, participated in the experiment. All subjects were students in an introductory, undergraduate, man-machines systems course.

Exerimental Materials

Three sets of written instructions were used in the experiment. The first set of instructions consisted of an introduction to the GT-MSOCC operator control functions (Appendix A). All twenty subjects received this introductory set of instructions. The second set of instructions explained detailed procedures for operating the system with either the conventional interface or the window interface, respectively (Appendices B and C). Both sets of procedures described the operator function priorities and contained exercises to teach the subject how to identify faulty communications equipment.

Other materials used in the experiment included reminding sheets that provided a summary of available control commands and information requests. In addition, subjects working with the conventional interface received blank paper for calculations and a table that listed the communications equipment required by each spacecraft. These other materials are located at the end of Appendices B and C.

A consent form was issued to subjects before the experiment began. After all experimental sessions were completed, a questionnaire was given to subjects to elicit their opinions about the interfaces and the experimental task (Appendix D).

Procedure

Subjects engaged in a total of 12 sessions each and were paid 5 dollars per session. The length of the first three sessions was 60 minutes; the remaining nine sessions lasted 45 minutes. Sessions were run on consecutive days with one session per day. Occasionally subjects missed one or two days or ran two sessions in a single day, with the sessions separated by at least two hours.

Overview of Experimental Sessions. Subjects controlled the GT-MSOCC system for 12 sessions. The first five sessions were considered training sessions, during which subjects received oral instructions and written exercises and controlled GT-MSOCC with and without assistance. During training sessions, an experimenter was available to answer all subjects' questions relevant to operating GT-MSOCC.

Within each session, three hardware failures and six software failures occurred. As described in Chapters II and III, there are three types of software failures: complete termination of data transmission, decreased rate of data transmission, and a high rate of error block accumulation. There were two occurrences of each of these software failures per session. Failures were scheduled to occur at set times on identical equipment across subjects. However, since not all subjects operated the system optimally (e.g., some subjects neglected to configure equipment for a scheduled contact) occasionally failures occurred on different pieces of equipment.

Every session, operators received three requests for support of unscheduled spacecraft contacts. Occurrences of equipment configuration requests were also identical across subjects.

Experimental Procedure. At the beginning of Session 1, subjects were informed that they would act as operators of a simulated NASA satellite communications system. Subjects signed a consent form stating that their participation was voluntary and informing subjects that payment was contingent on completion of all 12 sessions.

All subjects agreed to participate, and no one dropped out of the experiment. After signing the form, subjects were read an overview explaining the purpose and goals of GT-MSOCC. For the last 15 minutes of the first session, subjects controlled the system by requesting each of the information displays comprising the GT-MSOCC interface. An experimenter remained with the subjects throughout the first session and provided assistance in interpreting displays.

During Session 2, subjects were read instructions explaining detailed procedures for operating GT-MSOCC using either the conventional interface or the window/icon interface. Subjects received online practice using the system to compensate for each type of system malfunction. An experimenter assisted the subject with requesting information displays and implementing commands to carry out operator control functions.

Session 3 began with written exercises for detecting system failures. After completing the exercises, subjects operated GT-MSOCC for approximately 45 minutes. An experimenter noted system problems that occurred and aided the subject in correcting these situations.

Starting with Session 4, subjects controlled the system for the entire session. An experimenter was available for the fourth and fifth sessions to give advice and answer questions. The amount of experimenter assistance during these two sessions depended on the difficulty subjects had supervising and controlling the system. For the remaining seven sessions (Session 6 through Session 12), the experimenter did not offer assistance.

After completing the last session, subjects received a questionnaire asking them to specify positive and negative aspects of the user interface. Then, the purpose of the experiment was explained, and any questions concerning this research were addressed.

Dependent Measures. Rather than computing one overall measure of operator performance, several measures were collected to reflect how well subjects operated the GT-MSOCC system. Each dependent measure reflects operator performance on one of the control functions presented in the previous chapter. Tables 4.1a and 4.1b provide an overview of the relationship between the icon/window interface, the control functions and the performance measures.

Table 4.1a lists the control functions that are supported by the window environment and those that are supported by dynamic icons. To analyze operator performance, a number of dependent measures were recorded, each of which required the completion of one or more control functions. Table 4.1b maps performance measures to control functions. The first dependent variable, time to replace a component that has a hardware or software problem, measures operator performance on monitoring, fault identification, and fault compensation. Referring to Table 4.1a it can be seen that monitoring and fault

Table 4.1a
The Operator Control Functions Suppported by the Dynamic Icons and the Window Environment

OPERATOR CONTROL FUNCTIONS SUPPORTED BY DYNAMIC ICONS	OPERATOR CONTROL FUNCTIONS SUPPORTED BY A WINDOW ENVIRONMENT
- Monitoring - Fault Detection	- Fault Compensation - Compensation for Schedule Conflicts - Configuration to Meet Support Requests - Strategic Planning

Table 4.1b
The Operator Control Functions Required for Each Performance Measure

PERFORMAMCE MEASURE	REQUIRED CONTROL FUNCTIONS
Time to replace a component that has a hardware or software problem	Monitoring, Fault Detection, Fault Compensation
Time to configure a scheduled pass	Compensation for Schedule Conflicts, Strategic Planning
Number of correct responses to support requests, Time to respond, Time to configure for support	Configuration to Meet Support Requests
Number of operator caused schedule conflicts	(Poor) Fault Compensation
Number of unnecessary replacements	(Poor) Fault Detection
Time to Deconfigure Equipment	Manual Deconfiguration

detection are supported by dynamic icons and that the windowing environment supports fault compensation. Thus, both the dynamic icons and the windowing environment influence time to replace malfunctioning equipment. Similarly, each of the other dependent variables provides a measure of the utility of the dynamic icons, the window environment, or both. An exception is the measure of time to deconfigure which can be accomplished without using either icons or the window environment.

Data for a total of eleven performance measures was analyzed. The performance measures can be grouped into three broad categories: fault compensation, equipment configuration and deconfiguration, and operator errors. Table 4.2 lists and defines each dependent measure within these three categories.

Equipment may fail due to either hardware or software malfunctions. Four performance measures were collected to indicate time to compensate for faulty equipment. The four performance measures reflect time to correct faulty equipment that has failed in one of four ways. The first type of failure is due to hardware problems. A component that has the second type of failure experiences terminated data transmission due to software problems. The third kind of failure is a more subtle case in which software problems cause data to flow at a decreased rate. Finally, faulty software can cause data blocks to become garbled or arrive out of sequence.

To compensate for a failure, the operator identified the malfunctioning component and replaced it with a comparable operational

Table 4.2
GT-MSOCC Operator Performance Measures

MEASURE	OPERATOR TASK
FAULT COMPENSATION	
1. Time to Compensate for a Hardware Failure	Replace a component that is inoperable due to hardware malfunctioning
2. Time to Compensate for Software Failure 1	Replace a component that has stopped processing data due to software bugs
3. Time to Compensate for Software Failure 2	Replace a component that is processing data at a decreased rate
4. Time to Compensate for Software Failure 3	Replace a component producing error blocks due to a software problem
EQUIPMENT CONFIGURATION AND DECONFIGURATON	
5. Time to Compensate for a Schedule Conflict	Configure a mission replacing unavailable scheduled equipment
6. Number of Correct Responses	Determine the feasibility of an unscheduled spacecraft contact
7. Time to Respond to a Support Request	Determine the feasibility of an unscheduled spacecraft contact
8. Time to Configure an Unscheduled Contact	Configure equipment for unscheduled mission support
9. Time to Deconfigure	Deconfigure manually configured equipment when support is completed
OPERATOR ERRORS	
10. Number of Operator Errors: Type 1	Cause a scheduling conflict by using a scheduled component
11. Number of Operator Errors: Type 2	Replace a component that is not failed

component. If an operator neglected to replace a failed component, the data point recorded was the time the failure occurred to the time the mission(s) using the faulty component completed data transmission.

The next four performance measures reflect the time to configure or deconfigure GT-MSOCC equipment. The first of these measures (numbered 5 in Table 4.2) is the time to compensate when the system scheduler can not configure equipment automatically. If any equipment in the scheduled configuration is unavailable at the time of the pass, equipment configuration to support the satellite contact becomes the operator's responsibility. There is a three minute time limit to configure equipment for a scheduled support.

The next three measures reflect operator performance in configuring equipment to support contacts that were not scheduled. The first of these measures is the number of correct responses per session to requests for unschedules support. Unscheduled passes were requested three times each session so each data point recorded was a number from zero to three. The next measure (measure 7) was the time to respond to the query. Response times were included only when the operator responded correctly. Thus, this was an unbalanced design, in that between zero and three data points were collected for each subject in each session. When the subject correctly answered that equipment for unscheduled support could be configured, the time to configure equipment was recorded (measure 8).

When the operator configured or altered an equipment string, he or she deconfigured the equipment at the completion of the contact.

Time to deconfigure equipment was the next performance measure. Since there was no time constraint on this measure, the data point recorded when an operator neglected to manually deconfigure equipment was the time between when the event occurred and when the session ended.

Two types of operator errors were measured. The first operator error is causing an equipment scheduling conflict. Replacing normally functioning equipment is the next error. The number of times that subjects committed each of these errors per session was computed.

Statistical Analysis

The linear statistical model used to analyze the data from this experiment is a mixed effect, balanced, nested factorial design. Not all dependent measures had a fixed number of repetitions per cell, and thus, the design in some cases is unbalanced. The main factors in this experimental design were condition and session.

There were two display conditions. Subjects in the first condition used the conventional NASA interface described in Chapter II that consisted of a dedicated color graphics page and about one hundred full-screen, alphanumeric display pages. This condition is referred to as the conventional display condition. In the second display condition, termed the window display condition, subjects used an interface controlled by a cognitive model of the operator's information needs. This interface, described in the previous chapter, incorporated dynamic icons and an alphanumeric windowing environment.

In the experimental design, subjects were nested within condition, since each individual participated in only one of the two display conditions. There may be a condition x session interaction and a session x subject within condition interaction. No condition x subject interaction can exist, however, since subjects did not participate in both display conditions. Similarly, there can be no three-way condition x subject x session interaction.

Since a mixed design with nested factors and unbalanced data was analyzed, it was necessary to construct approximate F statistics. Satterwaite's method (Montgomery, 1984; Satterwaite, 1946) for approximating an F statistic by taking a ratio of linear combinations of expected means squares was used. The degrees of freedom for the numerator and denominator may not be integers. For these cases, it is necessary to interpolate in the tables of the F distribution. Appendix E describes the method used to construct approximate F statistics and to compute degrees of freedom for this study.

Statistical analyses were performed using the General Linear Model (GLM) procedure of SAS statistical software (Spector, Goodnight, Sall and Sarle, 1985). This statistical package allows the user to specify the linear statistical model and define effects as either fixed or random. The General Linear Model procedure gives the expected mean square for each effect, and the user determines the appropriate numerator and denominator to create each F statistic.

Results

Analyses of variance were performed to determine the effect of the independent variables (condition, session, subject) on each of the eleven dependent measures listed in Table 4.2. In this analysis, a nonsignificant result is defined an effect having a significance level of greater than .05. Since the influence of condition (conventional interface vs. window-based interface) is the factor of interest, this section examines differences in subject performance in the two display conditions. The effect of session, subject nested in condition and the influence of interaction effects on performance measures are also noted in the following sections.

Compensation for Hardware Failures

When time to compensate for a hardware failure was used as the dependent measure, the effect of display condition was not significant, although group means differed in the expected direction. That is, the mean time to compensate for hardware failures in the conventional display condition (56.4 seconds) was higher than in the window display condition (42.5 seconds). The condition x session interaction effect, $F(6,108) = 2.66$, $p < .02$, did significantly affected time to compensate for hardware failures. However, the main effect of session was not significant. The effect of subject (condition) also was not significant, nor was the session x subject (condition) interaction effect. A plot of the means for the two display condi-

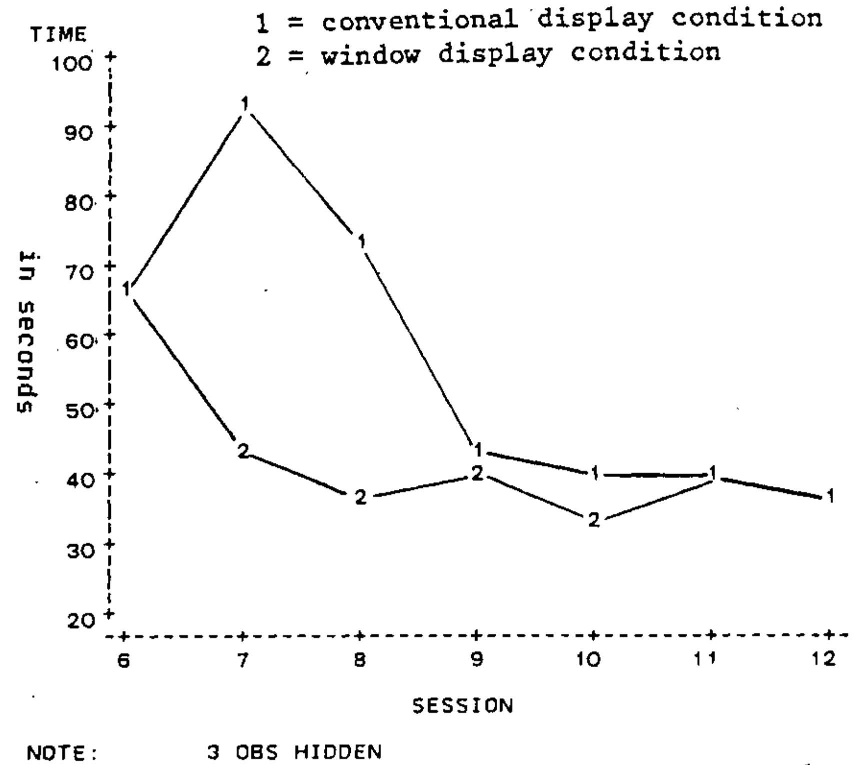


Figure 4.1a Mean Time to Compensate for Hardware Failures per Session by Display Condition.

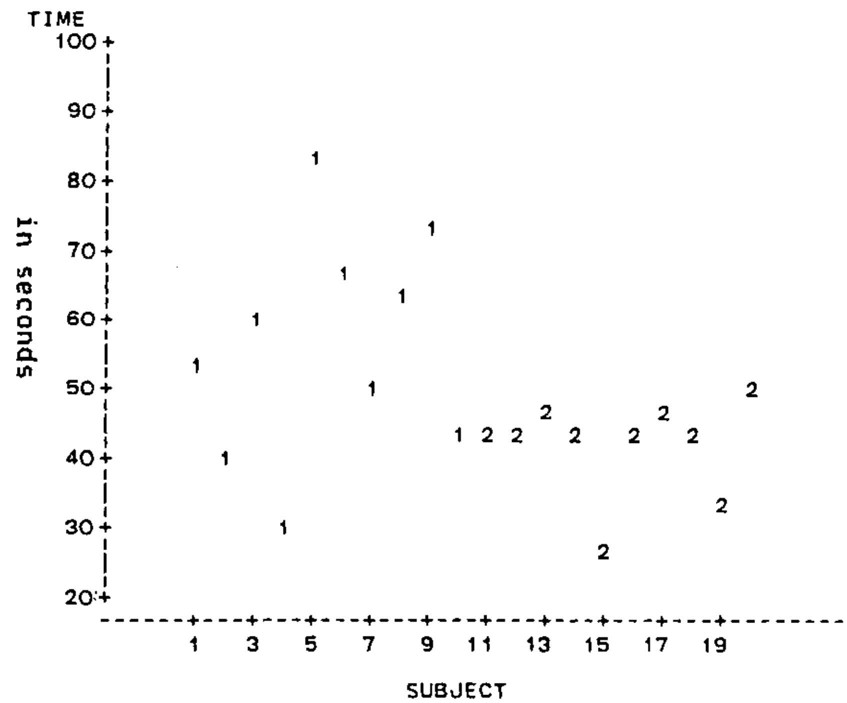


Figure 4.1b Mean Time to Compensate for Hardware Failures per Subject within Display Condition.

tions for each session is provide in Figure 4.1a. Figure 4.1b shows the means for each individual subject.

__Compensation for Software Failure 1: Termination of Data Flow__

The main effect of condition, $F(1.07,7.27) = 10.95$, $p < .02$, was significant when time to compensate for the first type of software failure was analyzed. This failure caused the termination of data transmission at a component due to software problems. As with the previous measure, the mean time to compensate was greater in the conventional display condition (312.4 seconds) than in the window display condition (56.9 seconds). Also significant was the condition x session interaction, $F(6,108) = 11.78$, $p < .001$. The main effect of session was not significant. No significant difference was detected between subjects for subject (condition). Nor did the session x subject (condition) interaction achieve significance. Means across sessions and subject for the two display conditions are shown in Figures 4.2a and 4.2b.

__Compensation for Software Failure 2: Decreased Rate of Data Flow__

The second software failure caused a decreased rate of data processing at a component. When the effect of condition on time to compensate for this software failure was analyzed, $F(1.00,10.60) = 92.01$, $p < .001$, the mean for the conventional condition (398.98 seconds) was significantly higher than the window condition mean (71.7 seconds). The effect of session and the higher order effect of

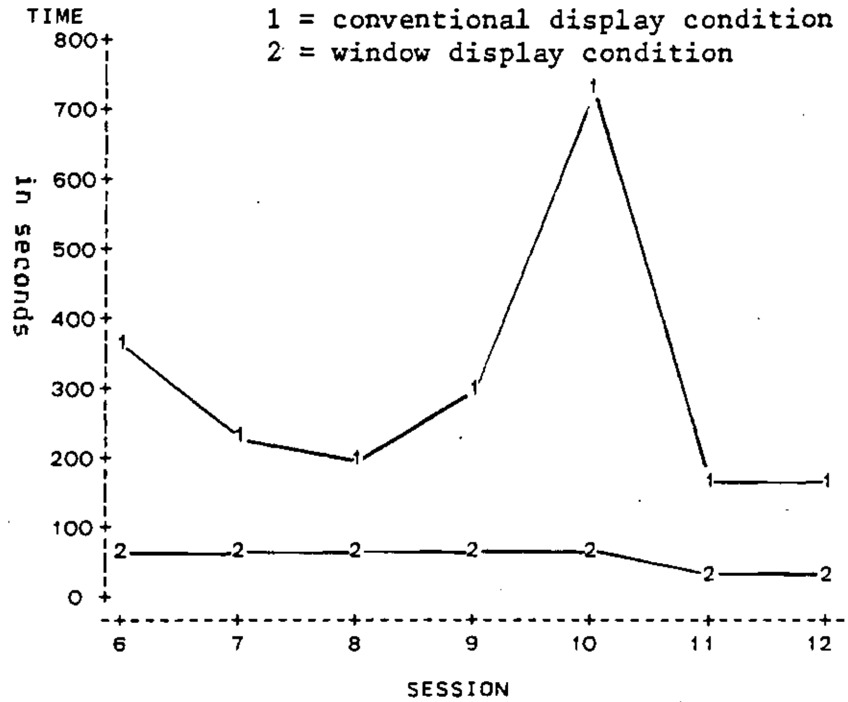


Figure 4.2a Mean Time to Compensate for Software Failure 1 per Session by Display Condition.

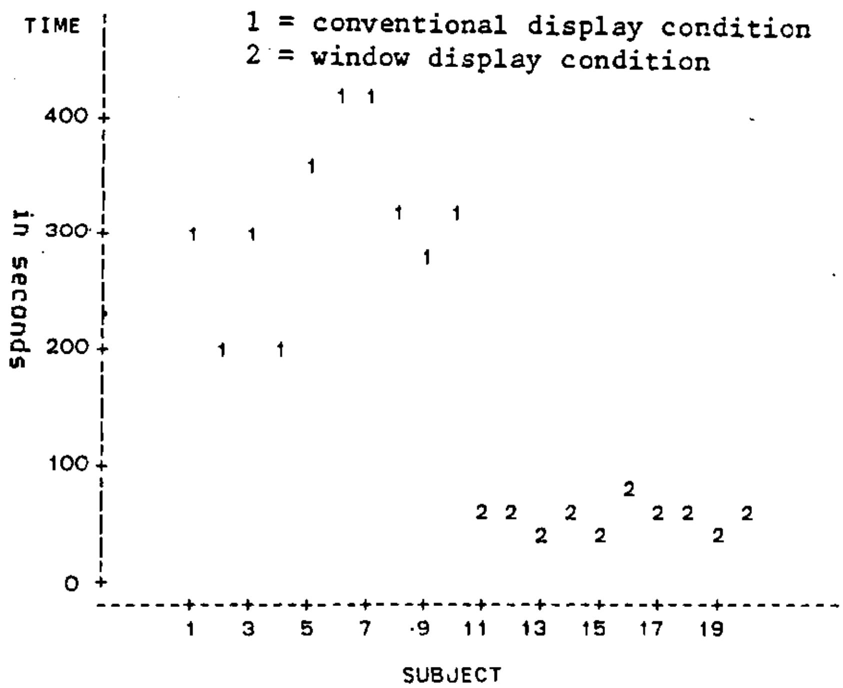


Figure 4.2b Mean Time to Compensate for Software Failure 1 per Subject within Display Condition.

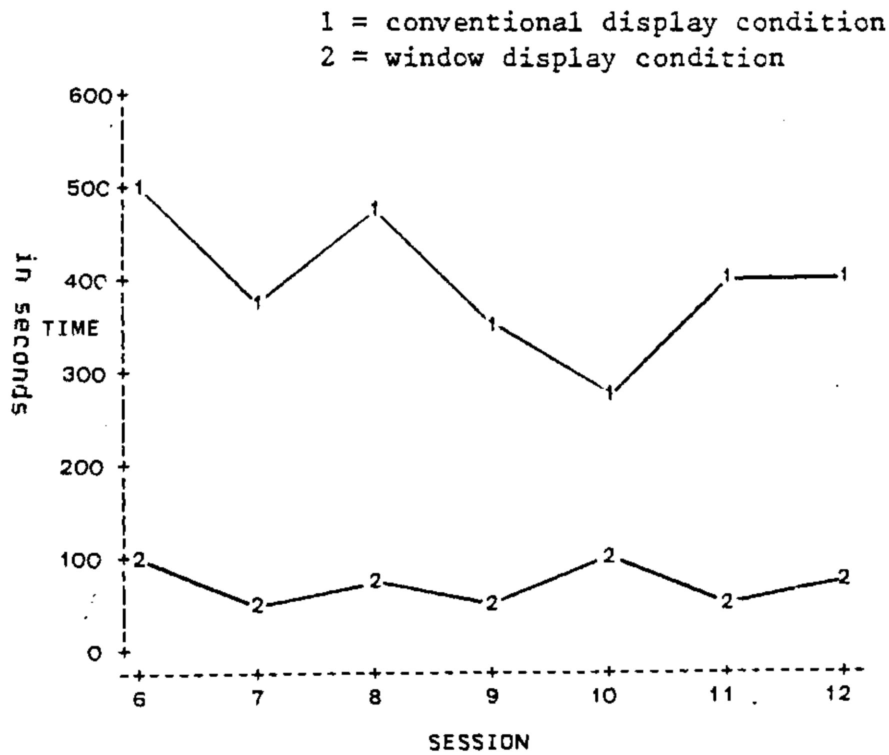


Figure 4.3a Mean Time to Compensate for Software Failure 2 per Session by Display Condition.

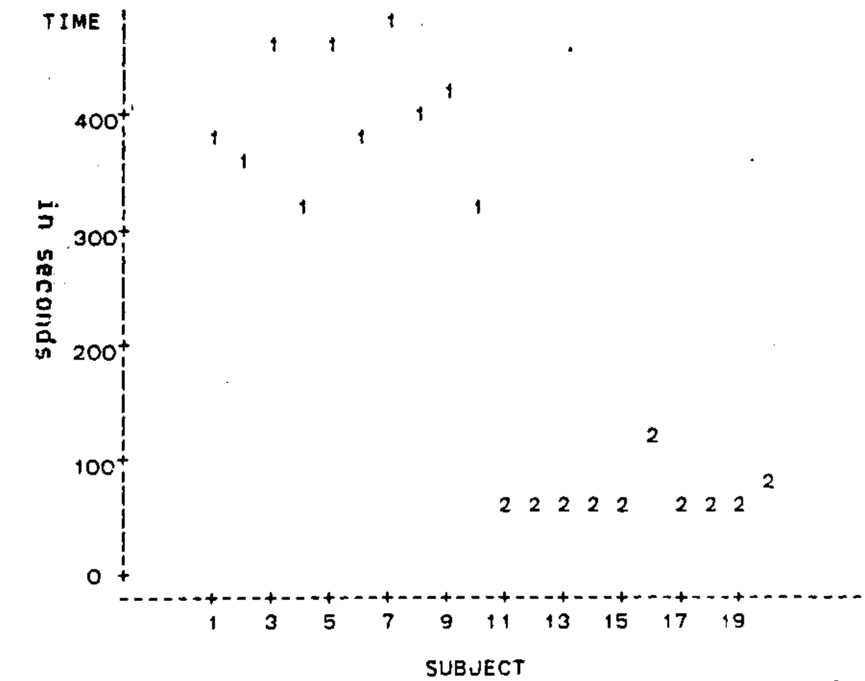


Figure 4.3b Mean Time to Compensate for Software Failure 2 per Subject within Display Condition.

session x subject (condition) were the only effects not to reach significance. The effects of subject (condition), $F(18,108) = 2.00$, $p < .02$ and condition x session, $F(6,108) = 5.59$, $p < .001$ were both significant. Plots for this measure are shown in Figures 4.3a and 4.3b.

Compensation for Software Failure 3: High Error Block Rate

Using the time to compensate for failures caused by high error block counts as the dependent variable indicated that the main effect of condition, $F(1.02,13.25) = 18.30$, $p < .001$, was significant in the expected direction. In the conventional display condition, the mean was 356.7 seconds, and the window display condition mean was 206.0 seconds. The effect of session and the effect of the session x subject (condition) interaction on time to compensate for this software failure were once again not significant. However, the other effects were significant: subject (condition) = 2.38, $p < .003$, and condition x session, $F(6,108) = 4.22$, $p < .001$. See Figure 4.4a and 4.4b for plots.

Compensation for Scheduling Conflicts

The time to compensate for scheduling conflicts was not significantly affected by display condition, although the means lie in the expected direction, (i.e., 75.9 seconds for the conventional display condition and 46.9 seconds for the window display condition). The effects to achieve significance on this measure were, session, $F(6.01,6.02) = 6.18$, $p < .02$ and subject (condition), $F(18.00,125.49)$

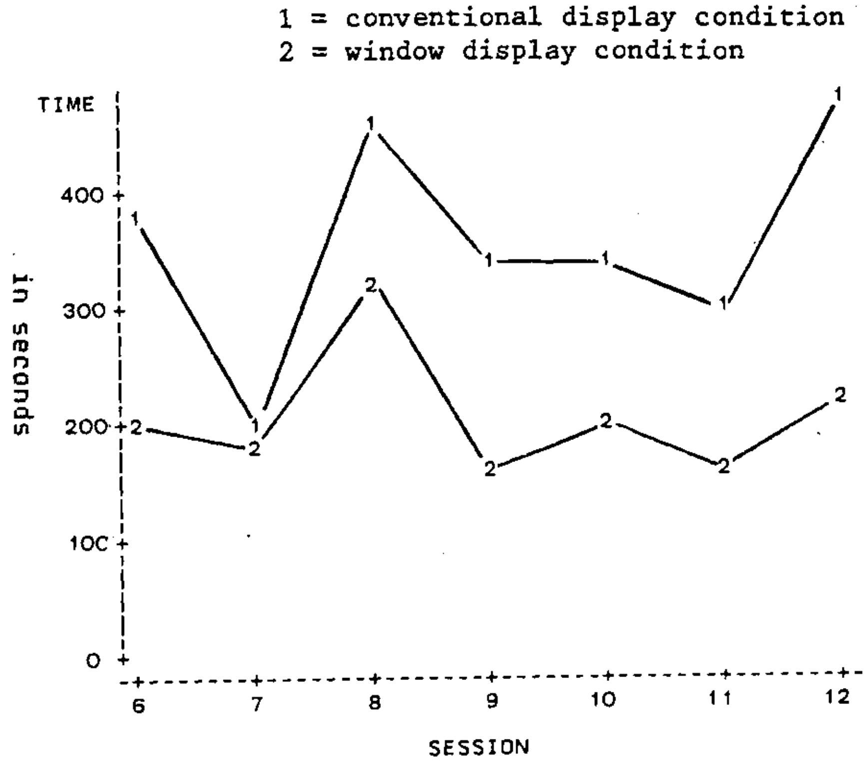


Figure 4.4a Mean Time to Compensate for Software Failure 3 per Session by Display Condition.

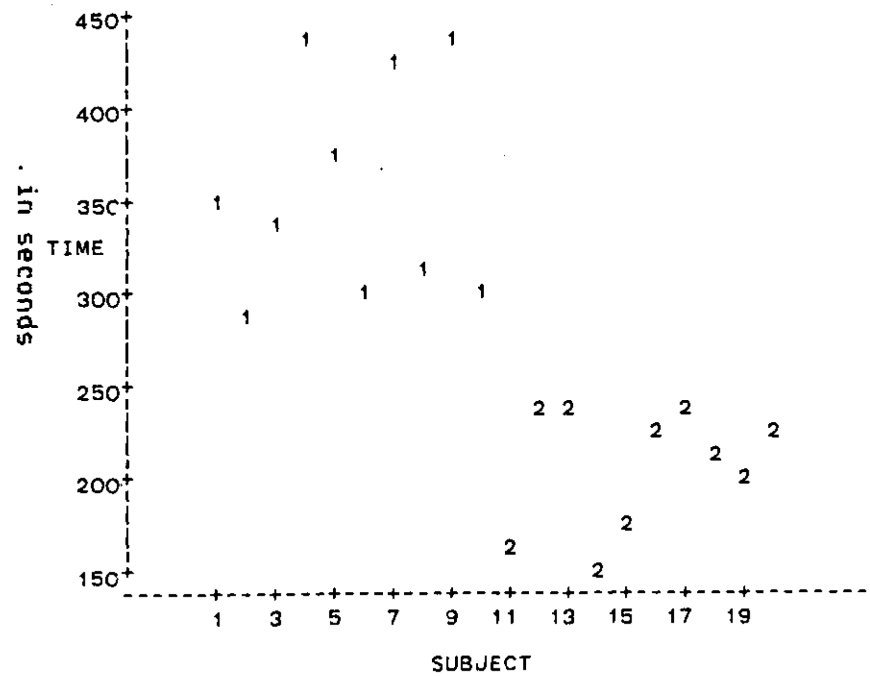


Figure 4.4b Mean Time to Compensate for Software Failure 3 per Subject within Display Condition.

= 5.03, $p < .001$. The significance level was also reached by the condition x session interaction, $F(6.00,115.23) = 2.42$, $p < .03$. As with all previous measures, the session x subject (condition) interaction fell below the significance level. See Figure 4.5a and 4.5b for mean times to compensate for scheduling conflicts across session and across subjects.

Support of Unscheduled Spacecraft Contacts

Three performance measures were analyzed to reflect how well operators responded to requests for unscheduled support. The first measure was the number of correct responses per session. The session and subject means for this dependent variable are found in Figures 4.6a and 4.6b. Recall that three support queries occurred each session. Display condition was the only significant effect on this measure, $F(1.10,19.32) = 8.47$, $p < .01$. The subjects in the window display condition answered significantly more questions correctly (means were 2.14 versus 2.67). The second measure was time to correctly respond to questions. Subjects received three requests per session, but only the correct responses were included in the analysis of time to respond. Condition was again significant, $F(1.01, 23.22) = 25.74$, $p < .001$, and in addition the effect of subject (condition) was significant, $F(18.00,121.29) = 5.83$, $p < .001$. Figures 4.7a and 4.7b indicate mean times to respond to queries across session and subject. Time to configure equipment for unscheduled support was the final measure. Configuration time was not included in the analysis,

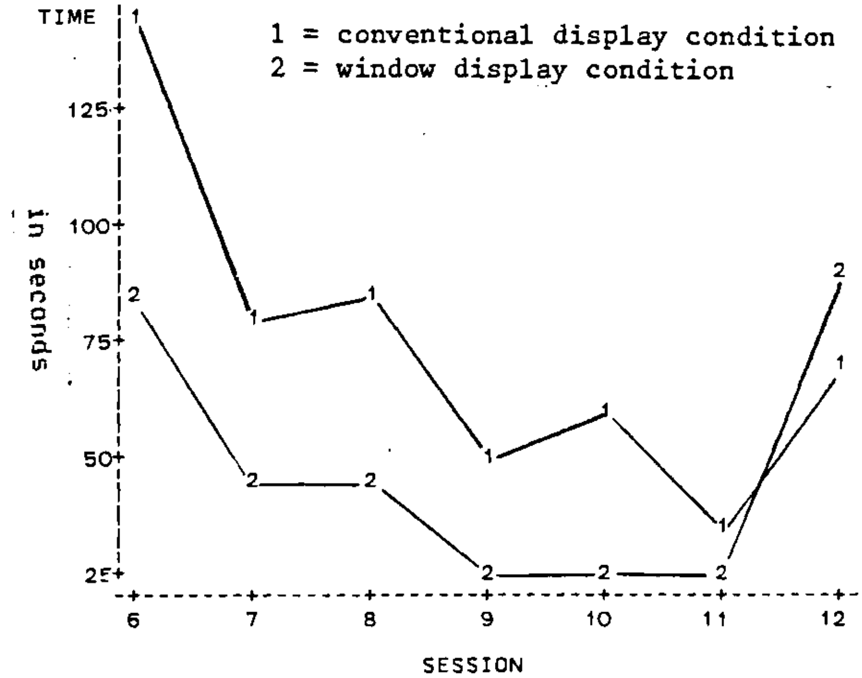


Figure 4.5a Mean Time to Compensate for Scheduling Conflicts per Session by Display Condition.

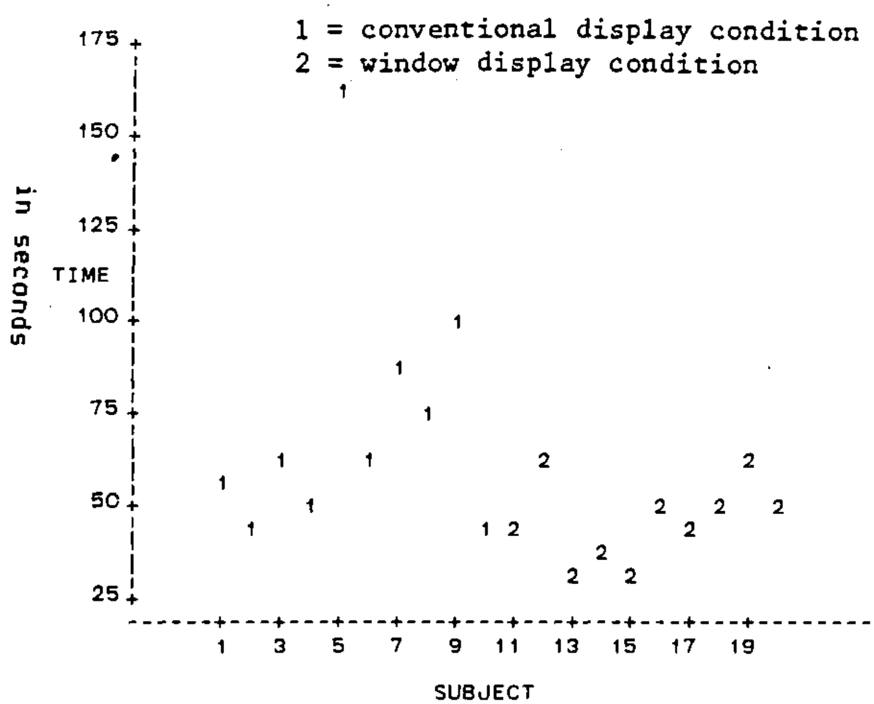


Figure 4.5b Mean Time to Compensate for Scheduling Conflicts per Subject withn Display Condition.

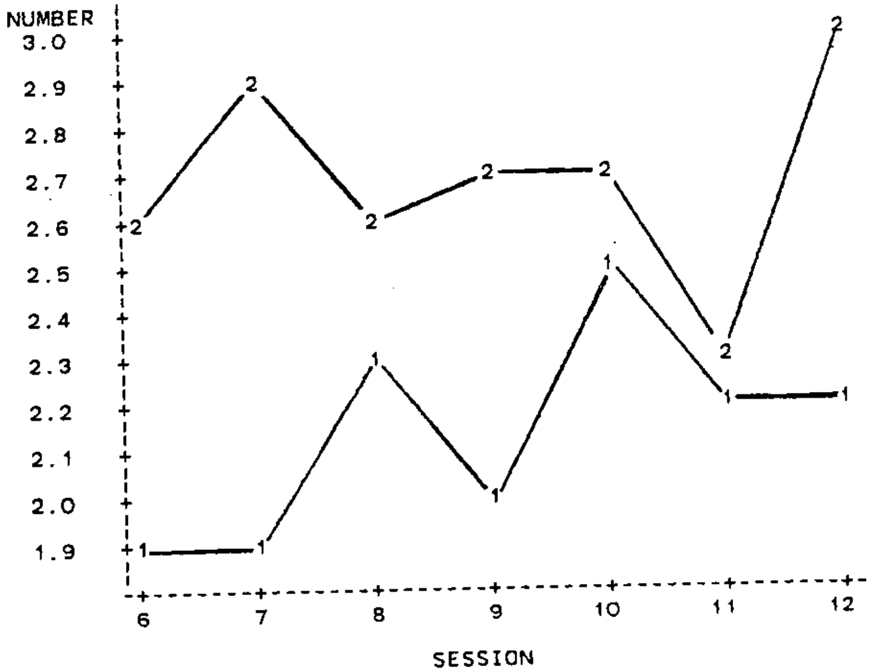


Figure 4.6a Mean Number of Correctly Responded to Support Requests per Session by Display Condition.

1 = conventional display condition
2 = window display condition

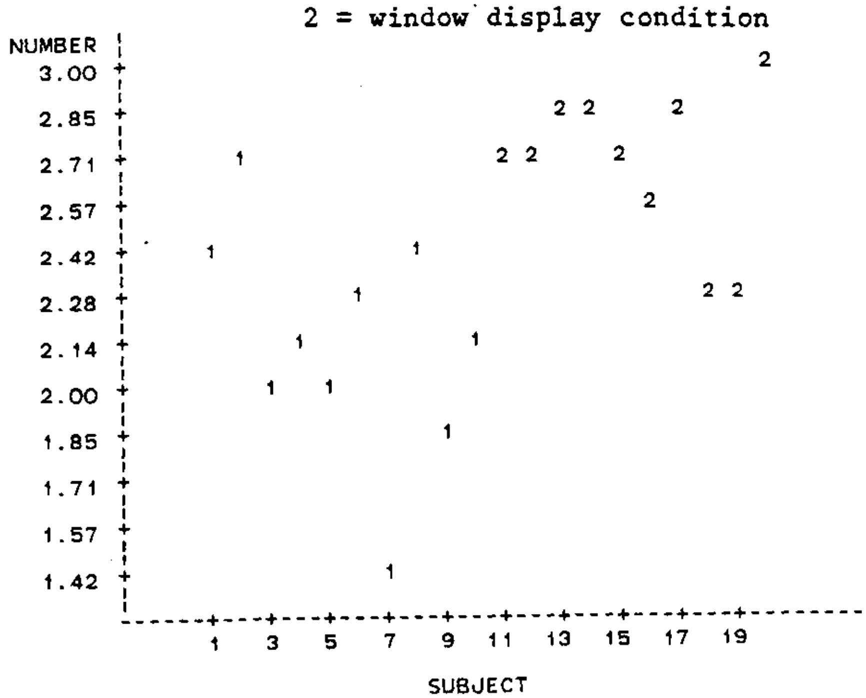


Figure 4.6b Mean Number of Correctly Responded to Support Requests per Subject within Display Condition.

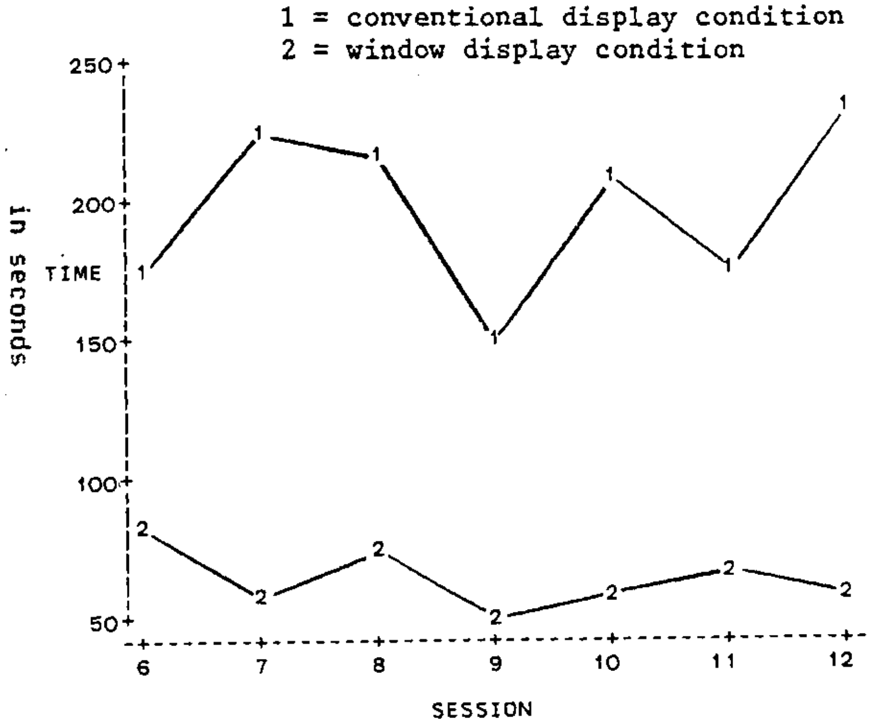


Figure 4.7a Mean Time to Respond to Support Requests per Session by Display Condition.

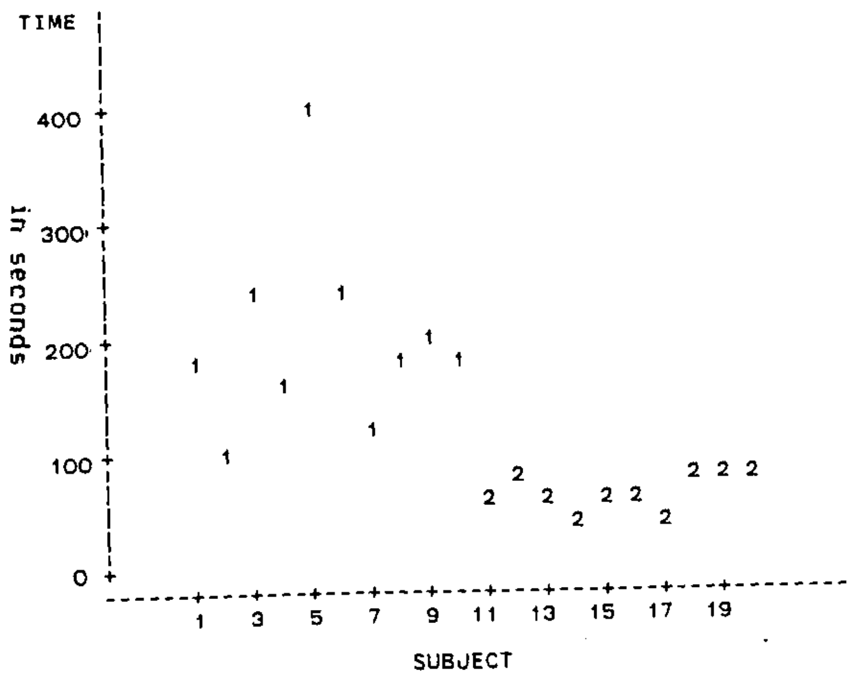


Figure 4.7b Mean Time to Respond to Support Requests per Subject within Display Condition.

when the subject configured equipment for a support that was not feasible. Again condition, $F(1.00, 22.84) = 36.82$, $p < .001$, and subject (condition), $F(17.00,91.54) = 3.90$, $p < .001$, were the only two effects to be significant. Subjects in the window display condition answered significantly faster (199.3 seconds versus 63.0 seconds) and configured equipment more quickly (260.4 seconds versus 90.6 seconds). Mean times to configure support requests across session and subject are plotted in Figures 4.8a and 4.8b.

Deconfiguration

Display condition was the only main effect to significantly affect time to deconfigure an equipment string, $F(1.16, 18.09) = 6.14$, $p < .02$. Subjects in the window display condition (mean = 11.1 seconds) manually deconfigured equipment faster than those that used the conventional interface (mean = 22.6 seconds). For the first time, the session x subject (condition) interaction was significant, $F(108,768) = 1.75$, $p < .001$. However, none of the remaining effects obtained significance. Session, subject (condition), and condition x session did not significantly influence performance. For each session the mean time to deconfigure is shown in Figure 4.9a; Figure 4.9b shows the across subject mean deconfiguration time.

Operator Error 1: Operator Caused Schedule Conflicts

Condition did not appear to have a significant effect on the number of times this error was committed, although the mean number of

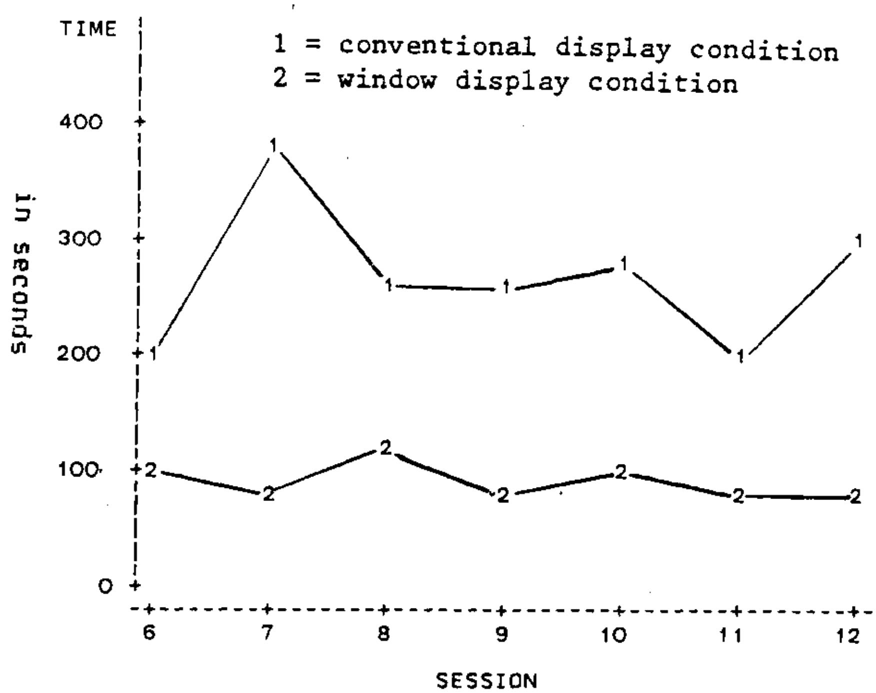


Figure 4.8a Mean Time to Configure Support Requests per Session by Display Condition.

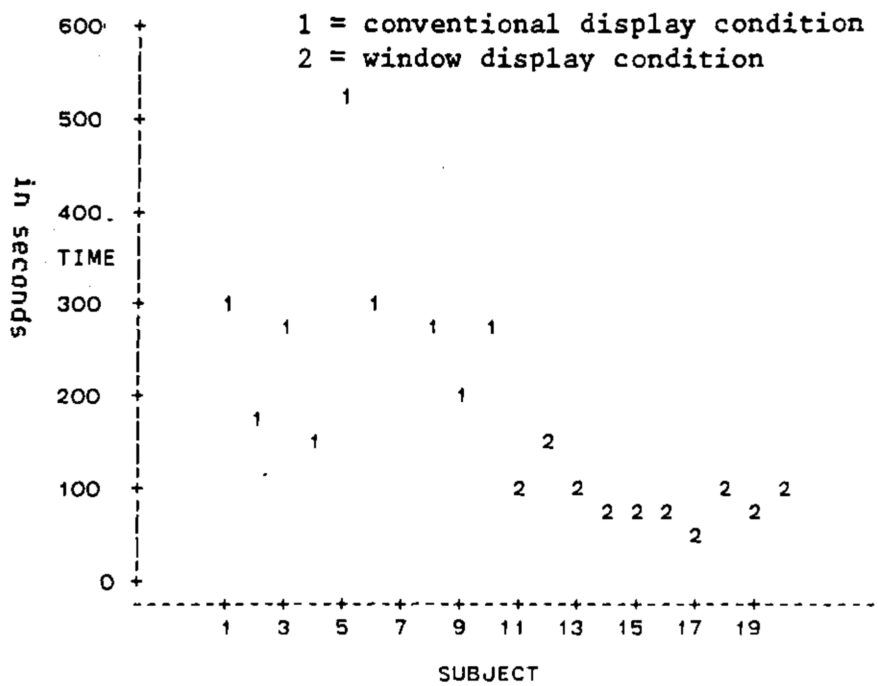


Figure 4.8b Mean Time to Configure Support Requests per Subject within Display Condition.

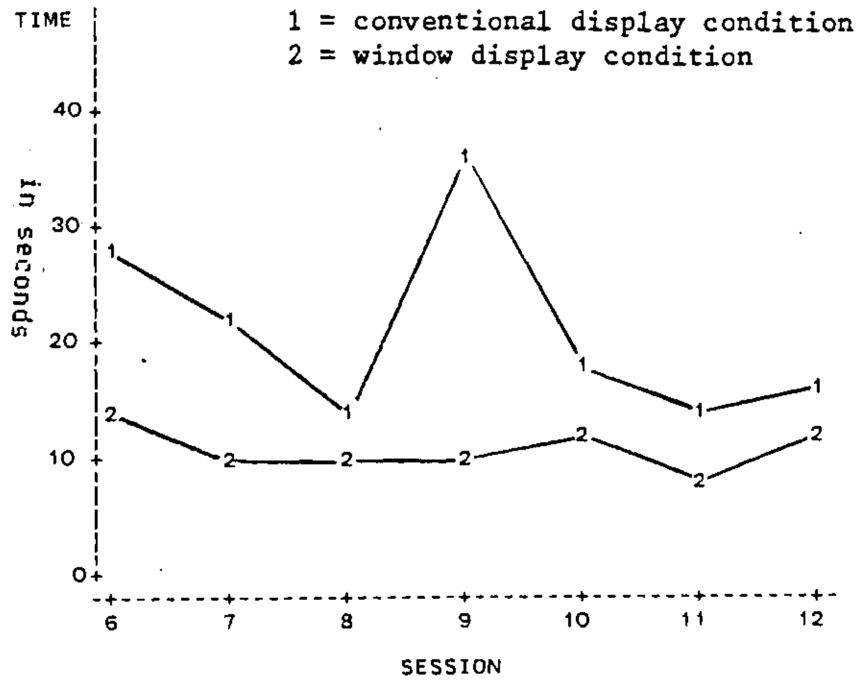


Figure 4.9a Mean Time to Manually Deconfigure per Sess- Session by Display Condition.

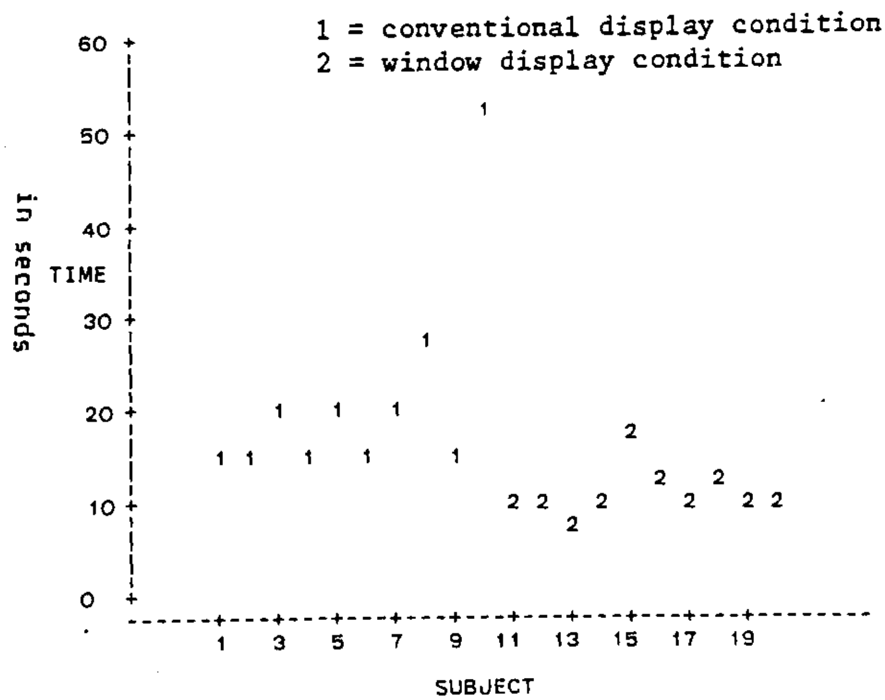


Figure 4.9b Mean Time to Manually Deconfigure per Sess- Subject within Display Condition.

operator caused schedule conflicts was somewhat less in the window display condition (.60 errors per session) than in the conventional display condition (.96 errors). The main effect of session also was not significant. The other effects did influence the occurrence of operator errors. Condition x session, $F(6,108) = 5.19$, $p < .001$, realized a high level of significance. Differences between individuals were also significant, subject (condition), $F(18,108) = 2.04$, $p < .01$. Since only one data point was collected per subject for each session, this was not a repeated measures design. Thus, the higher order effect of session x subject (condition) interaction can not be analyzed.

The measure of operator caused conflicts was reanalyzed after system errors were distinguished from true operator errors and system induced errors were removed. The reason for the reanalysis is explained in detail in the discussion section. Reanalysis shows that condition, $F(1.12,13.10) = 4.68$, $p < .05$ and condition x session, $F(6,108) = 2.72$, $p < .02$ were significant. Session and subject (condition) did not influence the occurrence of this operator error. For every session the number of scheduling conflicts caused by each subject was recored. Figure 4.10a shows the mean number of operator caused scheduled conflicts per session. Data points in Figure 4.10b reflect the mean number of scheduling conflicts each subject caused. Figures 4.10a and 4.10b and information contained in Tables 4.3 and 4.4 reflect results from the reanalysis.

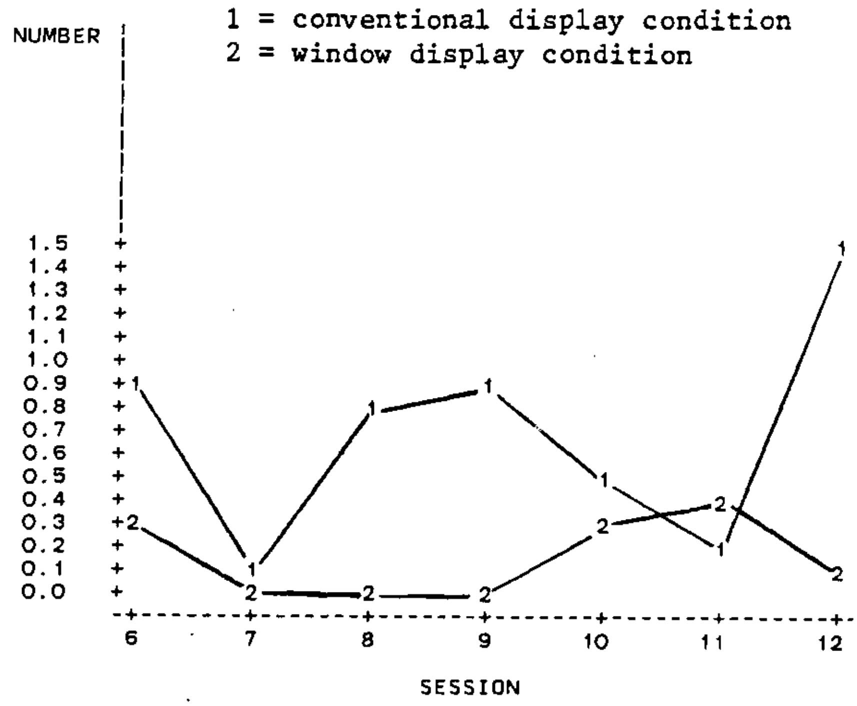


Figure 4.10a Mean Number of Operator Caused Schedule Conflicts per Session by Display Condition.

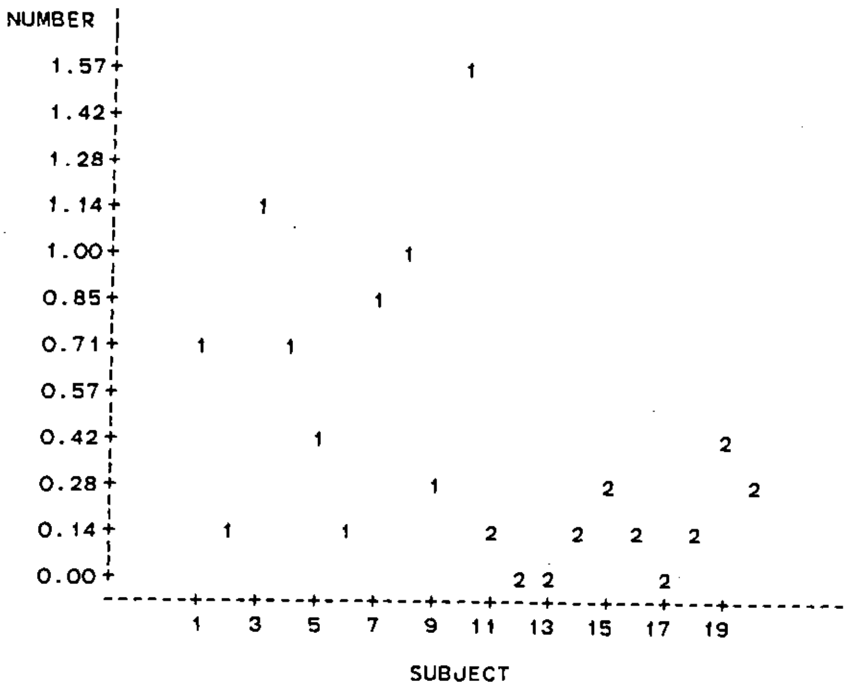


Figure 4.10b Mean Number of Operator Caused Schedule Conflicts per Subject within Display Condition.

Operator Error 2: Unnecessary Equipment Replacements

The main effect of condition was significant when occurrence of the second type of operator error was analyzed, $F(1.05,23.97) = 15.93$, $p < .01$. As expected, subjects using the conventional interface replaced operational equipment significantly more times per session (1.13 errors versus .23 errors). The only other effect to achieve significance was subject (condition), $F(18,108) = 1.79$, $p < .04$. The effect of session did not quite reach the significance level, $F(6,108) = 3.93$, $p < .06$. No significant interaction between condition and session was discovered. Figures 4.11a and 4.11b show plots of means across session and subject, respectively.

Significance levels for all effects on each dependent measure are qualitatively summarized in Table 4.3. The means and standard deviations for the display conditions are provided in Table 4.4.

Subject Reactions

At the completion of the last experimental session subjects were given a questionnaire. A copy of the questionnaire is included in Appendix D. Subjects were asked to define easy and difficult control tasks as well as good and poor system attributes.

Subjects first defined difficult operator control tasks. In the window display condition, subjects found it hard to configure equipment for unscheduled support and to monitor error blocks. In the conventional display condition subjects also stated that these tasks were problematic. In addition subjects who used the conventional

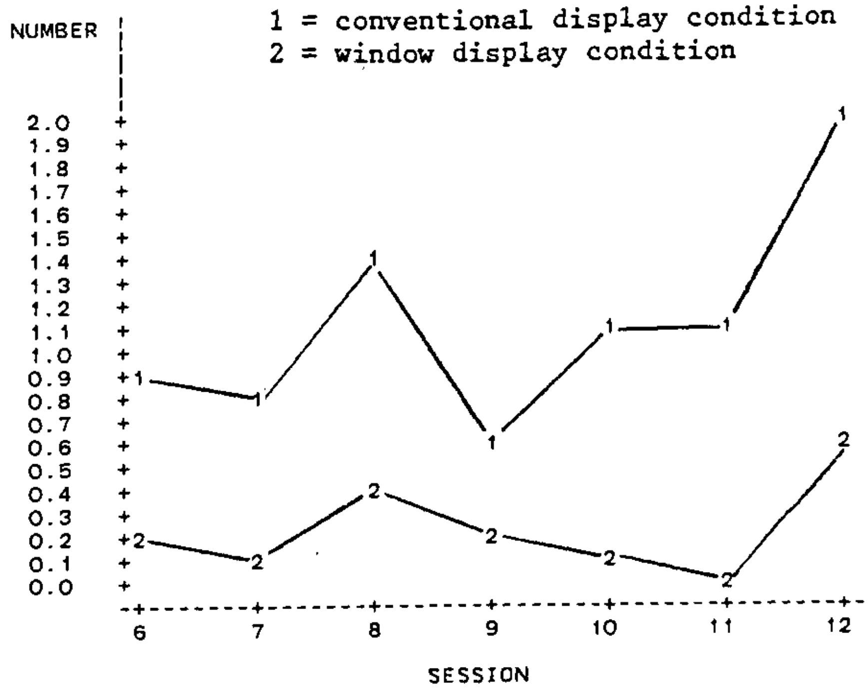


Figure 4.11a Mean Number of Unnecessary Equipment Replacements per Session by Display Condition.

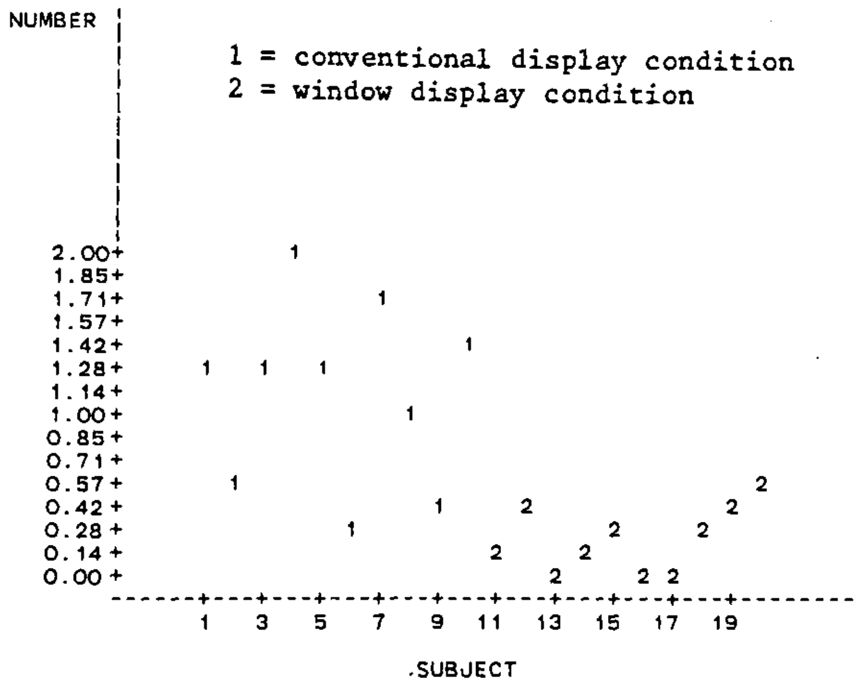


Figure 4.11b Mean Number of Unnecessary Equipment Replacements per Subject within Display Condition.

Table 4.3
Significance Levels for All Effects on Each
Performance Measure

	Condition	Session	Subj(Cond)	Cond x Sess	Sess x Subj(Cond)
Hardware Failure	–	–	–	**	–
Software Failure 1	**	–	–	***	–
Software Failure 2	***	–	**	***	–
Software Failure 3	***	–	***	***	–
Schedule Conflicts	–	**	***	*	–
Correct Responses	***	–	–	–	N/A
Time to Respond	***	–	***	–	–
Configure Requests	***	–	***	–	–
Deconfigure	**	–	–	–	***
Operator Error 1	*	–	–	**	N/A
Operator Error 2	***	–	*	–	N/A

– denotes a nonsignificant effect
* denotes $p < .05$
** denotes $p < .025$
*** denotes $p < .01$

Table 4.4
Means and Standard Deviations for the Effect of Condition on Each Performance Measure

	Conventional Display Condition		Window Display Condition		Units
	Mean	Std. Dev.	Mean	Std. Dev.	
Hardware Failure	56.42	64.04	42.45	25.62	seconds
Software Failure 1	312.44	361.16	56.95	32.51	seconds
Software Failure 2	398.91	192.90	71.19	60.66	seconds
Software Failure 3	356.73	181.66	206.02	111.92	seconds
Schedule Conflicts	75.89	56.77	46.53	37.66	seconds
Correct Responses	2.14	0.87	2.69	0.50	per session
Time to Respond	199.31	159.09	62.97	41.62	seconds
Configure Requests	264.93	114.10	90.61	52.50	seconds
Deconfigure	21.86	52.25	10.99	10.94	seconds
Operator Error 1	0.70	1.07	0.16	0.40	per session
Operator Error 2	1.13	1.18	0.23	0.54	per session

interface had difficulty monitoring data block counts for software failures, detecting the source of software problems and replacing faulty equipment.

On the questionnaire, subjects noted attributes of the full screen or window environment which made them difficult to use. Subjects using the conventional displays felt that unused information contained in the displays slowed the time it took to find relevant information. As one subject stated, "the unused information displayed clutters the screen". In the conventional display condition, most subjects wrote that they had difficulty switching between screens and integrating information, "it's tough to analyze six different screens to get one fact."

Improvements to the window environment were suggested by subjects in the window display condition. Subjects suggested having a command that erased all user defined windows except the event log and the overall system schedule. Apparently many subjects displayed these windows continually and were forced to recall them every time they issued the command to erase all windows. One subject said the overall schedule obstructed other windows, but indicated this was due to its size, not positioning. This subject felt accomplishing system tasks was facilitated by windows of the same type appearing in the same place each time, but she would have liked control of repositioning windows once they appeared.

Whereas most subjects who worked with the conventional interface found monitoring and detecting equipment for decreased rate of data flow difficult, most subjects in the window display stated that these were easy tasks. One subject wrote, "all you had to do was glance at them (spigot icons) and you could tell if something was wrong." Another stated, "the idea of faucets pouring in information was very easy to understand, it made the whole thing easy." In addition, subjects in the window display condition found replacing equipment easy, "because the computer did most of the work for you."

In both display conditions, but more so in the window display condition, subjects stated that equipment deconfiguration was an easy task.

Subjects who used the conventional interface found replacing certain types of equipment easy, although many subjects stated that in general equipment replacement was difficult. A few mentioned that the overall schedule was useful when attempting to find replacement components, "the MSOCC schedule gives a list of all (scheduled) equipment so you don't have to switch screens to look up equipment availability." Most subjects in the conventional display condition felt the dedicated, graphics configuration and status page made controlling the system easier. They felt that color coded equipment made the operator tasks easier, and they liked how the display provided an overview of the current allocation of all GT-MSOCC equipment.

The general reactions to the system indicate that subjects in both conditions felt that operating the system was difficult when a number of tasks demanded attention simultaneously. In the conventional display condition some subjects felt overwhelmed by this situation, "at times the load was too much to expect from the operator." Others enjoyed it, "when things got busy it gave you a little challenge which was fun." Comments from subjects indicate that the workload may have been less in the window display condition. Subjects using the window interface stated that although a busy system was more difficult to operate, the task was not always demanding. One wrote, "There were times when I was busy for 10 minutes straight, there were also times when I could sit back for 10 minutes." A subject who used the conventional interface gave a different account of the system operation stating that "there was not much free time between commands".

Subjects in both display conditions said they enjoyed participating in the experiment but felt that this would be a monotonous full time job.

Discussion

This section presents a more detailed discussion and interpretation of the experimental results.

The Effect of Condition on Performance

The results obtained from the statistical analyses indicate that condition is a major determinant of operator performance. Most analyses of the dependent variables showed that the operators controlled the GT-MSOCC system more effectively using the window-based interface. When the effect of condition was not significant, all the mean performance times followed the same trend, i.e., superior performance in the window display condition.

Condition did not significantly affect time to compensate for hardware failures, although the window display condition produced a lower mean time (42.5 seconds versus 56.4 seconds). Compensation for a hardware failure is comprised of two activities, i. e., identifying the faulty component and replacing it. The conventional interface provided the operator a simple means for carrying out the first of these activities. When a piece of equipment had a hardware failure, a representation of the component turned to red on the dedicated status and configuration page (Figure 2.2). Unfortunately, no simple means was provided for finding replacement hardware. In the conventional display condition, information from at least three pages on three terminals had to be integrated to identify a candidate replacement component. It is possible that both conditions provided adequate displays for hardware failure detection; aiding windows in the window display condition, however, may have provided a quicker means for identifying replacements.

Software failures were more subtle system malfunctions, where equipment was still operational but was degrading the data flow or quality. Subjects in the conventional display condition inferred data flow and quality from monitoring updating data and error block counts in order to diagnose software problems. Information was integrated from schedule and equipment status displays to identify replacement components. Subjects assigned to the window display condition compensated for software failures more rapidly. They used dynamic icons for fault identification and aiding windows to determine candidate replacements.

The time to compensate for a scheduling conflict was not significantly affected by condition, although it took longer to compensate in the conventional display condition (75.9 seconds versus 46.5 seconds). A possible explanation for this is that subjects using the traditional interface were more aware of potential problems and were more likely to plan for their occurrence. The traditional interface displayed a dedicated page that color coded failed equipment in red. With the window interface, this information was embedded in seven status windows.

Subjects in both conditions were forced to refer to status and schedule information for planning. Subjects who used the traditional interface were more accustomed to data retrieval from schedules. It is possible that in the analysis of time to compensate for scheduling conflicts, the beneficial effects of online aiding in the window display condition was dampened by planning that occurred in the conventional display condition.

On the next measure, number of correct responses to requests for unscheudled support, subjects who used the window interface answered significantly more queries correctly. These subjects also took less time to respond requests, and when support was feasible, they configured equipment for the unscheduled contacts more rapidly.

A somewhat unexpected result was that subjects in the window display condition took significantly less time to deconfigure equipment. Both groups received the same message to deconfigure an equipment string (e.g., ERBE support ended: deconfigure manually) and both groups were simply required to enter a command freeing equipment (e.g., Deconfigure ERBE). Subjects were also told that deconfiguration should preempt any current operator task. In spite of this, subjects in the window display condition (mean = 11.1 seconds) took significantly less time than those in the conventional display condition (mean = 22.4 seconds) to deconfigure equipment.

One explanation is that subjects using the conventional interface had a heavier workload, thereby causing these individuals to fail to acknowledge or delay in responding to the message to deconfigure. Another possible explanation is that in the window display condition the mission icon with an empty time bar and terminated data flow acted as a secondary alert that a satellite had completed transmission.

Condition did not have a significant effect upon the number of operator caused schedule conflicts. This operator error can be committed in two ways. Either the operator allocates a scheduled component to another mission, or else the operator neglects to deconfigure equipment before it is scheduled to support another spacecraft contact.

An examination of GT-MSOCC system events showed that in some instances time between spacecraft contacts was so brief that operators were not provided a reasonable period to deconfigure equipment before it was again scheduled. To discount these instances, when an operator had 15 seconds or less to deconfigure system configured equipment, a resulting scheduling conflict was deemed as system induced. In all cases, when the system scheduled a component on two consecutive missions, the first of which required manual deconfiguration, the contacts were scheduled either under 15 seconds apart or over 60 seconds apart. Thus, fifteen seconds was chosen as the break between system induced and operator induced errors.

After errors induced by the system were separated from actual operator errors, the influence of condition on operator caused schedule conflict was reanalyzed. The reanalysis showed condition to have a significant effect on the occurrence of operator errors, $F(1.12,13.10) = 4.68$, $p < .05$. The mean number of operator caused schedule conflicts per session was .70 in the conventional display condition and .16 in the window display condition. Poor equipment scheduling by the system was responsible for a number of scheduling conflicts. Had this not been the case, operators using the window interface would have caused significantly less schedule conflicts.

Subjects working with the conventional interface more often misinterpreted components as faulty and replaced normally functioning equipment. This suggests that the updating alphanumeric data and error block counts on full screen displays in the conventional interface were problematic for operators to interpret. Fault detection was more accurate using the dynamic icons that represented data transmission through equipment.

Summary. The effects of display condition on the performance measures can be summarized as follows:

1. Condition did not significantly influence time to compensate for hardware failures, although the mean time was lower in the window display condition. One interpretation is that the status and configuration page in the conventional interface served as an adequate means for diagnosing hardware failures.

2. The window interface enabled subjects to compensate for software failures more rapidly.

3. Although the trend was for better performance in the window display condition, there was no significant difference in time to compensate for scheduling problems. Perhaps this was because subjects using the conventional interface planned for these events.

4. In the window display condition, subjects' responses to requests for unscheduled support were correct more often. These subjects responded to queries more quickly, and they took less time to configure equipment for the satellite contacts.

5. Subjects in the window display condition more quickly deconfigured equipment, thereby freeing it for other use.

6. Performance in the window display condition was more accurate. Operators caused fewer scheduling conflicts and were less likely to replace normally functioning equipment.

The Effect of Session on Performance

The main effect of session was nonsignificant on every measure except compensation for scheduling conflicts. For this measure, the mean of the first experimental session was significantly higher than every other mean, suggesting that even after five training sessions subjects were having difficulty implementing this command. On every other measure the influence of session was negligible. However, the session x condition interaction was significant on seven of eleven measures. This indicates that the difference between the means of the two display conditions was not consistent across sessions.

Examination of plots of display conditions across sessions is necessary to determine why the two display conditions did not differ consistently across sessions. Plots of the means over sessions for the two display conditions show that in many cases performance in the window display condition was stable across sessions and performance in the conventional display interface was not. Of the eleven dependent measures, five showed uniform performance across sessions in the window display condition. This is apperant in Figures 4.2a, 4.3a, 4.7a, 4.8a and 4.9a which depict the across session means for time to compensate for components with no data flow, time to compensate for components with partial data flow rate, time to respond to requests for unscheduled support, time to configure unscheduled contacts, and time to deconfigure equipment. Since the means varied in one condition and not the other, the difference between display condition means was not consistent across sessions, creating for some of these measures (represented in Figures 4.2a and 4.3a) a significant session x condition interaction effect. These plots show graphically that performance was not only better in the window display condition, but also more stable. These results suggest that for subjects using the conventional interface, speed and accuracy on these measures vary depending on the series of system events.

On the measures time to compensate for high error block count (Figure 4.4a), number of operator caused schedule conflicts (Figure 4.10a) and number of unnecessary replacements (Figure 4.11a) across session performance within the window displau condition is more variable than with the previous five measures. However, across session performance still appears to be less variable than in the conventional display condition.

The three remaining measures show comparable variability across session in the two display conditions. These three measures are time to compensate for harware failures (Figure 4.1a shows comparable across session variability between the two conditions in the last four sessions), time to compensate for scheduling conflicts (Figure 4.5a), and number of correct responses to requests for unscheudled support (Figure 4.6a).

Examination of plots of mean performace scores across session by display condition suggests that the window display condition induced more stable performance across sessions. The measures on which performance in the window display condition exhibits variability may indicate a need to improve some aspects of the window interface.

The Effect of Subject on Performance

Individual performance was significantly different on six of eleven dependent measures (Table 4.3). The effect of subject did not significantly influence performance on the easier tasks such as deconfiguring equipment, compensating for hardware failures, and detecting terminated data flow. Individual differences did influence performance on compensation for the more subtle sofware failures and influenced the number of operator errors committed. Differences may be due to individual talent or to the strategies developed by subjects. Figures 4.1b to 4.11b provide plots of subject performance across sessions in the two display conditions. For many of the measures performance in the window display condition was stable across subjects, whereas the conventional display condition produced variable performance across subjects. This finding suggests that in the conventional display condition selection and training of the GT-MSOCC operator are important considerations.

Conclusions

In conclusion, the effect of display condition was significant for the majority of dependent measures. When display condition did influence operator performance, operators using the window-based interface consistently produced better performance. The two dependent measures that were not significantly influenced by display condition were the time to compensate for a hardware failure and the time to compensate for an automated schedule conflict. Although these were nonsignificant dependent measures, their means indicate the same trend: better performance in the window display condition.

On many dependent measures not only was performance better in the window display condition, it was also more predictable (i.e., stable performance across session and subjects). On some measures, however, the window display condition did not lead to significantly better performance, and on others there was variability across sessions or subjects. These findings may reflect flaws in implementing the interface design. In the next chapter, potential reasons for these results are discussed and improvements to the GT-MSOCC interface are suggested. Implementing the modifications may result in improved performance and reduced variability across sessions and subjects in the window display condition.

CHAPTER V

CONCLUSIONS

The preceding chapter found display condition to be a significant determinant of operator performance in monitoring and controlling a supervisory control system. Subjects using the window interface controlled the system more effectively and made fewer errors than the subjects using the conventional interface. In this chapter, modifications to improve the GT-MSOCC window interface further are recommended. Measures on which there was high variability in the window display condition across subjects or across sessions are examined as potentially problematic design areas. The measures on which performance was not significantly different in the two display conditions are also re-examined. This examination is used to suggest possible modifications to the GT-MSOCC interface which is followed by suggestions for further research. Finally, the use of such an interface for real-world supervisory control systems is considered.

Improvements to the GT-MSOCC Interface

The window display condition was not significantly better than the conventional display condition for two of the eleven performance measures (i.e., compensation for hardware failures and compensation for automated schedule problems). Figure 4.1a shows that, in the last four experimental sessions, mean time to compensate for hardware failures was comparable in the two display conditions. When a hardware failure occurred, an operator using the conventional interface was immediately notified; an icon representing the failure turned red. In the window display condition, however, hardware failures were no easier to detect than the more subtle software problems. Whether the problem was caused by hardware or software malfunctioning, initially the symptom was the same: decreased flow rate on the mission icon. Instead, the mission icon in the window display condition could be modified to indicate more clearly that a hardware failure had occurred. When a component failed, the mission icon for the satellite supporting the component could turn red. This would provide an obvious mechanism to inform the operator that a component within the mission configuration string was failed.

The second measure on which condition failed to be significant was time to compensate for automated schedule problems. Some subjects in the window display condition did not issue the correct command (HELP CONFIGURE) to access the set of needed windows. Instead, they used a series of HELP REPLACE commands to access each required window. A simple modification would have the window interface reject the HELP REPLACE command when a component was scheduled but not in use. Thus, the system would reject the inappropriate command and encourage the operator to use the correct command.

Another modification that might decrease time to compensate for automated schedule problems in the window interface would be to provide the operators aid in planning for these situations. The information that should be provided to the operator is specified in the GT-MSOCC operator model within the plan to compensate for known future problems function (Figure 3.2f). A "HELP PLAN" command could be implemented to inform the operator which components were currently failed, when failed equipment was likely to cause scheduling conflicts, and what equipment was available to serve as replacements.

Although time to compensate for a high rate of error blocks was significantly less in the window display condition, operators did not correct this software problem as quickly as they did other software failure. Mean times to compensate for the three types of software failures are shown in Table 4.4. This was probably due to the manner in which the mission icon in the window interface represented error block transmission. The other software failures were detected by observing data transmission that was represented as rate of change. When a problem occurred, the decreased rate was immediately noticeable. Error block transmission, on the other hand, was represented as the total amount accumulated. Thus, an error transmission problem was apparent only after several updates with high error block production. This result may indicate that the representation of error block transmission would be better when depicted as rate of change on the dynamic icons.

In the window display condition the number of correctly answered questions showed variability across session (Figure 4.6a) and across subject (Figure 4.6b). When an operator requested support, the windows occasionally provided information that was no longer accurate when the question was answered a few seconds later. Windows could be improved by taking into account the operator response time.

The mean number of operator caused schedule conflicts was also variable across session and subject. One reason conflict occurred was because equipment was not deconfigured before it was again scheduled. Aiding windows occasionally induced this situation by providing replacements that were unscheduled over a period that did not account for the time required to deconfigure equipment. Deconfiguration time probably should be incorporated into the algorithm that selects candidate replacement hardware.

As a final note, the underlying GT-MSOCC simulation should also be modified so that the system's automated scheduler accounts for deconfiguration time between two scheduled uses for a piece of equipment. Occasionally, due to the manner of equipment scheduling, operators were not provided with a reasonable period to deconfigure scheduled conflicts, and thus, scheduling conflicts resulted.

Future Research

This thesis provided an illustration of the implementation of intelligent windows. Windows appeared when and where they were needed and provided information required to perform the operator functions.

However, the research did not directly address a more basic question: to what degree was the window environment responsible for the facilitation of information retrieval for operators of the complex supervisory control system? A creative researcher may be able to use existing data from this experiment to separate the contributions of the operator function model, the dynamic icon display and the window environment. For example, fault compensation can be divided into two tasks: detecting faulty components and replacing them. The first task is accomplished using the dynamic icons, the second using computer windows.

Future research could directly address the question of whether the model, the window environment or both enhanced operator functioning by implementing a new full screen interface that is based on a model of the operators information needs for accomplishing control tasks. The window and full screen conditions could then be compared to provide a clearer indication of the contribution of computer windows.

Another research area is the effect of the user interface after certain system parameters are modified. The parameters of GT-MSOCC may be altered to simulate a more heavily loaded system in which equipment is scheduled to support a higher average number of concurrent missions. Other system parameters may also be altered. For example, the number of equipment failures and requests for support can be either increased or decreased. In addition, the effects of varying session lengths might be examined. It may be the case that using the conventional interface for an eight hour day would exhaust an operator due to the level of concentration and the amount of elementary calculation involved.

GT-MSOCC is a high fidelity simulation, and as such can serve as an experimental tool for applications beyond windowing technology and interface design. Given the high fidelity of the GT-MSOCC simulation, this experimental environment can serve as a tool for studying a number of other applications. For example, GT-MSOCC could be used as a testbed to study aiding devices or operator training methods.

Concluding Comments

As a high fidelity simulation, GT-MSOCC provides insight as to how a model-based window interface might improve operator performance in a real supervisory control system. The overall goal in controlling the actual MSOCC system is to maximize the amount of data captured and to ensure the quality of recorded data. The conventional NASA interface to the MSOCC system presents full screen pages of updating numbers to reflect system functioning together with pages of mission and component schedules to reflect equipment use. This research suggests that such a system induces operator error and does not facilitate quick compensation for system problems. A model-based, window interface to the MSOCC system would be likely to decrease time to detect and correct faults, and thus increase the amount of quality data captured. Results indicate correct responses to questions about the feasibility of unscheduled support are higher and that operator response time is lower with the window interface. In addition, performance across subjects and across sessions is stable. The important implication is that in a real system, operator responses to emergency support requests would be more accurate, faster and more predictable.

Experimental results apply to systems beyond the specific MSOCC satellite system. MSOCC was selected as a typical control room environment, and the results from the GT-MSOCC simulation provide strong support that operator performance can be greatly influenced by the user interface. In the wider area of supervisory control, an interface that displays task specific information contained in computer windows at the appropriate time and on a single screen may provide a superior methodology over conventional interface design.

APPENDIX A

Initial GT-MSOCC Operator Instructions

GEORGIA TECH-MULTISATELLITE OPERATIONS CONTROL CENTER: (GT-MSOCC) OPERATOR

Here at Georgia Tech we have built GT-MSOCC, a control room simulator of a NASA satellite communications system. Your job is to perform the role of the operator of this simulated control system. The GT-MSOCC operator manages the satellite communication and computer equipment needed to communicate with near-earth orbiting NASA spacecraft. Your responsibilities will be explained in detail after a brief description of the overall system.

Introduction to NASA-Godard

NASA-Goddard Space Flight Center (GSFC) currently has 16 operational satellites including Lansat, Atmospheric Explorer and Dynamic Explorer. These satellites orbit the earth gathering data about the weather, atmosphere, sun and earth. The satellites periodically transmit their scientific data or "telemetry" to an earth groundstation which in turn forwards data on to GSFC. Contact with a spacecraft is brief, and may only be made while the spacecraft's orbit is within the range of a groundstation's communication equipment. When the spacecraft passes out of range, communication is ended. Each contact with a spacecraft is called a pass; the duration of a pass is typically about ten minutes.

During a pass, the spacecraft sends data down to GSFC, and mission controllers at GSFC send back commands as well as check the overall health and safety of the spacecraft. Two types of data are transmitted by the spacecraft: telemetry data and nontelemetry data. Telemetry is science data, gathered by the scientific instruments onboard the spacecraft. Nontelemetry data are health and safety data transmitted by the spacecraft, e.g., measurements of the spacecraft's position in space and internal equipment status.

At GSFC, human operators manage and control each spacecraft. Spacecraft-specific operators work in Missions Operation Rooms (MORs). MORs are staffed 24 hours a day, seven days a week to receive telemetry, to monitor the spacecraft's status, and to issue new spacecraft commands. The fourteen MORs use a number of computers and communication systems for real time interaction with spacecraft. Although communication with each spacecraft occurs approximately twenty time in a twenty-four hour day, the duration of each pass is only about ten minutes. As a result, although MORs are spacecraft-specific, most of the computer and communications hardware that supports command and control of spacecraft are shared resources. This set of shared equipment constitutes the Multisatellite Operations Control Center (GT-MSOCC is Georgia Tech's version of this control room) and the GT-MSOCC operator supervises the GT-MSOCC support function for controllers in the MORs. The GT-MSOCC operator functions include monitoring an automated equipment scheduling and control system; manually configuring and deconfiguring computer and communication system support networks when the automated system fails; monitoring data transmission during real-time spacecraft contacts to ensure transmission continuity and data integrity; and, in the case of equipment problems, detecting and compensating for failures. The sections that follow provide details for the GT-MSOCC system functions and operator responsibilities.

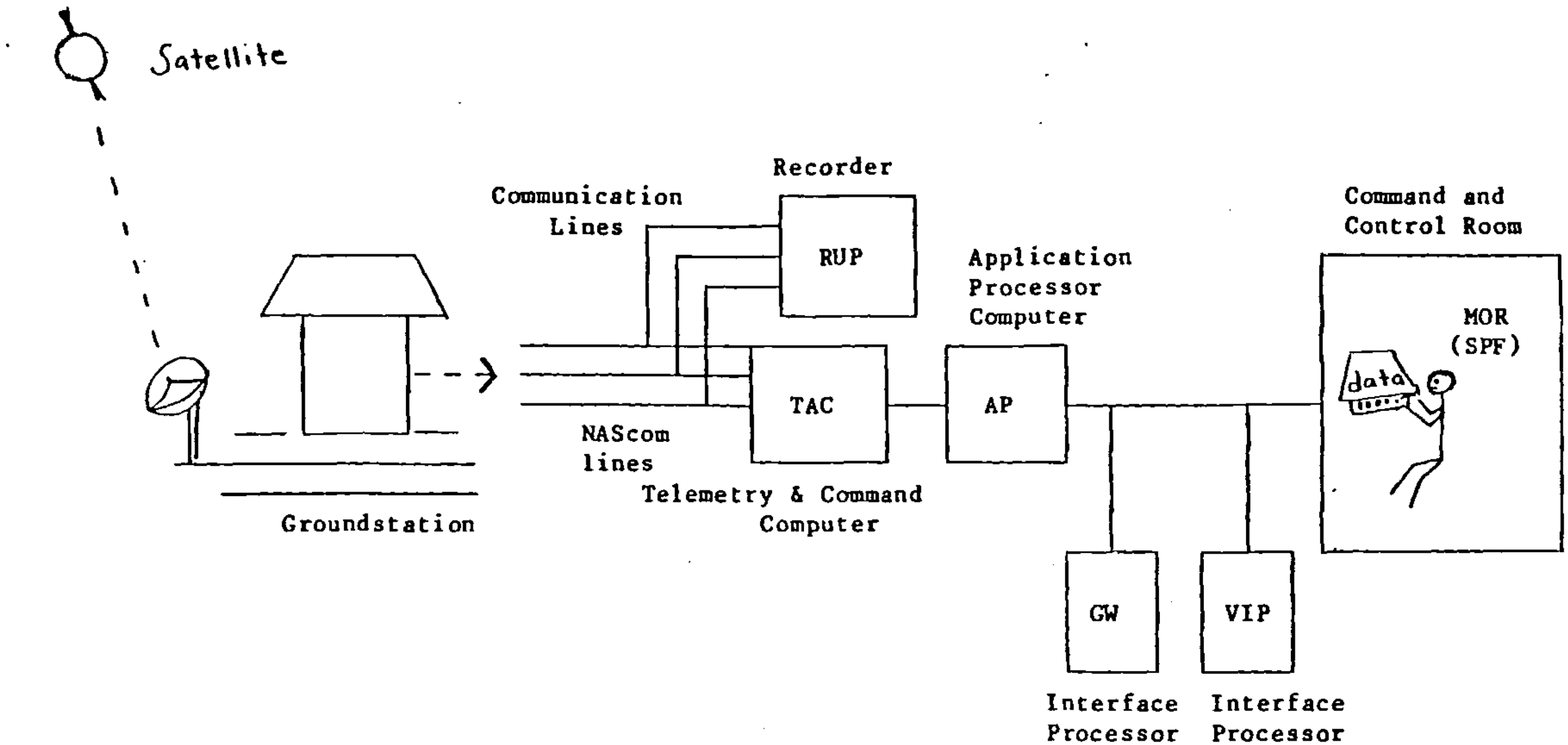
Overview of GT-MSOCC Functions and Equipment

Scheduling, configuring and deconfiguring GT-MSOCC resources are done by an automated scheduling and control system. The GT-MSOCC operator (that's you) is a supervisor who monitors this system, intervening to compensate when scheduled equipment is unavailable, equipment in use fails, or unscheduled spacecraft passes are requested.

The section below describes the various equipment types constituting the GT-MSOCC system, examples of various equipment configurations needed by spacecraft supported by the GT-MSOCC facility, the specific data flow paths through these systems, and the range of operator control functions. Detailed instructions for undertaking specific operator tasks are given in a later section.

GT-MSOCC Equipment

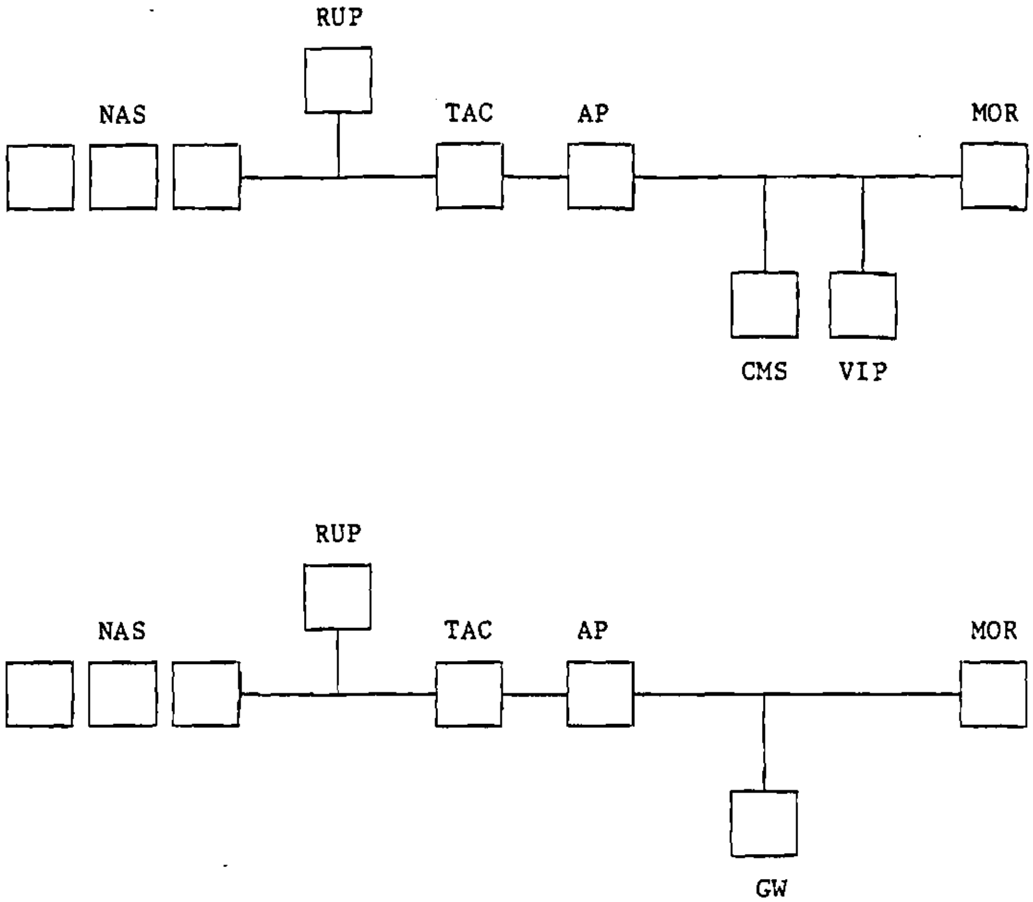
Data flow through GT-MSOCC via a series of communication lines and computers. At various points, data may be decoded, enhanced, of recorded. A typical equipment string supporting a spacecraft contact is given in Figure 1. Data arrive at GSFC via three NAScom (NAS Communication Network) lines. Arriving telemetry data are often



Typical equipment string supporting spacecraft contact.
Figure 1.

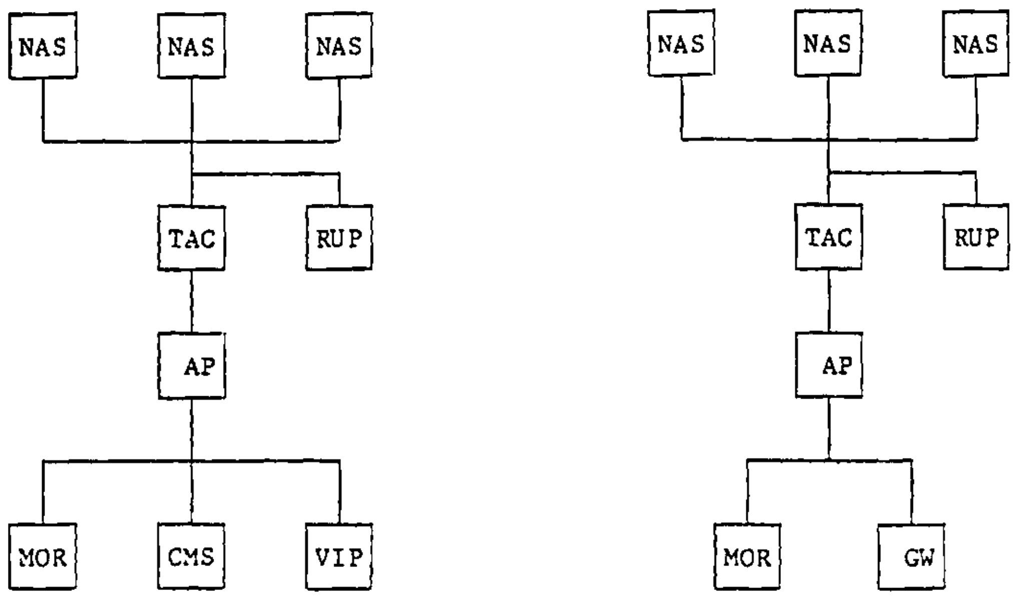
recorded, in unprocessed form, on one the RUP (recorder/utility processor) computers. The RUP processor provides a backup copy of telemetry data in the event that data integrity is compromised in subsequent processing. Arriving data are normally routed to a TAC (telemetry and command) computer for preprocessing and error checking. Data arrive in blocks that the TAC computer decodes, checks for integrity, and usually forwards to an AP (application processor) computer for additional spacecraft-specific processing. Finally, data are transmitted to the MOR where spacecraft controllers monitor the quality of incoming telemetry as well as the status of the spacecraft itself.

In addition to the MOR, portions of incoming data may be sent to several other systems. These include a CMS (commands management system), a VIP (virtual interface processor), and a GW (gateway network). The command management system (CMS) receives data blocks that confirm that the spacecraft computer has received commands transmitted to it from the MOR. The VIP (virtual interface processor) is a computer that converts data blocks from AP format to a format usable by another computer. Finally, the gateway network (GW) is a switching computer that routes portions of processed data to other NASA divisions. Typical configurations using these components are given in Figure 2. Inspection of the figure shows that the MOR, RUP, CMS, VIP, and GW are all terminal points in the GT-MSOCC incoming data flow.



Typical Equipment Configurations to Support Satellite Contacts
Figure 2

Note: figure is for the convention display condition.



Typical Equipment Configurations to Support Satellite Contacts
Figure 2

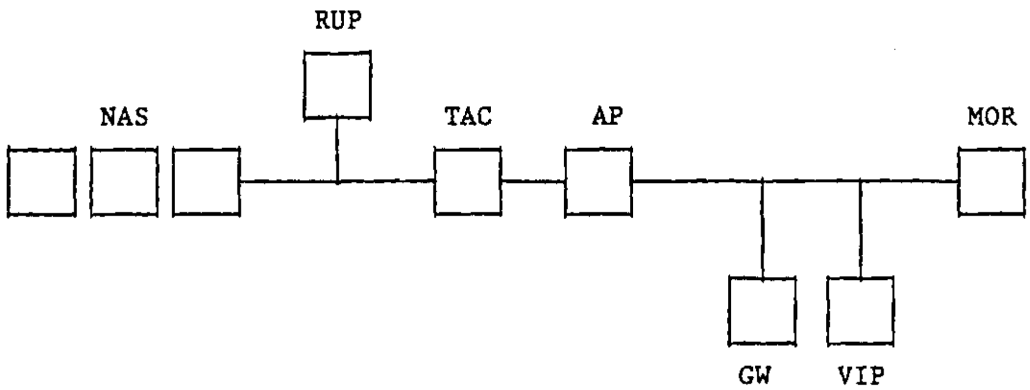
Note: figure is for the window display condition.

GT-MSOCC Data

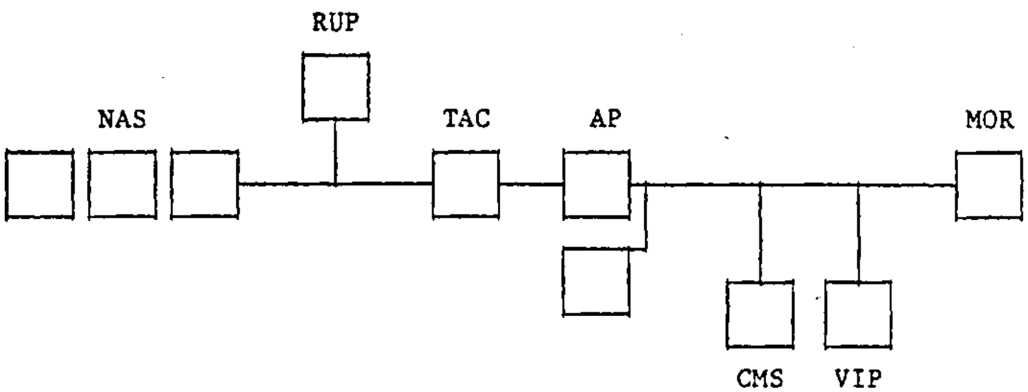
All GT-MSOCC equipment contains computers that log the amount and quality of incoming and outgoing data as they arrive at the individual piece of equipment. Thus, during a pass, at each piece of equipment supporting the spacecraft contact, there are counts of total numbers of telemetry and nontelemetry blocks transmitted, total telemetry received, total polynomial errors detected, and total sequence errors detected. The polynomial and sequence error block counts are measures of data quality. A polynomial error block is caused by problems with integrity within a data block. Sequence errors detect missing data blocks. Transmitted data blocks are numbered sequentially, and a sequence error occurs when the next block received is not identified by the next number in sequence.

GT-MSOCC Equipment Configurations

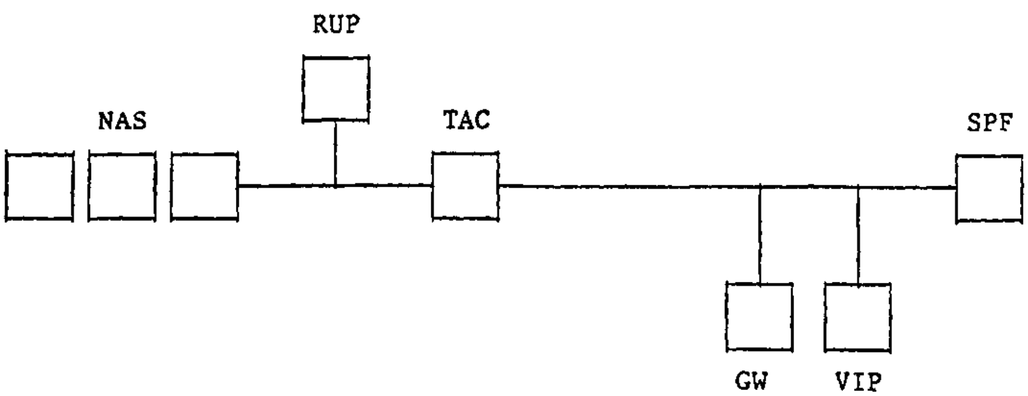
The most typical equipment configuration to support a pass is given in Figure 1. There are several alternatives, however. Some satellites require no TAC processor. Other satellites require two AP processors; some do not require any AP processor. RUP, GW, VIP, and CMS usage also vary from satellite to satellite. Alternative configurations are given in Figure 3. A total of seventeen missions are supported by GT-MSOCC, sixteen satellites plus voice communications for the Space Shuttle. A list of these missions is given in Table 1.



(a) Standard Configuration



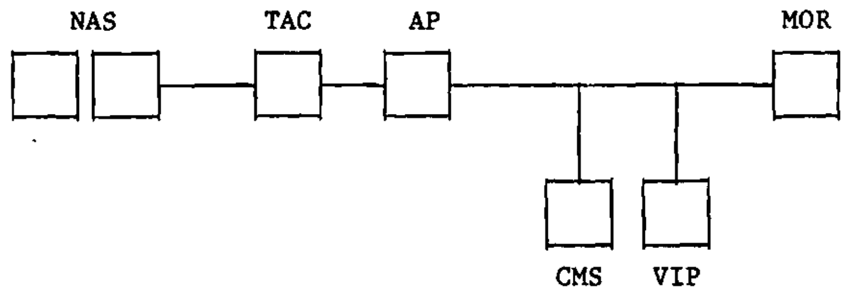
(b) Two APs, a CMS and no GW



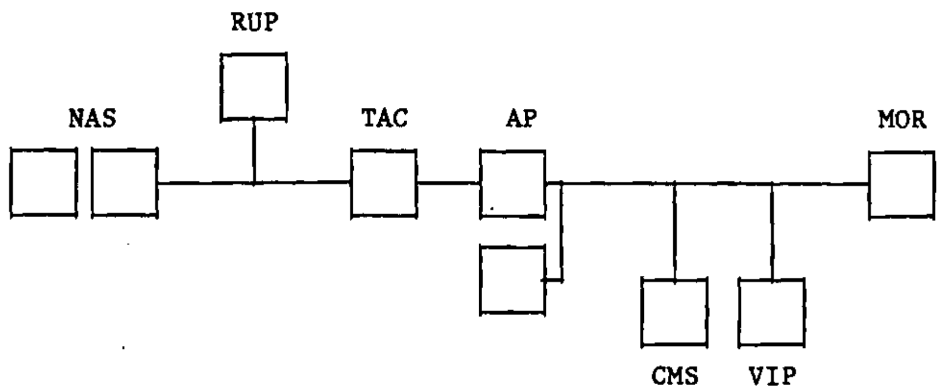
(c) SPF rather MOR, no AP

Possible Equipment Configurations to Support Satellite Contacts
Figure 3

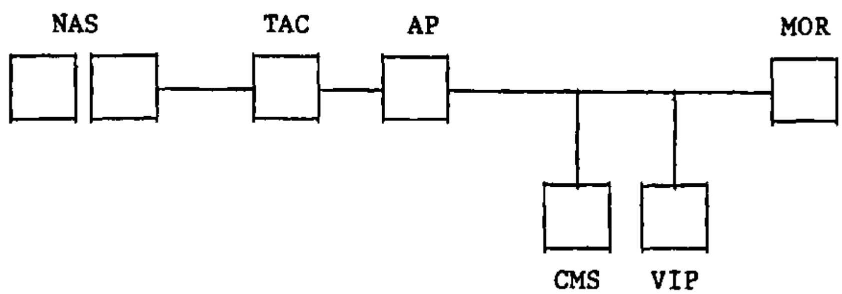
Note: figure is for the convention display condition.



(d) 2 NAS lines, no RUP, a CMS and no GW

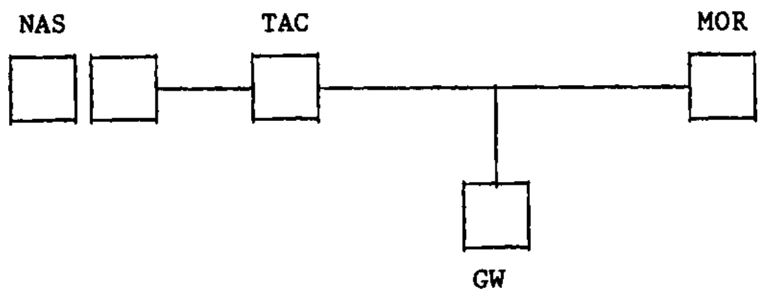


(e) 2 NAS lines and 2 APs, a CMS and no GW

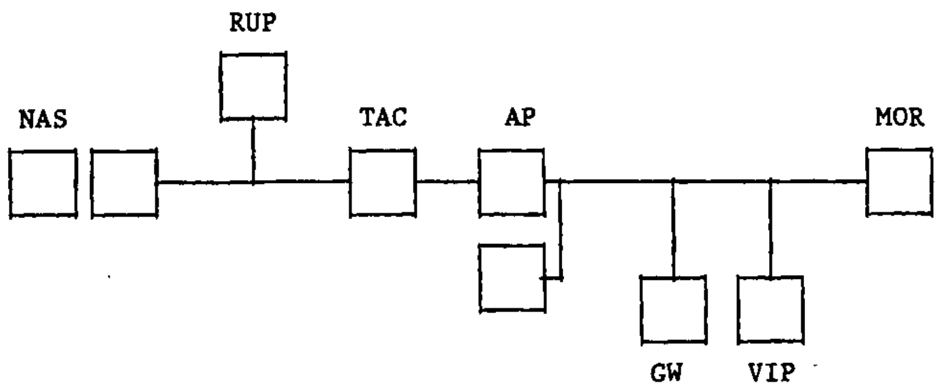


(f) 2 NAS lines, no RUP, a CMS and no GW

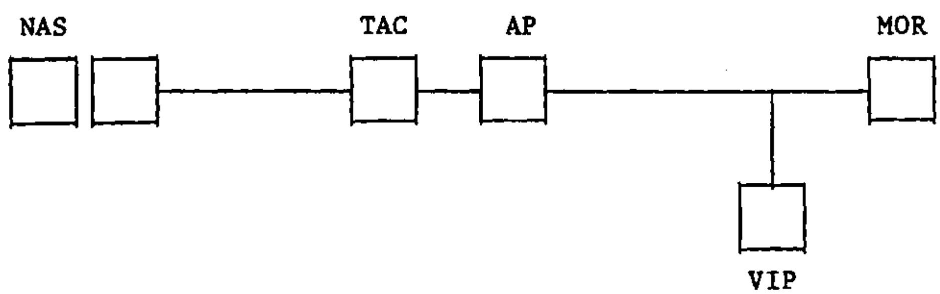
Possible Equipment Configurations to Support Satellite Contacts
Figure 3 (continued)



(g) No APs, no RUP, no VIP

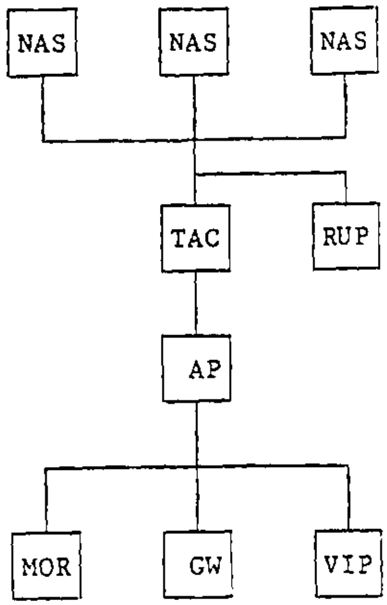


(h) 2 NAS lines, 2 APs

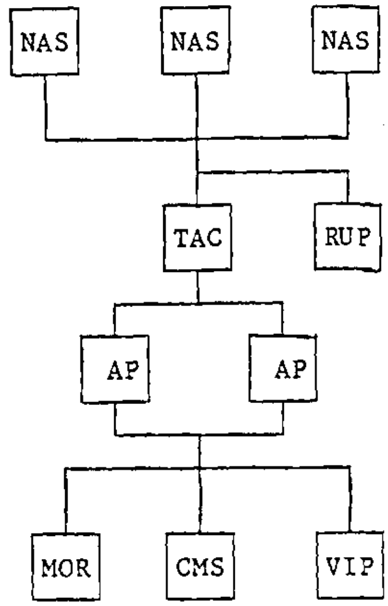


(i) 2 NAS lines, no RUP, no GW

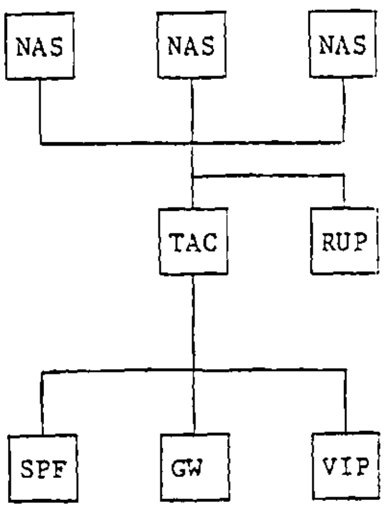
Possible Equipment Configurations to Support Satellite Contacts
Figure 3 (continued)



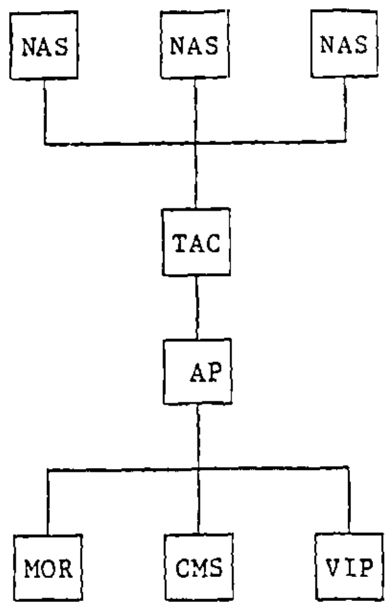
a) Standard configuration



b) Two APs, a CMS and no GW



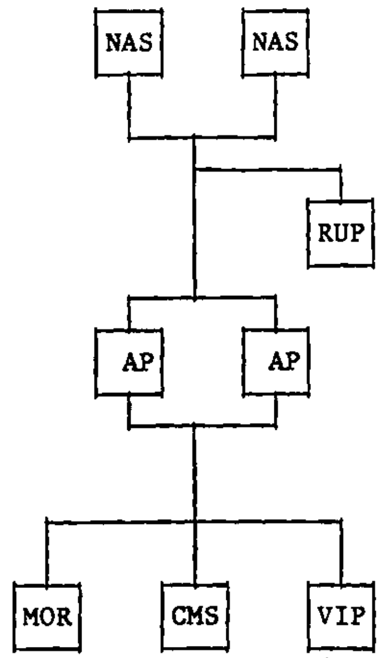
c) SPF rather than MOR, no AP



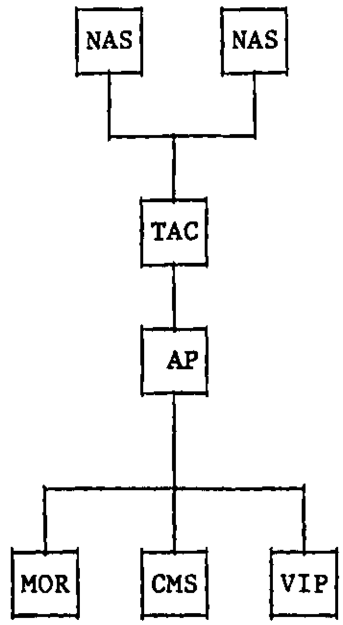
d) No RUP, a CMS and no GW

Possible Equipment Configurations to Support Satellite Contacts
Figure 3

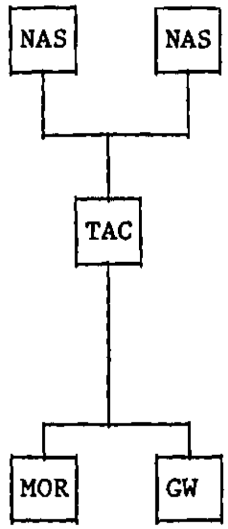
Note: figure is for the window display condition.



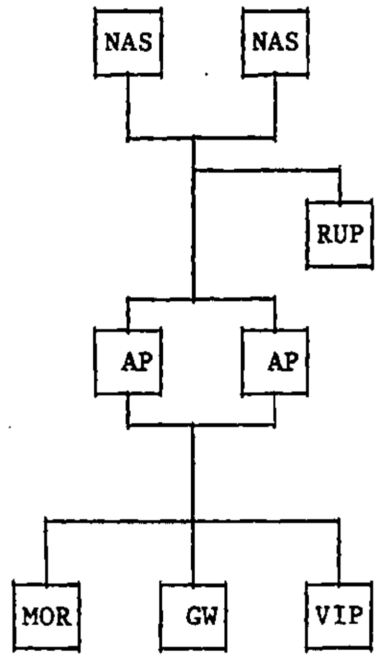
e) 2 NAS lines and 2 APs, a CMS and no GW



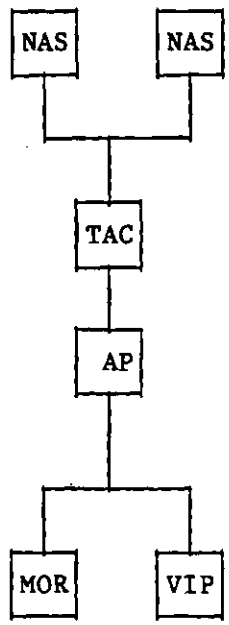
f) 2 NAS lines and no RUP, a CMS and no GW



g) No APs, no RUP, no VIP



h) 2 NAS lines, 2 APs, no TAC



i) 2 NAS lines, no RUP, no GW

Possible Equipment Configurations to Support Satellite Contacts
Figure 3 (continued)

There are several identical pieces of each GT-MSOCC equipment type. Specifically, GT-MSOCC is composed of:

- 33 NAScom lines
- 3 RUP processors
- 8 TAC processors
- 7 AP processors
- 3 VIP processors
- 2 CMS processors
- 2 GW processors
- 15 MOR rooms
- 2 SPF rooms

The last two items are not computers but are rooms that the GT-MSOCC operator links into the GT-MSOCC network via communication lines. The SPF (Shuttle payload facility) rooms are similar to MORs, but are Shuttle rather than satellite control rooms. The SPFs are used to transmit and monitor audio communications between the Space Shuttle and NASA Johnson Space Center during Shuttle missions.

Most GT-MSOCC equipment can only be used by one satellite contact at a time. In particular, NAScom lines, TACs, APs, MORs, SPFs, GWs, and CMSs can only support one user at a time. MORs and SPFs, in fact, are mission-specific, e.g., the ERBE mission always uses the same MOR and the Space Shuttle always uses a SPF. Both RUPs and VIPs, however, are multiuser devices. A single RUP can support as many as three users concurrently and a VIP can support up to two users concurrently.

GT-MSOCC Operator Control Functions

1) Supervision of spacecraft contacts currently being supported. This function has two subfunctions.

   a) Monitoring the data flow for each currently supported pass to ensure continuity and integrity of the data.

   b) In the event of problems with data flow, identifying and compensating, if possible, for equipment failures.

2) Compensation for automated schedule problems.

3) Response to requests for unscheduled spacecraft contacts.

4) Deconfigure all manually configured or reconfigured equipment strings.

The sections below provide more detail on each of these operator functions.

1. Supervisory Control of Current Spacecraft Contacts. As stated above, GT-MSOCC is an automated system. Resource scheduling, equipment configuration, and equipment deconfiguration are performed by an automated scheduling and control system. When the system is functioning as planned, e.g., there are no unscheduled passes being supported nor pieces of failed equipment, the operator's primary responsibility is to monitor the transmission of data through the GT-MSOCC equipment to ensure that data flow at the expected rate and that the quality of data is acceptable, i.e., that the error block count does not become too high. The operator performs this task by monitoring displays which provide information about the quality of data. At any given time, there may be as many as five concurrent passes, i.e., communications with up to five different spacecraft is

being supported concurrently. The GT-MSOCC operator is responsible for overseeing all of them.

During a pass, the GT-MSOCC operator monitors computer displays to detect any of several types of problems that may occur. Problems may be separated into two broad types: equipment failures and data transmission degradation. The former, equipment failure, is a situation in which a computer or communication system becomes completely inoperable. This is a fairly easy problem to detect. Failed equipment terminates the data flow recorded at the failed equipment point and affects the flow at every subsequent point in the equipment string supporting a pass. Examining Figure 1 we can see, for example, that if the TAC failed, operator displays would show that the TAC, AP, GW, VIP, and MOR were all not receiving any data. The obvious inference is that the TAC is causing the problem. After verifying this inference, the GT-MSOCC operator would then attempt to find a replacement for the failed TAC. A replacement is an available piece of equipment whose immediate use will not cause any subsequent automated schedule problems. Operators are expected to exercise a great deal of caution to avoid causing an automated schedule problem. A careless selection of replacement equipment that causes automated schedule conflicts is considered a serious operator error.

Since MORs are spacecraft-specific, a faulty MOR cannot be replaced with another. After finding a problem at an MOR, the GT-MSOCC operator sends a message to other system users reporting the problem. Similarly, if a faulty component cannot be replaced because no units are available, the operator should send a message to this effect.

The second type of problem, data transmission degradation, is much harder to detect. Although these failures may take several forms, all involve a degradation in data transmission even though individual pieces of hardware appear to be functioning adequately. Such failures may be thought of as "soft" failures. The cause may be a software problem at one of the GT-MSOCC transmission points. If a faulty GT-MSOCC equipment item is suspected, the operator is expected to attempt to verify the problem and, if possible, replace the suspect equipment.

The GT-MSOCC transmission problems are generally one of three types: full termination of data transmission, decreased transmission flow rate, and a significant increase in error block counts. The first type is full termination of data transmission. In this situation, software problems at some piece of equipment terminate data processing. As with complete equipment failure, this type of problem is comparatively easy to detect since data stop arriving at the point of transmission, thus affecting the flow at all subsequent points in the equipment string.

A related but more subtle problem is decreased rate of data flow. Given the number of NAScom lines supporting a pass, there is an expected data flow rate. A significant decrease in this rate is cause for further examination of related equipment. The operator must monitor displayed data to detect decreased flow rate problems.

If a problem is confirmed, once again the operator should replace the faulty equipment, if possible, or report the problem if no replacement units are available.

The last type of transmission problem is high error block counts in received data. A certain amount of error blocks is expected, but a rapid increase in the number of errors requires the operator to more closely examine the error propagation through the GT-MSOCC equipment string supporting the pass to see if one of the pieces of GT-MSOCC hardware is causing errors. As in previous cases, if a faulty piece of GT-MSOCC hardware is suspected, the operator is expected to replace it, if possible. If it is not possible to replace it, a message should be sent to this effect.

2. Compensation for Automated Schedule Problems. The second major control task of the GT-MSOCC operator is to compensate for automated schedule problems. The automated schedule that controls the allocation of specific pieces of GT-MSOCC equipment to specific spacecraft passes is always at least twelve hours old. As a result, recently failed equipment or equipment originally scheduled but currently supporting another mission (perhaps being used for an emergency, unscheduled spacecraft pass) is not taken into account by the automated schedule and control system. When the automated control system finds that scheduled equipment is not available to support the pass that it is attempting to configure, it sends the operator a message to that effect and makes no further attempt to configure the equipment. At this point, the equipment configuration becomes a

manual operator control task and as such is a major function of the GT-MSOCC operator.

After receiving a message that the automated control system is unable to configure a scheduled mission due to equipment unavailability, the operator is expected to identify and, if possible, replace equipment. If replacement equipment is found, the operator then manually configures the equipment for the mission. An example scenario is given in Figure 4(a). As with equipment replacement, the operator is expected to exercise caution in selecting replacement equipment and avoid causing subsequent automated schedule conflicts.

Once equipment for a pass is manually configured, it is no longer under the direction of the automatic GT-MSOCC controller. As a result, when the pass is terminated, the equipment must be manually deconfigured. The operator will receive a message stating that the pass is ended and to deconfigure the equipment manually. Prompt operator response is important, otherwise equipment will not be available for upcoming automatically schedules passes. A sample scenario for deconfiguring is given in Figure 4(b).

3. Responding to Special Requests. The third major GT-MSOCC operator function is responding to special requests for unscheduled support of spacecraft passes. Periodically, the GT-MSOCC operator receives requests to configure the required equipment in order to allow communications with a specific spacecraft, or the operator may be asked to check on the availability of needed equipment for such support. After receiving such a request, the GT-MSOCC operator is

```
System:  Unable to configure ERBE: NAS3 unavailable.

Operator:  CONFIGURE ERBE REPLACE NAS3 NAS4
```

Example of Manual Configuration
Figure 4(a)

```
System:  ERBE support ended:  deconfigure manually.

Operator:  DECONFIGURE ERBE
```

Example of Manual Deconfiguration
Figure 4(b)

expected to examine equipment schedules to determine the feasibility of unscheduled support. If it is possible to configure the equipment, the operator replies with an affirmative answer and commences to configure the necessary equipment for the spacecraft. If there is insufficient equipment available, the operator responds to the request with a negative answer.

4. Deconfiguring Manually Configured or Replaced Equipment. Once an operator has manually configured or intervened in a automatically configured equipment string, i.e., replaced a failed component for a component that was unavailable when the automated schedule tried to configure a scheduled pass, the operator must manually deconfigure the equipment at the end of the pass. This should be done promptly since until it is deconfigured the equipment is not available for other use.

Summary

This concludes the overview to the GT-MSOCC system functions. Details on displayed information and procedures the operator uses to carry out management and control functions follow.

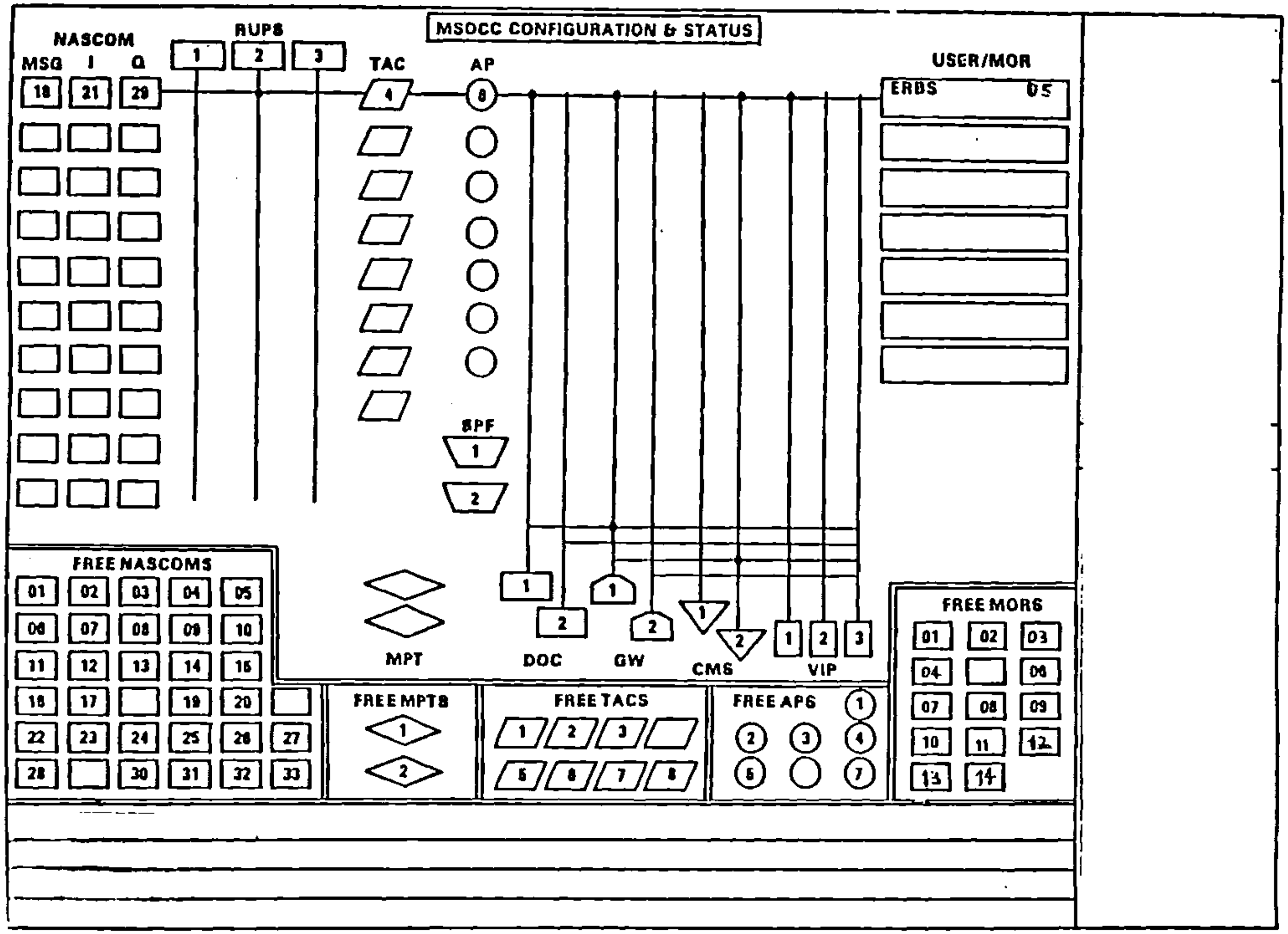
APPENDIX B

GT-MSOCC Operator Instructions for the Conventional Display Condition

GT-MSOCC OPERATOR WORKSTATION

The GT-MSOCC operator workstation consists of three CRT screens and a keyboard on which the operator enters commands and information requests. The center screen of the workstation is the GT-MSOCC Configuration and Status page. It provides the current status of all the GT-MSOCC communication and computer equipment as well as what equipment is supporting current passes. The lower portion of the screen shows the status of equipment not currently supporting a spacecraft contact. For example, in the bottom left hand corner of the page (see Figure 5) the numbered blocks show the NAScom lines not currently supporting spacecraft contacts. Similarly, other blocks show the status of TAC and AP processors and mission operations rooms (MORs). These blocks are color coded to show current status of the individual hardware items. A blue box indicates that the item is in fully operational condition and available for use. A red box indicates that the item is down and not available for use. In the top portion of the page, equipment in use and functioning correctly is coded in green. A failed component is coded in red.

The lower half of the GT-MSOCC Configuration and Status page shows the status of an additional type of equipment other than those described in the overview, two pieces of hardware that are called Mission Planning Terminals (MPTs). An MPT is an microcomputer that provides the GT-MSOCC system with the automated schedule of passes and GT-MSOCC equipment reserved to support each pass. It is an input computer to the GT-MSOCC system. The display shows the GT-MSOCC



Configuration and Status Display Page
Figure 5

operator which MPT is currently active, i.e., which MPT is providing the GT-MSOCC schedule. The MPT is not under GT-MSOCC operator control. Changes from one active MPT to the other are automatic and do not affect system function.

The upper portion of the screen shows the GT-MSOCC operator what passes are currently being supported and which pieces of GT-MSOCC equipment are supporting each pass. In the example shown in Figure 5, only the ERBS spacecraft is currently being supported. Reading across the line from left to right, the display shows that the equipment supporting ERBS consists of three NAScom lines (18, 21, 29), RUP2, TAC4, AP6, GW1, CMS2, VIP1, MOR5, and DOC1.

The last piece of equipment, the DOC, is similar to the MPT. Like the MPT, the DOC (Data Operations Computer) is not under GT-MSOCC operator control. It is the control and coordination system that supervises all GT-MSOCC equipment. The DOC receives the automated schedule from the MPT and, as long as all the reserved equipment are available, automatically configures the equipment for spacecraft contacts contained on the GT-MSOCC schedule; at the end of the pass, the DOC also automatically deconfigures equipment that it has configured. There is always one DOC processor working; since the DOC performs a critical function, system reliability requires a fully redundant backup DOC processor. Changes from one DOC processor to another are automatic and do not affect overall system operation. The configuration and status display merely provides the GT-MSOCC operator information about which DOC processor is currently active.

In addition to the center GT-MSOCC Configuration and Status page, the GT-MSOCC operator has two other CRTs available. The GT-MSOCC operator can call up a number of different pages on both the left and the right terminals. The left screen is used to display schedule information and the right screen is used to display data flow and error block count information for current passes. The schedule information displayed on the left screen includes an overall GT-MSOCC spacecraft pass schedule (e.g., DISPLAY MSOCC SCHED) as well as individual schedules for each spacecraft (e.g., DISPLAY ERBE SCHED). See Figures 6 and 7 for examples. In addition to spacecraft schedules, there are display pages with the schedule for each piece of GT-MSOCC equipment (e.g., DISPLAY TAC1 SCHED). Equipment schedules are available in alphanumeric as well as graphical form (e.g., DISPLAY TAC AVAIL will call a graphical representation of the TAC processors schedules). Figure 8 gives an example. Finally, the left screen is also used to display an events/alarm log. The events/alarm log shows all the alarm and event messages recently sent to the GT-MSOCC operator. Figure 9 provides an example of the events/alarm log page (i.e., DISPLAY EVENTS). Table 2 contains a summary of all schedule information retrieval and events/alarm log page requests. The right screen contains data block and error block counts for all equipment currently supporting passes as well as for each piece of GT-MSOCC equipment. The page most often displayed on this screen shows telemetry and nontelemetry data block counts as well as polynomial and sequence errors block counts received at the MOR for currently supported pass (e.g., DISPLAY TELEM). This page

110/18:25:47
DOC1 ONL

DAY 116-117 MSOCC SUPPORT SCHEDULE 116/00:00 TO 117/23:59

STRT	AOS	LOS	END	USER	ORBIT	STA	TYPE	LINES			EQUIPMENT
1910	1930	2010	2017	ISEE-1	03490	GWM	RT	03	08	14	RUP1 TAC1 AP3 DOC1 MOR5
1940	2000	2020	2025	ISEE-3	01578	MAD	RT	18		27	TAC4 AP2 DOC1 MOR3
1940			2100	CSC-BASL			SW				AP4 SW1
2000			2130	CSC-SPIF			SW				SPF2 MOR8
2050	2110	2157	2202	DE-1	01831	AGO	RT	01	05	14	RUP3 TAC1 AP5 GW2 DOC1 MOR7
2110	2130	2300	2305	ISEE-3	01578	EAST	RT	19	21		RUP2 TAC2 AP6 DOC1 MOR3
2205	2215	2235	2237	ATS-1	00290	WEST	RT	04	08	12	TAC3 AP1 DOC1 MOR2
2230	2240	2300	2305	ATS-5	00184	AGO	RT	22	28	32	RUP2 TAC2 AP2 DOC1 MOR1
2300			0000	CSC-GW			TST				AP1 AP7 GW1 GW2 DOC1 MOR8
											LCR1
2310	2320	2340	2342	ATS-3	00225	GDS	RT	23		33	TAC2 AP2 DOC1 MOR1
2312	2332	0117	0122	DE-1	01832	ORR	RT	01	11		RUP3 TAC5 AP5 DOC1 MOR7
2330			0020	IMP-8			PB				RUP2 TAC8 AP6 DOC1 MOR4
0000			0100	MNT-DEC			PM				TAC1
0020	0030	0120	0122	ISEE-1	03491	HAW	RT	05	09	17	TAC7 AP3 DOC1 MOR5
0040			0150	COBE			SIM				RUP1 TAC1 TAC3 TAC2 AP4
											DOC1 MR10
0100			0400	MNT-DEC			RM				AP6
0110			0230	ERBS			OFL				AP7 DOC2 MOR4

Sample MSOCC Schedule Page Called by
"DISPLAY MSOCC SCHED"
Figure 6

365/14:05:29
DOC1 ONL
DAY 001-003 AP3 SUPPORT SCHEDULE 001/00:00 TO 007/23:59

STRT	AOS	LOS	END	USER	ORBIT	STA	TYP	LINES	EQUIPMENT
1910	1920	2015	2017	ISEE-1	04360	GWM	RT	02 06 11	RUP1 TAC1 AP3 DOC1 MOR7
2020			2105	CSC-GW			SW		AP3 SW1
2110	2120	2259	2301	ISEE-3	01742	EAST	RT	18 20 33	RUP2 TAC5 AP3 DOC1 MOR3
2301			2350	ISEE-3			PB		RUP2 TAC5 AP3 DOC1 MOR3
0110	0120	0312	0314	IMP-8	00560	WEST	RT	01 02 03	RUP1 TAC1 AP3 DOC1 MOR1
0400			0500	MNT-POC			PM		RUP1 TAC5 AP3 MOR9
0510			0603	ATS-1	00322	WEST	RT	04 11 12	TAC7 AP3 DOC1 MOR1
0625			0715	COBE			SIM		RUP2 TAC4 TAC6 TAC3 AP3 DOC1 MR10
0715	0725	0836	0838	DE-1	01890	ORR	RT	15 16 17	RUP3 TAC5 AP3 DOC1 MOR7
0850			0940	CSC-GW			TST		AP3 AP7 GW1 GW2 DOC1 MOR8 LCR1
1320			1410	IMP-8			PB		RUP1 TAC7 AP3 DOC1 MOR1
1410			1610	CSC-BASL			SW		AP3 MR10
1730			1830	MNT-OPS			PM		AP3 MOR9
1850	1900	2030	2032	ISEE-1	04361	GDS	RT	03 16	TAC7 AP3 DOC1 MOR5
2020			0330	MNT-DEC			RM		AP3
0350			0440	ATS-3			OFL		AP3 DOC2 MOR4
0500	0510	0710	0712	ERBS	00006	WEST	RT		RUP1 TAC3 AP3 GW1 CMS1 DOC1

Sample Equipment Schedule for AP3
Called by "DISPLAY AP3 SCHED"
Figure 7

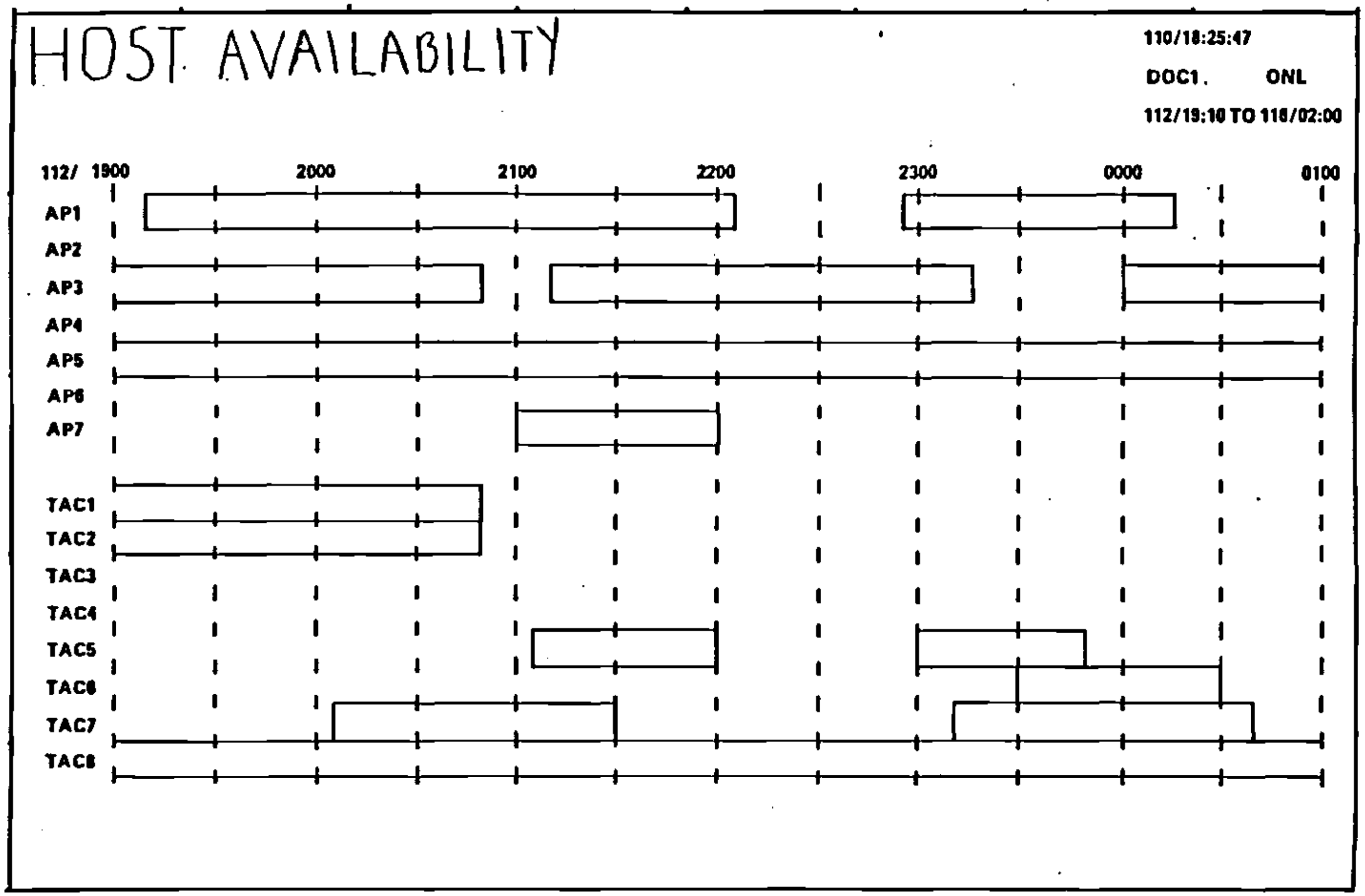
```
DISPLAY MSOCC SCHED
DISPLAY Mission-name* SCHED
DISPLAY component-name** SCHED
DISPLAY EVENTS
```

*See Table 1 for a list of mission-name abbreviations.

**See handout for a list of all component types and names.

A summary of the schedule information retrieval and events/alarm log page requests

Table 2



Sample Graphic Schedule Page Called by
"DISPLAY TAC AVAIL" or "DISPLAY AP AVAIL"
Figure 8

```
                                                                  DOC 2 ONL

                               DOCS EVENTS/ALARMS

ALARM   DATE                                  MESSAGES
-------------------------------------------------------------------------
        083:11:53:32      LNSAT configured automatically.
 **     082/11:54:21      Q 1265 Can we support ASTRO for 6 minutes?
        083:11:57:53      ERBE configured automatically.
        083:12:00:00      LNSAT support ended: deconfiguration complete.
 **     083:12:13:34      Unable to configure GEO  : NAS6 unavailable.
```

Sample Operator Events/Alarms Log Page
Called by "DISPLAY EVENTS"
Figure 9

provides three time samples of incoming data block and error block counts for each spacecraft pass currently supported. An example of this page is given in Figure 10. Secondary data block flow and error block count pages are also displayed on the right CRT screen. Summary status of several pieces of GT-MSOCC equipment is available (e.g., DISPLAY TAC TELEM). Examples are provided in Figures 11 and 12.

The other page that can be displayed on the right CRT is a list of pending missions. This is a list of recent spacecraft passes that were not able to be configured automatically (i.e., DISPLAY PENDING). In such situations, the operator attempts to manually configure the pass before the scheduled time during which contact with the spacecraft is passed. If the operator configures the pass or if the window of time during which contact with the spacecraft can be made is passed, the spacecraft is automatically deleted from the pending mission list. An example of the pending missions page is given in Figure 13. The keyboard and three CRTS with the pages described above constitute the GT-MSOCC operator interface to this system. Details for GT-MSOCC operator supervisory control procedures follow.

TELEMETRY STATUS/QUALITY 083/00:24:43

NAME	SITE	TIME DOWN	TYPE	DEST	MSID	TBR	BC	TBP	FLAGS
DSEI	MAD	083/00:28:00	RT	MAD	0077	01228	001535	00002	00013
DSEI	MAD	083/00:28:00	RT	MAD	0077	00431	000539	00000	00004
DSEI	MAD	083/00:28:00	RT	MAD	0077	00834	001043	00001	00008
PM	EAST	083/00:25:00	RT	EAST	0069	10276	012845	00039	00195
PM	EAST	083/00:25:00	RT	EAST	0069	10789	013486	00042	00214
PM	EAST	083/00:25:00	RT	EAST	0069	11435	014293	00047	00236
AE-D	ORR	083/00:30:00	PB	ORR	0079	00415	000519	00000	00004

Sample MOR Data Page Called by "DISPLAY TELEM".
There are currently three missions supported:
DSEI, PM and AE-D.
Figure 10

MODLAN PERFORMANCE

HOST	STATUS	BLKS RCVD	BLKS XMTD	RCVD ERRORS	XMTD ERRORS
DOC1	OK	4000	1406	2	1
DOC2	OK	2000	1002	0	0
AP1	NORECV	0	0	45	0
AP2	OK	200	10	0	0
AP3	OK	100	15	0	0
AP4	DOWN	0	0	0	0
AP5	OK	50	11	0	0
AP6	OK	30	10	0	0
AP7	NOSEND	46	15	0	15
GW1	OK	200	20	1	0
GW2	OK	150	10	0	1
SPF1	OK	15	2	0	0
SPF2	DOWN	0	0	0	0
CMS1	OK	30	2	0	0
CMS2	OK	100	4	0	0
VIP1	OK	15	6	0	0
VIP2	OK	20	15	0	0
VIP3	OK	40	20	0	0

Equipment Status Page Called by
"DISPLAY MODLAN TELEM"
Figure 11

AP PERFORMANCE

AP #	STATUS	S/ID	ORBIT #	TLM BLKS	TLM LOCK	N-TLM BLKS	POLY ERRS	NCC ODM BLKS	ATT BLKS	CMD SITE	GCMR BLKS	CMD BLKS
1	ONLINE	DE-1	111	76	YES	0	0	0	3	ORR	0	2
				114	YES	0	0	0	10		0	0
2	DOWN											
3	OFFLIN											
4	ONLINE	ISIS	444	4094	YES	0	1	0	0	ORR	0	2
				118	NO	0	15	0	0		0	0
5	DOWN											
6	DOWN											
7	OFFLIN											

AP Performance Page Called by
"DISPLAY AP TELEM"
Figure 12

```
                                                               083/00:58:32
                                                                  DOC2 ONL
                               PENDING MISSIONS

STRT AOS  LOS  END  USER   ORBIT  STA  TYPE  LINES    SCHEDULED EQUIPMENT
---- ---  ---  ---  ----   -----  ---  ----  -----    -------------------
0038 0039 0041 0042 AE-D   03021  ORR  PB    06 07    TAC7 GW1 MOR4
```

Sample pending mission page with
one mission, AE-D, pending called
by "DISPLAY PENDING"
Figure 13

GT-MSOCC SUPERVISORY CONTROL PROCEDURES

As stated in the introduction, the GT-MSOCC operator has three major functions:

1) Supervision of spacecraft contacts currently being supported. This function has two subfunctions.

   a) Monitoring the data flow for each currently supported pass to ensure continuity and integrity of the data.

   b) In the event of problems with data flow, identifying and compensating, if possible, for equipment failures.

2) Compensation for automated schedule problems.

3) Response to requests for unscheduled spacecraft contacts.

4) Deconfigure all manually configured or reconfigured equipment strings.

Specific procedures and examples for each follow.

1. Supervisory Control of Current Spacecraft Contacts

The GT-MSOCC operator ensures that all GT-MSOCC equipment is functioning properly so that information from a satellite reaches the MOR (or SPF) and any other terminal points in the equipment string supporting the pass, namely the RUP, CMS, GW, or VIP.

The operator monitors the currently supported spacecraft contacts for problems. On the left terminal the operator may want to call the events/alarm log page, which keeps a record of important system events and alarm messages to the operator. The command "DISPLAY EVENTS" will call the events/alarm log. On the center screen the operator monitors a continuously displayed graphics screen, indicating which mission(s) each piece of equipment is currently supporting and whether or not any component has suffered a hardware failure. The center screen is also where the operator enters commands and receives messages before they are automatically logged onto the events/alarm log. On the right terminal the operator monitors telemetry and nontelemetry blocks received and error blocks detected at each MOR (SPF) to assure adequate data flow rate and quality as data reach the MOR (SPF) terminal point of the system. The command "DISPLAY TELEM" calls up this page on the right screen.

The GT-MSOCC operator monitors equipment status to quickly detect hardware failures in the computer and communication equipment supporting a spacecraft contact. As indicated the overview, GT-MSOCC equipment problems may be separated into two broad types: equipment failures and data transmission degradation. The former, equipment failure, is a situation in which a computer or communication system becomes completely inoperable. The cause of this type of problem is typically hardware failure. A component with failed hardware is usually easy to detect. The icon representing the equipment on the Configuration and Status page turns red. In addition, hardware failure terminates the data flow at every subsequent component in the equipment string supporting a pass. The operator should, if possible, immediately replace a component with a hardware failure.

The second type of GT-MSOCC problem, a transmission failure, is much harder to detect. Although these failures may take several forms, all involve a degradation in data transmission even though individual pieces of equipment appear to be functioning adequately.

Such failures may be though of as "soft" failures. The cause is typically a software problem in the mission-specific software at one the GT-MSOCC pieces of equipment supporting the spacecraft contact.

GT-MSOCC transmission problems occur in one of three ways: a full termination of data transmission, a significant decrease in the data transmission rate, or a significant increase in error counts. The first problem results in a full termination of data transmission. In this situation, software problems at some piece of equipment terminate data processing. As with equipment hardware failures, this type of problem is comparatively easy to detect since the data stop arriving at the faulty component and affect the data flow at all subsequent points in the equipment string.

A related but more subtle problem is decreased rate of data flow. Given the number of NAScom lines supporting a pass, there is an expected data flow rate (approximately 10 blocks arrive at each NAScom line per second). A significant decrease in this rate is cause for further examination of related equipment. The operator must monitor displayed data block counts to detect decreased flow rate. If a problem is confirmed, the operator should, if possible, replace the faulty equipment.

The last type of transmission problem is high error block counts in received data. A certain amount of error is expected but a rapid increase in the number of error blocks requires the operator to more closely examine error propagation through the GT-MSOCC equipment string supporting the pass to see if one of the pieces of GT-MSOCC hardware is causing the errors.

The box below summarizes information retrieval commands used in the GT-MSOCC supervisory control function. Subsequent sections provide operator procedures to detect each of the equipment problems described above.

```
Operator Commands for Monitoring

  DISPLAY EVENTS (LEFT SCREEN)
  DISPLAY TELEM (RIGHT SCREEN)
  DISPLAY MODLAN TELEM
  DISPLAY NAS TELEM1
  DISPLAY NAS TELEM2
  DISPLAY TAC TELEM
  DISPLAY AP TELEM
  DISPLAY RUP TELEM
  DISPLAY GW TELEM
  DISPLAY CMS TELEM
  DISPLAY VIP TELEM
```

Detecting a Hardware Failure. Hardware failures can be detected on the center terminal Configuration and Status page. This screen indicates which missions are currently engaged in a pass and shows graphically which GT-MSOCC components are supporting each pass. The status of the components is color coded: blue means the component is idle, green means the component is in use and red indicates that the component has had a hardware failure. A failed component requires offline maintenance and should be replaced immediately, if possible. A failed component is unavailable for further use until it is repaired. Once the component is repaired, its icon changes from red to blue on the GT-MSOCC Configuration and Status page.

Detecting a Software Failure: Block Error Counts Too High. Besides hardware failures, software failures can occur. Typically, software failures are errors caused by mission-specific software running at one of the processors in the equipment string supporting a pass. A component with a software failure may still process data, but it may process at a decreased rate or it may degrade the data blocks flowing into the component. The three types of software failures that can occur are unacceptably high error block counts, data transmission that is fully terminated, or a data transmission rate that is significantly reduced. Procedures to detect a failure causing an unacceptable number of error blocks at a component are given below. The next section provides procedures to detect data transmission rate errors, which may be either a significantly decreased rate or transmission that is fully terminated due to a software problem.

Unacceptably high error block counts can be detected by means of telemetry pages on the right screen. The command "DISPLAY TELEM" will bring up a display containing information about data as they reach the MOR (SPF). On this page, columns labeled FLAGS and TBP indicate the two types of block errors that occur. A high proportion of error blocks to data blocks may indicate a problem with the MOR (SPF) or some previous component in the equipment string. Typically, error blocks arrive at a NAS line at a rate of .1 block per line per second, or 2 blocks per line per screen update. So the total error count, i.e., TBP plus FLAGS, should increase at the MOR, for example at about six errors per update at a 3 NAS line configuration that is correctly functioning.

There is some system jitter, so a small number of slowly updating block errors may not indicate any problem. If a problem is suspected, however, the operator may request information about other pieces of equipment. The command "DISPLAY MODLAN TELEM" will give some information about data as they flow into certain types of components, specifically the DOC, AP, GW, SPF, CMS, VIP, and RUP components. On the MODLAN TELEM display, the data block count is labeled as BLKS RCVD, and only a total error block measure combining FLAGS and TBP block counts is given. This measure is labeled RCVD ERRORS. A more detailed screen can be called for each component type, for example "DISPLAY TAC TELEM" and "DISPLAY VIP TELEM". These displays give data block counts (BC) and error block counts (both TBP and FLAGS) at each component. "DISPLAY NAS TELEM1" and "DISPLAY NAS TELEM2" will give information concerning data flow through all of the GT-MSOCC NAScom lines.

The GT-MSOCC operator is only responsible for detecting errors in the forward link, from the satellite to the MOR and other terminal points in the equipment string, e.g., RUP, GW, CMS, VIP, MOR, or SPF. There are two ways to detect a faulty component generating too many error blocks in the forward link. If there is more than one piece of equipment of a given type, for example multiple NAScom lines or multiple APs supporting the same satellite contact, all components of the same type should have approximately the same number of error blocks. If one NAScom line, for example, has twice as many error blocks as the other NAScom line(s) for the same pass, that line is likely to be faulty. The CMS, VIP and GW all receive data at the same point, that is after they are processed by the AP, and each of these receive approximately 30% of the total data and error blocks arriving at the MOR. Thus, the CMS, VIP, and GW should each receive approximately the same number of error blocks and total data blocks, unless one of the components is faulty.

It is important to remember, however, that the VIP may be supporting two missions concurrently and thus its error block count may be high due to the combined errors arriving from the two missions. In this case, the operator should monitor the transmission status of the previous piece of equipment in the equipment strings in both of the missions using the same VIP. If the VIP error block count is significantly higher that the total of the error block counts at the previous components, then a faulty VIP should be suspected and replaced, if possible.

This situation is also true of the RUP component. A single RUP may be supporting up to three passes concurrently. Thus, before diagnosing a faulty RUP due to an unacceptably high error block count, the total number of missions that the RUP is supporting must be determined and a calculation made of the expected number of error blocks.

Note, a RUP only receives 80% of the data and error blocks transmitted through the NAScom lines. As a result, a faulty RUP would have significantly more error blocks that 80% of the total error blocks recorded at each of the NAScom lines supporting passes connected to the RUP in question.

If multiple NAScom lines are in use, the NAScom error blocks must be totaled before a comparison is made with the subsequent component in the equipment string. For example, suppose NAS1 and NAS2 are both used by the same satellite. And suppose NAS1 has 10 RCVD ERRORS and NAS2 has 15 RCVD ERRORS. If all equipment is functioning properly, the MOR error block count is approximately 25 error blocks. To test for an error block problem at a unit subsequent to the NAScom lines, for example the TAC, the number of error blocks at the TAC should be compared to 25 to see if its value is about the same. If it is significantly higher than the total from the NAScom lines, then it is likely that the TAC has a transmission failure and should be replaced.

It is also important to note that each NAScom line receives 1/(number of NAScoms) of the data flowing through the system. If there are three NAScom lines, (1/number of NAScom) = 1/3, so that one third of the data arrives on each line; likewise, if there are two NAScom lines, one half of the data arrives on each line. Although only 80% of the total flowing into the system are transmitted to the RUP, all of the data from the NAScom lines flows through to the TAC, and on to the AP if there is one. If there are two APs, both of them process all the data. The GW, VIP, and CMS only receive about 30% of the data and error blocks that pass through the APs. However, all data and error blocks that flow through the AP(s) continue on to the MOR (SPF). Thus if 100 error blocks have accumulated at an AP, and there is no problem with the VIP, CMS, GW, or MOR (SPF), the GT-MSOCC operator should expect about 30 error blocks at the VIP, CMS, and GW, and 100 at the MOR (SPF). (NOTE: This assumes the VIP is only supporting one mission).

Recall error blocks are received at a NAS line at a rate of about 2 blocks per screen update in a correctly functioning system. A three NAS line configuration functioning correctly, for example, would expect an increase of about six total error blocks at the MOR per screen update. A significant increase in this rate is cause for suspicion.

Even if no error block problem is indicated in the main telemetry display (e.g., the DISPLAY TELEM page), detailed displays should occasionally be examined for problems at terminal points other than the MOR (SPF), e.g., the RUP, GW, CMS, and VIP. Detailed telemetry displays can be called for these components either by giving the command "DISPLAY MODLAN TELEM" or else "DISPLAY RUP TELEM", "DISPLAY GW TELEM", "DISPLAY CMS TELEM", and "DISPLAY VIP TELEM". The GT-MSOCC operator summary sheet has a list of these commands.

<u>Detecting a Software Failure</u>: <u>Decreased Data Transmission Rate</u>. Data flow problems can also be detected using pages displayed on the right terminal. The command "DISPLAY TELEM" will bring up a display

containing information about data as they reach the MOR (SPF). However, instead of looking for a large number of FLAG and TBP errors, the rate at which the data block count (BC) increases is important. Again, the GT-MSOCC operator is only responsible for detecting errors in the forward link, from a satellite to an MOR and to other terminal points in the equipment string supporting the pass.

The display updates the data block count every twenty seconds. In a properly functioning NAScom line, data blocks arrive at a rate of about ten blocks per second. As a result, in twenty seconds, it is reasonable to expect an increase of about 200 data blocks per line. Thus, in a three NAScom line configuration, data at the MOR (SPF) should increase approximately 600 data blocks per update. If the MOR (SPF) rate drops to 300 blocks per update, the operator should begin to examine data transmission rates at prior units in the equipment string to determine if there is a problem.

Similarly, for normally functioning equipment error blocks arrive at a rate of about .1 block per second or 2 blocks per 20 second screen update. So for a three NAScom line configuration, it is reasonable to expect about 6 error blocks (total) per 20 second screen update. Significantly more errors are cause for further examination of error block propagation through the system.

To obtain the data block counts for specific units, enter "DISPLAY MODLAN TELEM", "DISPLAY NAS TELEM1", "DISPLAY NAS TELEM2", "DISPLAY TAC TELEM", "DISPLAY AP TELEM", etc. When a component has a transmission rate problem, data will update more slowly for that unit than for the one preceding it. Since data have been processed at the problem unit at a decreased rate, the most obvious indication of a rate problem is that the data block count for the faulty unit will be less than that of the preceding unit.

Data block counts should be less for faulty components than for components of the same type (e.g., 2 APs supporting the same pass) or for components receiving data at the same point. For example, to find a transmission rate problem in NAScom lines, compare the total data block counts at the set of NAScom lines supporting that mission. If there is a large discrepancy, the NAScom line with the lowest data block count probably has a data transmission rate problem. The same method can be used to find a faulty AP when two APs are supporting a mission. Moreover, the GW, CMS, and VIP are all expected to receive approximately 30% of the data blocks received at the MOR (SPF) and thus should all receive about the same amount of data. (NOTE: This assumes that the VIP is supporting only one mission.)

Again, the main telemetry display will indicate problems only at the MOR (SPF) terminal point of the system. "DISPLAY MODLAN TELEM", "DISPLAY RUP TELEM", "DISPLAY GW TELEM", "DISPLAY CMS TELEM", and "DISPLAY VIP TELEM" should occasionally be called to search for problems at terminal points other than the MOR (SPF), namely at the RUP, GW, CMS, and VIP. Recall that the GW, CMS, and VIP only process 30% of the total data blocks. Thus, their normal rate of increase for incoming data blocks should be 30% of the rate coming through the previous unit. The RUP processes 80% of data transmitted through the

NAScom lines. Thus, a RUP supporting a single mission has a normal rate of increase of 80% of the combined rate of increase for the NAScom lines supporting the pass; for example, if two NAScom lines are being used, data blocks arrived at a rate of 400 blocks per update (i.e., every 20 seconds), and a RUP supporting only that mission should expect data blocks to arrive at a rate of 320 per update (i.e., 80% of 400).

In addition to significantly decreased transmission rates, another type of software failure is the complete termination of data transmission even when the component's hardware appears to be operating satisfactorily, i.e., its icon is coded green on the Configuration and Status page. A transmission termination failure at any of the terminal points, i.e., MOR (SPF), RUP, GW, VIP, or CMS, is indicated by a failure of the data block counts to update at all. If the failure occurs earlier in the equipment string supporting the spacecraft contact than a terminal point, e.g., at an AP, TAC, or NAScom line, a decreased tor terminated transmission rate can be detected on the MOR (SPF) data block count update display page. As with decreased data flow, other telemetry pages should then be checked to determine the cause of the error.

Replacing a Faulty Component. Once a component has been identified as faulty, it needs to be replaced, if possible, with another component of the same type. A replacement unit needs to be currently free, and also not scheduled for other use during the required time period. Several displays are available to view scheduled hardware allocation. The commands "DISPLAY NAS AVAIL1", "DISPLAY NAS AVAIL2", "DISPLAY NAS AVAIL3", "DISPLAY TAC AVAIL", "DISPLAY RUP AVAIL", "DISPLAY AP AVAIL", "DISPLAY GW AVAIL", "DISPLAY CMS AVAIL", "DISPLAY SPF AVAIL", "DISPLAY DOC AVAIL", "DISPLAY MPT AVAIL", and "DISPLAY VIP AVAIL" will display graphical representations of schedules for the respective classes of hardware components. These schedules will appear on the left screen. On the graphics schedule pages, the colored bars on the display indicate time periods during which components are scheduled for use.

Alphanumeric schedule displays are available for each individual component, for example "DISPLAY AP2 SCHED" and "DISPLAY NAS23 SCHED". These displays indicate which missions are scheduled to use the specific piece of equipment and for what time period, as well as other information. The command "DISPLAY MSOCC SCHED" will display the next several scheduled passes, the equipment reserved to support these passes, and other information.

Once the GT-MSOCC operator identifies a replacement component, a command must be given to make the replacement. The command to replace a faulty component with an available component of the same type is given in the box below. Once the operator has manually replaced a component, the equipment string must be manually deconfigured at the conclusion of a pass. The deconfigure command is also shown

Operator Commands to Replace a Faulty Component and Manually Deconfigure a Mission

REPLACE old-one new-one

DECONFIGURE mission-name*

*See Table 1 for a list of mission-name abbreviations.

For example, if AP1 has been detected as a faulty unit and AP5 is not in use, does not have a hardware failure, and is not scheduled for use, the GT-MSOCC operator makes the replacement with the command "REPLACE AP1 AP5". The box on the following page summarizes the information retrieval commands used to find replacements and replace equipment.

A faulty component cannot always be replaced. MORs are spacecraft-specific and are not interchangeable. Also, there may be no replacement unit available for other GT-MSOCC equipment. If a component has a problem and cannot be replaced, the GT-MSOCC operator must report this. To send a message the operator enters "ALERT" followed by the message, for example, "ALERT MOR12 IS GENERATING TOO MANY ERROR BLOCKS".

2. Compensation for Automated Schedule Problems

The third major GT-MSOCC operator function is compensation when the automatic schedule and control system is unable to configure a scheduled pass. If a pass is scheduled to occur, but one of its scheduled components has failed or is being used, the GT-MSOCC operator receives a message, such as "Unable to configure ERBE: TAC3 unavailable." The command "DISPLAY PENDING" will give a current list of missions that cannot be automatically configured because one or more of the scheduled components is unavailable. In our example, the operator must identify a suitable replacement for TAC3 and manually configure ERBE making the necessary replacement. This procedure is called configure and replace. The configure and replace command is

Schedule Information and Replacement Commands

<u>Graphics Schedule for Equipment Classes</u>

DISPLAY NAS AVAIL1
DISPLAY NAS AVAIL2
DISPLAY NAS AVAIL3
DISPLAY TAC AVAIL
DISPLAY RUP AVAIL
DISPLAY AP AVAIL
DISPLAY GW AVAIL
DISPLAY CMS AVAIL
DISPLAY VIP AVAIL
DISPLAY SPF AVAIL
DISPLAY DOC AVAIL
DISPLAY MPT AVAIL

<u>Alphanumeric Schedules for Equipment Items</u>

DISPLAY NASn SCHED n=1,...,33
DISPLAY RUPn SCHED n=1,2,3
DISPLAY TACn SCHED n=1,...,8
DISPLAY APn SCHED n=1,...,7
DISPLAY CMSn SCHED n=1,2
DISPLAY GWn SCHED n=1,2
DISPLAY VIPn SCHED n=1,2,3

<u>Replace Command for Faulty Component</u>

REPLACE old-item new-item

Operator Commands to Replace and Deconfigure Equipment for a Special Request

CONFIGURE mission-name* REPLACE old-one new-one

DECONFIGURE mission-name*

*See Table 1 for a list of mission-name abbreviations.

"CONFIGURE ERBE REPLACE TAC3 TAC5". The operator has about 3 minutes to configure and replace a problem with the automated schedule. If a replacement component cannot be found in that time, the pass will be removed from the pending mission list.

If the GT-MSOCC operator manually configures the equipment supporting a pass, he/she must also manually deconfigure the equipment when the pass is complete. For example, the operator gives the command, "DECONFIGURE ERBE", when the manually configured ERBE pass is complete.

Before replacing faulty equipment, the operator should identify a replacement component that is currently available and not scheduled for use for the duration of te current pass containing the faulty component. Replacing a failed component with one that is scheduled for use by another mission during the current pass' duration will cause a subsequent problem with the e automated scheduling system and is considered an operator error. Before beginning a replace command, the operator should carefully inspect the replacement component's schedule to ensure that no conflicts are caused. Finally, for components that can support multiple users, the operator must ensure that a replacement does not violate constraints for either current or scheduled use. If no replacements are available the operator should enter "ALERT" followed by a message to that effect. As with replacement of a faulty component, manually configured equipment requires the operator to manually deconfigure the equipment at the conclusion of a pass.

3. Responding to Special Configuration Requests

The GT-MSOCC operator may also be asked to configure an unscheduled, emergency pass for a given time duration or to respond to requests about the feasibility of immediately scheduling a particular spacecraft contact. The operator first needs to know what types of equipment the satellite will need to use. The operator is provided with a chart that gives this information (Table 3). Second, the operator should consult the equipment schedules to find specific components that are free for the required time period. A valid piece of equipment should be currently available, should not be failed, and should not be scheduled during the required time period. It is a serious operator error to prohibit an automatically scheduled configuration because a scheduled piece of equipment has been used by the operator. The operator must also ensure that a pass for this mission is not already scheduled within the requested time duration; "DISPLAY mission-name SCHED" will provide this information. Finally, before responding to the question, the operator should ensure that the proposed addition will not increase the total number of concurrently supported missions to more than five. The GT-MSOCC system is not capable of supporting more than five missions simultaneously. Violation of this constraint is considered a serious operator error. If all the needed equipment is available, the operator should respond to the question, referencing the question number and, if the request was to actually configure the equipment, proceed to do so. Several examples of questions and operator answers are given below.

MISSION NAME	ABBREV	NAS	RUP	TAC	AP	CMS	GW	VIP	MOR	SPF
GEOGRAPHIC EXPLORER	GEO	3	1	1	1	1	-	1	2	-
LANDSAT	LNSAT	3	1	1	1	1	1	1	3	-
ATMOSPHERIC EXPLORER QL	AE-QL	2	-	1	1	-	-	1	4	-
ERBE	ERBE	3	1	1	1	-	1	-	5	-
DEEP SPACE EXPLORER I	DSEI	2	1	-	2	-	-	1	6	-
GEOSAT	GSAT	3	1	1	1	-	-	1	7	-
INNER SPACE EXPLORER	ISE	3	1	1	1	-	-	1	8	-
WEATHERSAT QL	WS-QL	2	-	1	1	-	-	1	9	-
ASTROSEARCH	ASTRO	3	1	1	1	-	1	-	10	-
VENTURE	VENTR	3	1	1	2	1	-	1	11	-
PLANETARY MISSION	PM	3	1	1	1	1	-	1	12	-
SPACE SHUTTLE	SS	3	1	1	-	-	-	1	-	1
DYNAMIC EXPLORER	DE	3	1	1	2	-	-	1	13	-
ATMOSPHERIC EXPLORER D	AE-D	2	-	1	-	-	1	-	4	-
SOLARCRAFT	SOLAR	3	1	1	1	1	-	1	1	-
DEEP SPACE EXPLORER II	DSEII	2	1	-	2	-	-	1	6	-
WEATHERSAT D	WS-D	2	-	1	1	-	-	1	9	-

In the case of questions of type (b) or (d), after answering, if the required equipment is available, the operator should actually configure the equipment to support the request.

The command to configure an unscheduled spacecraft contact is entered in two lines. On the first line the operator enters the mission name and pass duration in minutes; on the second line the operator enters the equipment selected to support the pass. As with a configure-replace command, since the GT-MSOCC operator manually configured the pass, he/she must also manually deconfigure the equipment. When the pass is complete, the operator gives the command "DECONFIGURE mission-name". The format of the configure and deconfigure commands are given in the box below. The time duration is given in minutes.

a)	Q	12345	Can ERBE be supported for 5 minutes?
	A	12345	YES, EQUIPMENT IS AVAILABLE.
b)	Q	4123	Please configure ERBE for 4 minutes.
	A	4123	NO, NOT ALL EQUIPMENT IS AVAILABLE.
c)	Q	8179	Can AE-D be supported for 3 minutes?
	A	8179	YES
d)	Q	7431	Please configure SS for 6 minutes.
	A	7431	OK (Operator then proceeds to configure SS.)

Operator Commands to Manually Configure and
Deconfigure Equipment for a Special Request

CONFIGURE mission-name* duration-time <RETURN>
equip1 equip2 equip3 ... equipn** <RETURN>

DECONFIGURE mission-name*

*See Table 1 for a list of mission-name abbreviations.

**Do not specify MOR(SPF); system will know which to use for a given mission.

An example of a typical command sequence is

CONFIGURE ERBE 5 <RETURN>

NAS2 NAS3 NAS4 TAC1 RUP1 AP2 <RETURN>

DECONFIGURE ERBE

4. Deconfigure all Manually Configured or Reconfigured Equipment Strings. As stated above any equipment that has been manually configured or reconfigured due to a component failure during a pass or a problem with the automated schedule and control system must be manually deconfigured as quickly as possible. Failure to respond promptly means that configured equipment is unavailable for current use and may cause a subsequent problem with the automated schedule and control system. Operator-caused errors of this type are very serious.

GT-MSOCC Operator Function Priorities

As indicated above, the GT-MSOCC operator role is comprised of several operator functions:

1) monitoring the hardware status and data transmission quality of the components constituting the equipment strings supporting each currently active pass.

2) deconfiguring manually configured equipment.

3) compensating for problems encountered by the automatic schedule when attempting to configure reserved equipment for a previously scheduled pass.

4) responding either to requests for unscheduled mission support or to requests about the feasibility of unscheduled support. There is a priority implicit in these functions.

Under normal circumstancea, the operator is expected to monitor currently active passes. This function is preempted by any of the next three operator functions. The highest priority function is to deconfigure manually configured equipment. The GT-MSOCC operator should quickly deconfigure manually configured equipment. Failure to do so means that the configured equipment is unavailable for use, and if it is scheduled for immediate use, the current state will cause a subsequent automated schedule problem.

The next priority function is the compensation for problems with the automated schedule. When the operator is notified of an automatic schedule problem, other activities should cease until a replacement component is found and the mission is manually configured, or until the operator decides that no replacement component is available.

Compensating for failed equipment, i.e., equipment with either hardware or software failures, is the third priority function. Any time a hardware failure is detected or a software failure is suspected, the operator should attempt to confirm and replace the failed component.

Responding to ad hoc requests for special configurations is the lowest priority function. The operator should respond to requests as quickly as possible, but only if all currently supported passes appear to be operating satisfactorily and there are no missions pending that the automatic scheduling system is unable to configure.

A final comment on the overall operator function is needed. Operator-caused automatic schedule problems are considered serious operator errors. The operator should carefully select a component to replace a failed component in a current pass or to replace an unavailable component in a scheduled pass. The operator should ensure that the component is not already reserved for use at some time during the period for which the operator is making the manual replacement or configuration. When reponding to ad hoc requests, the operator should ensure that sufficient equipment is currently available and not previously scheduled for the period in question.

The second type of serious operator error is manually configuring a mission so that the total number of concurrently supported missions exceeds five. The GT-MSOCC system is not capable of supporting more than five passes simultaneously and the operator must ensure that this limitation is not violated.

Operator Errors

1. Failing to deconfigure an ended mission so that its equipment is needed, but not available for use, thus causing an automated scheduling problem.

2. Using equipment to
   a) replace a component
   b) configure a scheduled mission having an unavailable component
   c) configure a new pass

   when that equipment is already scheduled, and thus causing an automated scheduling problem.

3. Deconfigure a mission before its pass is completed.

4. Loading more than five missions concurrently. Also, manually configuring a mission without consulting the MSOCC schedule, so that more than five missions are scheduled concurrently. Note that the automatic scheduler never schedules more than five missions at once; this could only occur after the operator manually configures an unscheduled pass.

5. Manually configuring an unscheduled pass over a time when that spacecraft is already scheduled. The spacecraft specific MOR will not be available for the scheduled pass, thus causing an automated scheduling problem.

GT-MSOCC Procedures

1. Manually deconfigure a mission that
   a. has been manually configured.
   b. has had a component replaced during the pass.

   - When the pass is over, the operator receives a message, "ERBE pass completed: Please deconfigure manually."

   - Enter the command,
     "DECONFIGURE ERBE"

2. Configure and replace a component on a pending mission.
   - The operator receives a message,
     "Unable to configure ERBE: TAC7 is unavailable."

   - Find which components (TACs) are currently idle or are not loaded to full capacity, and are not failed. See the center Configuration and Status page.

   - Find the required time period. See down time for ERBE on, "DISPLAY PENDING"

   - See which of the components (TACs) that are currently available, are not scheduled before down time.
     "DISPLAY TAC3 SCHED"
     "DISPLAY TAC AVAIL"
     "DISPLAY MSOCC SCHED"

   - Enter the command,
     "CONFIGURE ERBE REPLACE TAC7 TAC3"
     if more than one component needs to be replaced, enter
     "CONFIGURE ERBE REPLACE TAC7 TAC3 NAS18 NAS30"

3. For equipment which is in use, monitor
   a. for hardware failure
   b. for poor data transmission quality (software failure).

   a) To detect a hardware failure, view the Configuration and Status page. Any component that is on the top portion of the page and is red, is in use and has failed.

   b) To detect a software problem
      i) decreased data flow rate
         - "DISPLAY TELEM" to view data in coming at the MOR. See BC, the Block Count column.

         - Make sure data is increasing at a rate of number of NAS * 200 blocks per update. If not, "DISPLAY AP TELEM" or "DÍSPLAY TAC TELEM", "DISPLAY NAS TELEM", "DISPLAY NAS TELEM2".

- To get information about other end points, "DISPLAY RUP TELEM" or "DISPLAY GW TELEM" or "DISPLAY CMS TELEM" or "DISPLAY VIP TELEM". The RUP should have 80% of the combined NAS BC. The GW, CMS and VIP should have 30% of AP Block Count.

ii) too many errors
- "DISPLAY TELEM" to find error count at the MOR. Look for lard numbers, quickly increasing under the columns TBP and FLAGS.

- Trace through the configuration for the cause, "DISPLAY TAC TELEM" or "DISPLAY AP TELEM", "DISLPAY NAS TELEM1", "DISPLAY NAS TELEM2".

- Other system end points should be checked for too high FLAG or TBP count. "DISPLAY RUP TELEM", "DISPLAY RUP TELEM", "DISPLAY GW TELEM", "DISPLAY CMS TELEM", "DISPLAY VIP TELEM". RUP receives 80% of combined NAS errors. The GW, CMS and VIP receive 30% of AP errors.

4. Replace a faulty component

- Once a component, say TAC6, has been detected as faulty, find components of the same type that are not in use, or are not loaded to full capacity, and are not failed. See Configuration and Status page.

- On Configuration and Status page see which mission the component (TAC6) is supporting.

- Find when that spacecraft's pass is over "DISPLAY TELEM".

- See which of the components (e.g., TAC3) that are currently available are not scheduled for use before down time
"DISPLAY TAC3 SCHED"
"DISPLAY TAC AVAIL"
"DISPLAY MSOCC SCHED"

- Enter the command, "REPLACE TAC6 TAC3"

5. Respond to requests and configure unscheduled missions.

- The system message is,
"Q 112 Please configure ERBE for 6 minutes".

- First note that to configure an unscheduled ERBE pass for 6 minutes:
1. An ERBE pass cannot currently be scheduled.
2. An ERBE pass should not be scheduled within 6 minutes.

3. Less than five concurrent passes should be scheduled during the next 6 minutes.

- For each needed component type, find which components are idle, or not loaded to full capacity, from the Configuration and Status page.

- For each needed component type, find which of these components are not scheduled within the next 6 minutes.
  "DISPLAY component-name SCHED", e.g., TAC3
  "DISPLAY component-type AVAIL", e.g., TAC
  "DISPLAY MSOCC SCHED"

- If all required components are available, answer the question affirmatively, "A 112 ERBE 6 minute pass is possible" and then configure the mission,
  "CONFIGURE ERBE 6" <RETURN>
  "NAS18 NAS7 NAS9 TAC3 RUP1 AP2 CMS1 VIP3"
  Note: It is not necessary to include the MOR.

- If the pass cannot be configured, answer negatively,
  "A 112 ERBE pass not possible no free CMS".

GT-MSOCC OPERATOR INPUTS

Operator Information Retrieval commands

1. To obtain telemetry pages:

DISPLAY TELEM
DISPLAY MODLAN TELEM
DISPLAY NAS TELEM1
DISPLAY NAS TELEM2
DISPLAY TAC TELEM
DISPLAY RUP TELEM
DISPLAY AP TELEM
DISPLAY GW TELEM
DISPLAY CMS TELEM
DISPLAY VIP TELEM

2. To obtain graphically displayed equipment schedule pages:

DISPLAY component-type AVAIL

DISPLAY NAS AVAIL1	DISPLAY GW AVAIL
DISPLAY NAS AVAIL2	DISPLAY CMS AVAIL
DISPLAY NAS AVAIL3	DISPLAY VIP AVAIL
DISPLAY TAC AVAIL	DISPLAY SPF AVAIL
DISPLAY RUP AVAIL	DISPLAY DOC AVAIL
DISPLAY AP AVAIL	DISPLAY MPT AVAIL

3. To obtain alphanumeric schedule pages:

- DISPLAY MSOCC SCHED
- DISPLAY PENDING

- DISPLAY mission-name SCHED
  Mission-name = ERBE, GEO, LNSAT, SS, AE-QL, DSEI, GSAT, WS-QL, ASTRO, VENTR, PM, DE, AE-D, SOLAR, DSEII, WS-D, ISE

- DISPLAY component-name SCHED

DISPLAY NASn SCHED n=1,...,33
DISPLAY RUPn SCHED n=1,2,3
DISPLAY TACn SCHED n=1,...,8
DISPLAY APn SCHED n=1,...,7
DISPLAY CMSn SCHED n=1,2
DISPLAY GWn SCHED n=1,2
DISPLAY VIPn SCHED n=1,2,3

4. To obtain events/alarm page:

- DISPLAY EVENTS

Command	Contents
"DISPLAY NAS AVAIL1"	NAScom Lines 1-15
"DISPLAY NAS AVAIL2"	NAScom Lines 16-30
"DISPLAY NAS AVAIL3"	NAScom Lines 31-33
"DISPLAY AP AVAIL" "DISPLAY TAC AVAIL"	All APs and all TACs
"DISPLAY RUP AVAIL" "DISPLAY GW AVAIL" "DISPLAY SPF AVAIL" "DISPLAY MPT AVAIL" "DISPLAY CMS AVAIL" "DISPLAY VIP AVAIL"	RUPs, GWs, SPFs, MPTs, CMSs, VIPs
"DISPLAY MOR AVAIL"	MORs
"DISPLAY DOC AVAIL"	DOCs

Operator Control Commands

1. To configure a scheduled mission and replace:
   CONFIGURE mission-name REPLACE old-component new-component

   Example: CONFIGURE ERBE REPLACE TAC6 TAC3

2. To replace a failed component in an ongoing pass:
   REPLACE old-component-name new-component-name

3. To manually configure an unscheduled mission:
   CONFIGURE mission-name minutes component component... *
   Example: CONFIGURE ERBE 9 <RETURN>
   NAS1 NAS2 NAS3 TAC1 RUP1 AP1...*
   *Do not specify MOR(SPF)
   (see Table 3)

4. To respond to a special request:
   A query-number message
   Example: A 13526 CAN NOT CONFIGURE ERBE
   A 5712 CAN CONFIGURE SS

5. To manually deconfigure a mission:
   DECONFIGURE mission-name
   Example: DECONFIGURE ERBE

6. To send an alert message:
   ALERT msg-string
   Example: ALERT MOR6 TOO MANY ERROR BLOCKS
   ALERT NO REPLACEMENT FOR RUP1

Note: Commands can be written upper or lower case and the words DISPLAY, REPLACE, CONFIGURE, ANSWER and HELP can be abbreviated c their first letter or first two or three letters.

MISSION NAME	ABBREV	NAS	RUP	TAC	AP	CMS	GW	VIP	MOR	SPF
GEOGRAPHIC EXPLORER	GEO	3	1	1	1	1	-	1	2	-
LANDSAT	LNSAT	3	1	1	1	1	1	1	3	-
ATMOSPHERIC EXPLORER QL	AE-QL	2	-	1	1	-	-	1	4	-
ERBE	ERBE	3	1	1	1	-	1	-	5	-
DEEP SPACE EXPLORER I	DSEI	2	1	-	2	-	-	1	6	-
GEOSAT	GSAT	3	1	1	1	-	-	1	7	-
INNER SPACE EXPLORER	ISE	3	1	1	1	-	-	1	8	-
WEATHERSAT QL	WS-QL	2	-	1	1	-	-	1	9	-
ASTROSEARCH	ASTRO	3	1	1	1	-	1	-	10	-
VENTURE	VENTR	3	1	1	2	1	-	1	11	-
PLANETARY MISSION	PM	3	1	1	1	1	-	1	12	-
SPACE SHUTTLE	SS	3	1	1	-	-	-	1	-	1
DYNAMIC EXPLORER	DE	3	1	1	2	-	-	1	13	-
ATMOSPHERIC EXPLORER D	AE-D	2	-	1	-	-	1	-	4	-
SOLARCRAFT	SOLAR	3	1	1	1	1	-	1	1	-
DEEP SPACE EXPLORER II	DSEII	2	1	-	2	-	-	1	6	-
WEATHERSAT D	WS-D	2	-	1	1	-	-	1	9	-

Exercise 1

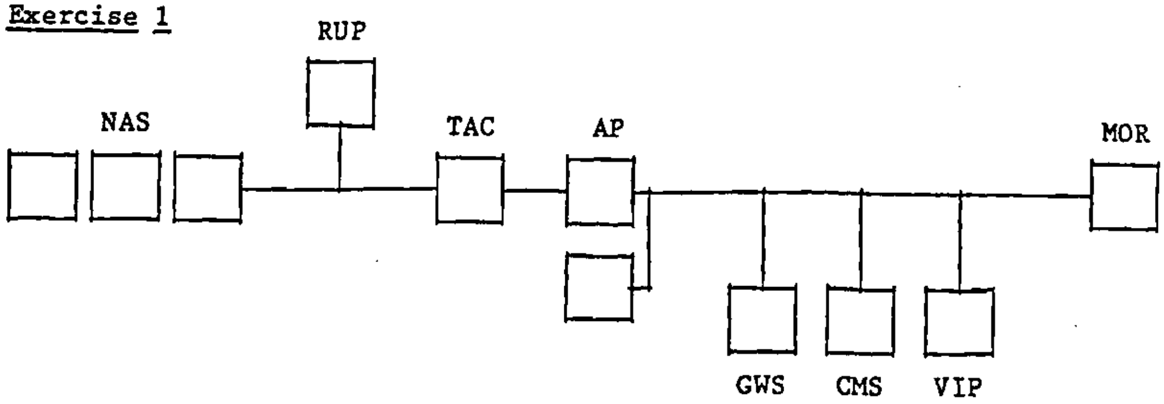


Figure 1

It is important to know the direction of data flow and the amount of data which should flow through each unit. The order of data flow is as follows. First data flow through NAScom lines. Second, data flow through the TAC and RUP processors; third, data flow through the APs, and finally, data flow through the GW, CMS, VIP and the MOR terminal points. In the configuration given above, since there are 3 NAScom lines, one-third of the data arrive at each NAS. The RUP receives 80% of the incoming data; but the TAC and the AP receive all of the incoming data. The GW, CMS and VIP receive 30% of the data, while the MOR like the TAC and AP receives all of the data.

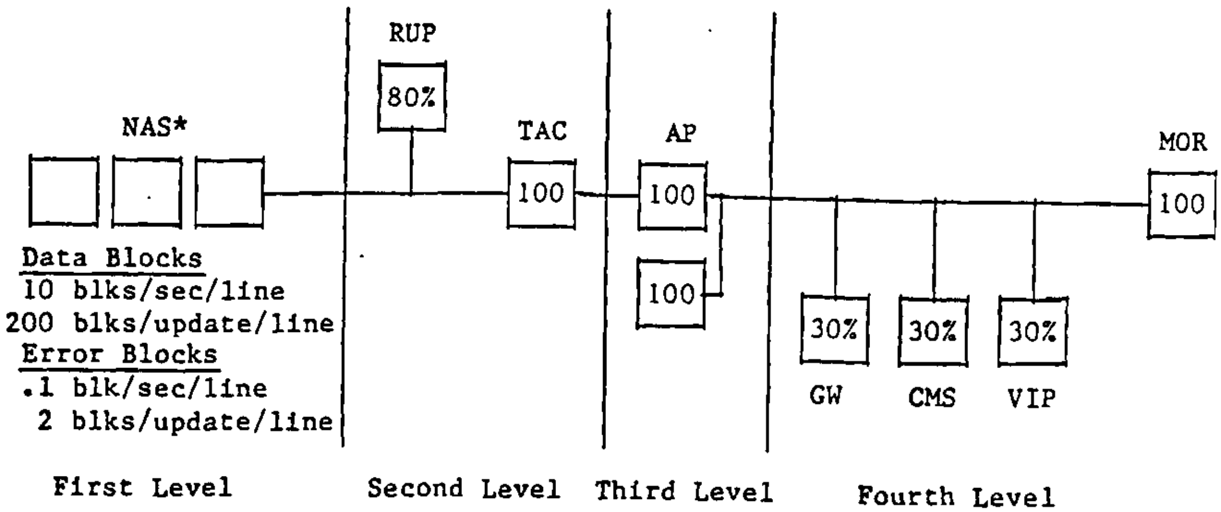


Figure 2

Question: If 600 blocks of data are received jointly by the 3 NAScom lines in a normally functioning configuration (shown below), how many blocks is each subsequent component in the equipment string expected to receive?

Answer:

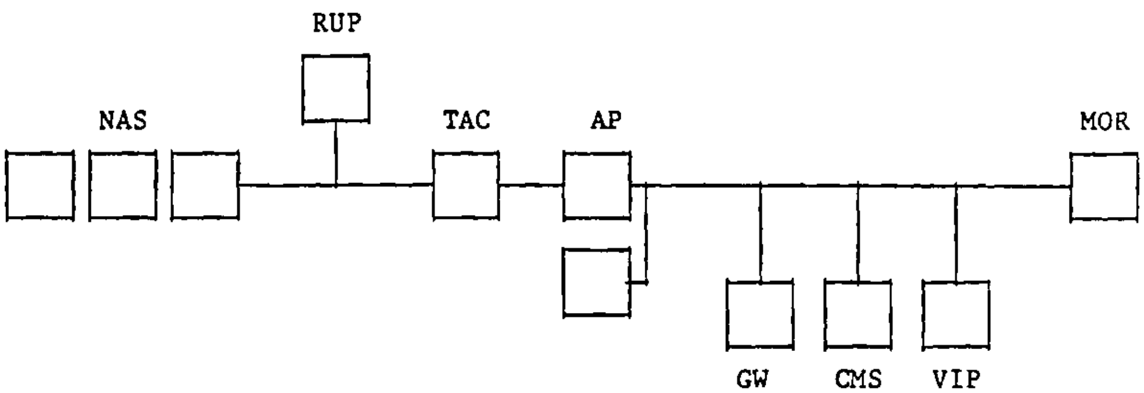


Figure 3

<u>Exercise 2</u> (See Figure 2 and 3)

Q 1: Suppose 200 blocks flow through each NAS and there are <u>no</u> faulty components; fill in expected block counts for subsequent components in the equipment string (line 1) given in the table below.

Q 2: The RUP is a faulty component, fill in expected block counts for subsequent components in the equipment string (line 2) given in the table below.

Q 3: The TAC is a faulty component, fill in expected block counts for subsequent components in the equipment string (line 3) given in the table below.

Q 4: The GW is a faulty component, fill in expected block counts for subsequent components in the equipment string (line 4) given in the table below.

Q 5: The AP is a faulty component, fill in expected block counts for subsequent components in the equipment string (line 5) given in the table below.

	NAS	NAS	NAS	RUP	TAC	AP	AP	GW	CMS	VIP	MOR
Q1:	200	200	200								
Q2:	200	200	200	480							
Q3:	200	200	200	480	300						
Q4:	200	200	200	480	600	600	600	45			
Q5:	200	200	200	480	600	300	600				

TABLE 1

(See Figure 3)

Q 6: Suppose data began flowing at a decreased rate at the MOR. which components might be causing the problem?

Q 7: Suppose data began flowing at a decreased rate at the RUP. which components might be causing the problem?

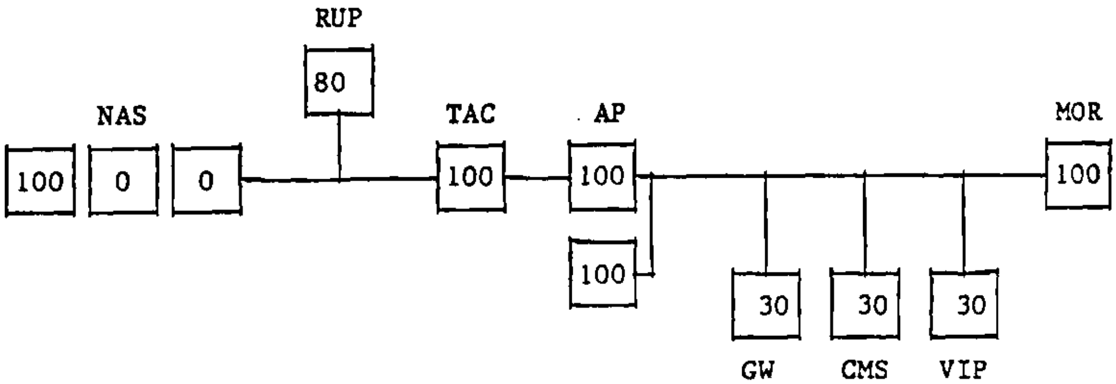
Q 8: Suppose data began flowing at a decreased rate at the CMS. which components might be causing the problem?

Exercise 3

Decreased data flow rate is only one indication of a faulty unit. A faulty piece of equipment may create error blocks, i.e., blocks that are garbled or arrive out of sequence. Any error blocks created at a components are passed to subsequent components.

For example, suppose a NAScom line created 100 error blocks in the configuration below.

Error Block



Since GW, CMS and VIP only receive 30% of the data flowing through the configuration, correctly functioning units are expected to receive approximately 30% of the error blocks.

Given this background, fill in the answers in the table below.

Q1: Given 100 error blocks at one NAScom line and none at the other two lines, how many error blocks are expected at subsequent units?

Q2: Given 100 error blocks in a TAC, how many error blocks are expected at subsequent units?

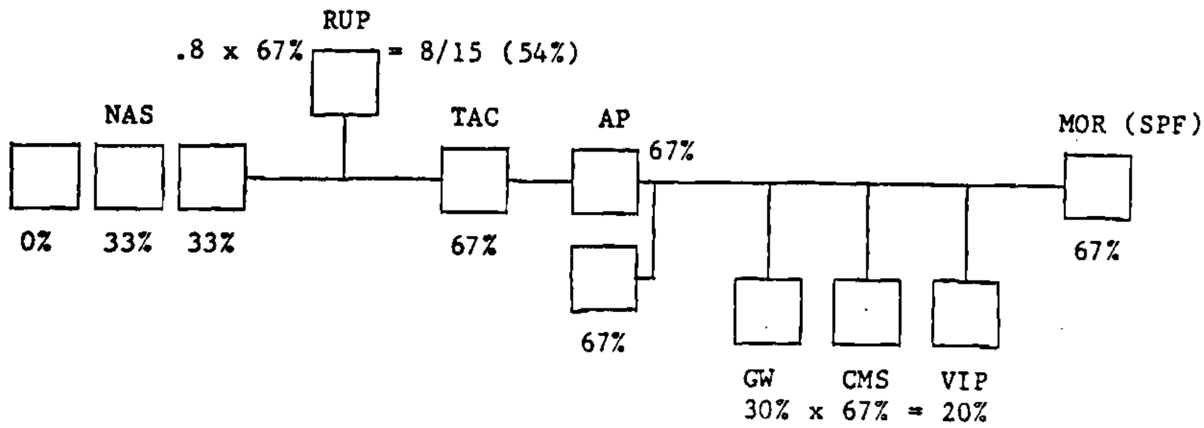
Q3: Given 100 error blocks in one AP, 50 in the other AP, how many error blocks are expected at subsequent units?

Q4: Given 100 error blocks at a GW, how many error blocks are expected at subsequent units?

	NAS	NAS	NAS	RUP	TAC	AP	AP	GW	CMS	VIP	MOR
Q1:	0	100	0								
Q2:	0	0	0	0	100						
Q3:	0	0	0	0	0	100	50				
Q4:	0	0	0	0	0	0	0	100			

Exercise 4

A more serious data transmission rate problem occurs when data flow completely stops at a unit. If data flow stops at one NAS, for example, data would flow through the components at the following rate:



Q1: Suppose data flow stopped at the MOR. Which components could be responsible (in the configuration above)?

Q2: Why couldn't a NAS be responsible for this problem?

Q3: If the configuration had only one AP, could it be responsible?

Exercise 5

A low flow rate or a high error block count found at a terminal point in the equipment string, such as the MOR, may be an indication of a problem with the MOR, with preceding units in the equipment string, or may be no problem at all. There is some system jitter, so if errors are evenly dispersed throughout the system, there probably is not a problem. To find errors, look for significantly more error blocks on a suspected component than on previous units, or less flow than at previous untis, or different flow rates on units at the same level.

PROBLEMS:

1. The satellite has been transmitting and the amount of data blocks received are as follows are given below. Assume in all of these problems that RUP and VIP are supporting only one mission.

NAS	NAS	NAS	RUP	TAC	AP	GW	CMS	VIP	MOR
1948	2018	2019	4825	5925	0	0	0	0	0

2. The following error blocks have been transmitted.

NAS	NAS	NAS	RUP	TAC	AP	GW	VIP	MOR
11	12	12	28	36	36	10	10	38

Which is (are) faulty component(s)?

3. The following error blocks have been transmitted.

NAS	NAS	RUP	AP	AP	CMS	VIP	MOR
68	33	81	105	101	29	30	107

Which is (are) faulty component(s)?

NOTE: It is important to remember that the RUPs can support up to 3 satellites at once, and that the VIPs can each support up to two satellites at once. Thus, the amount of data blocks received and the number of error blocks for a RUP or a VIP may reflect transmission from more than one satellite.

APPENDIX C

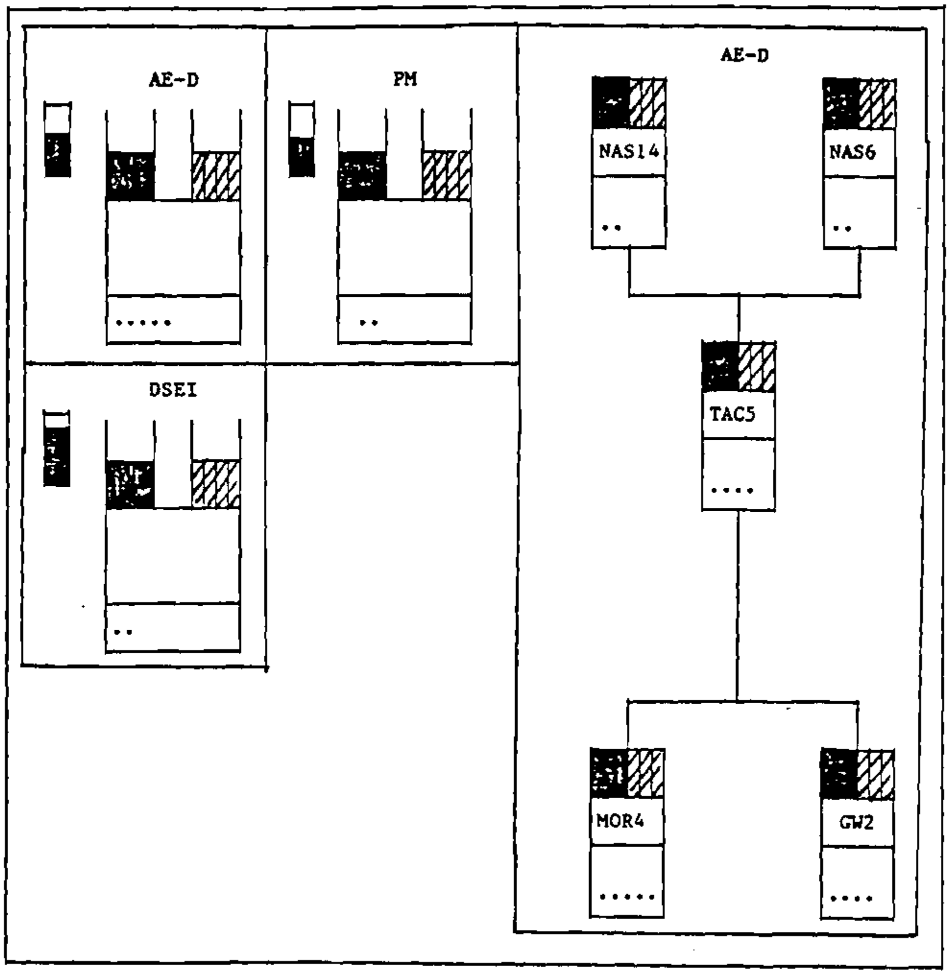
GT-MSOCC Operator Instructions for the Window Display Condition

GT-MSOCC OPERATOR WORKSTATION

The GT-MSOCC operator workstation consists of two CRT screens and a keyboard on which the operator enters commands and information requests. The right screen is the monitoring and supervisory control screen; information displayed on this screen enables the GT-MSOCC operator to monitor incoming data flow rates and block error counts for all currently supported spacecraft contacts. The left screen provides alphanumeric windows through which the GT-MSOCC operator can replace failed equipment, configure unscheduled spacecraft contacts, and inspect spacecraft and equipment schedules. The remainder of this section will provide more detail about these screens.

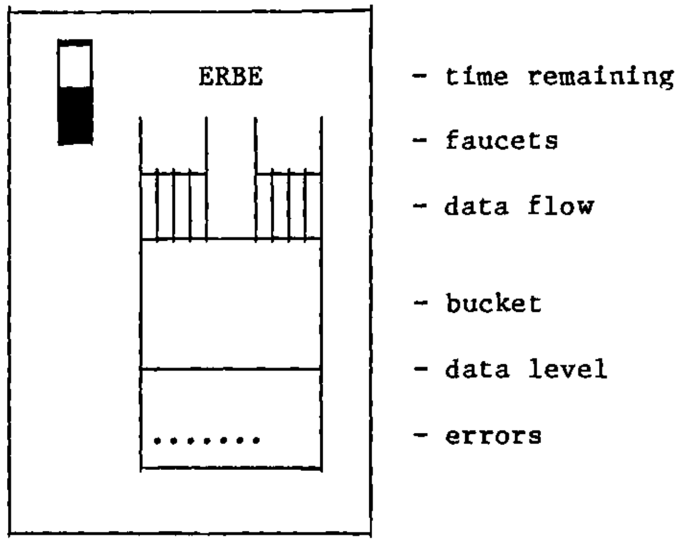
The right screen of the workstation provides information about data flow rate and data quality as it reaches a terminal points in the GT-MSOCC system. Figure 5 shows an example of the display page. Each mission which is currently transmitting data has its own spigot icon displayed on the left side of the right screen. Each mission icon is labeled with the name of the mission that it represents and has a partially filled blue bar that indicates time remaining for the current pass. The area of the entire bar represents 10 minutes, thus, a half filled bar indicates that about 5 minutes of data transmission remain for the pass.

The mission spigot icon depicted in Figure 6 is a qualitative representation of the most significant features of the data blocks and error blocks flowing through the equipment string supporting an individual satellite contact. The spigot icon represents information



A Sample of the Right Graphics Terminal

Figure 5



A faucet icon is provided for each satellite currently engaged in a pass.

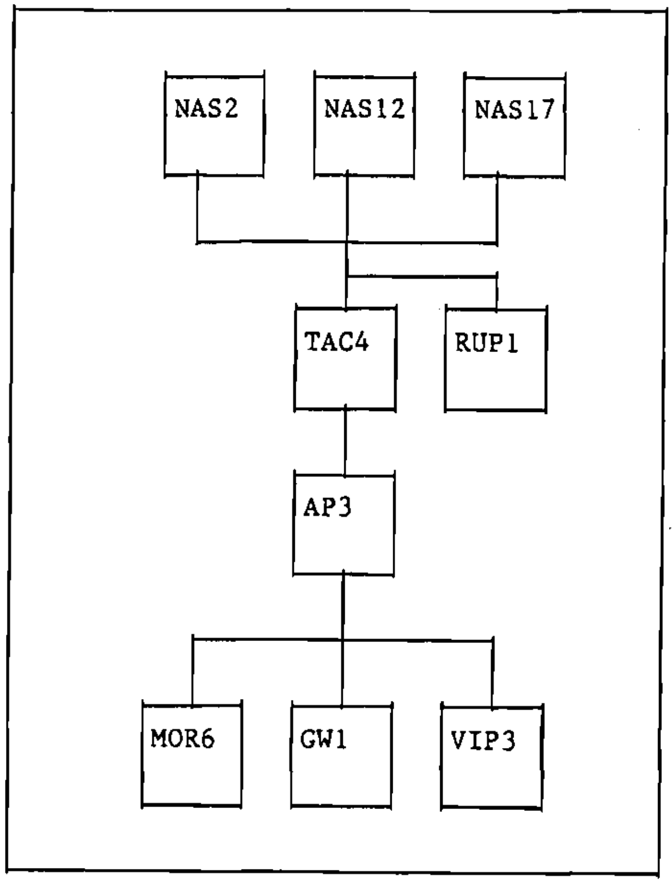
Figure 6

about data block flow rate, error block counts, and represents information flow as data reach terminal points in the equipment string supporting the spacecraft contact. The data flow rate depicted by the icon is dynamic. It represents the smallest data flow rate at any of the terminal points in the mission's equipment string, i.e., it represents the flow rate at either the MOR (SPF), RUP, GW, CMS, or VIP depending on the individual configuration. The operator monitors the flow rate to detect either significantly decreased or fully terminated data flow, problems requiring immediate operator intervention. Two types of data, telemetry and non-telemetry, flow through the two spigots into a bucket. The rising level on the icon represents the amount of data that has reached the MOR (SPF) terminal point. The red dots collecting at the bottom of the bucket represent the amount of bad data i.e., error blocks that have been transmitted through the system. As with data flow rate, the error block counts depicted at the bottom of the bucket are dynamic, and, at any one time, depict the terminal point for the mission equipment configuration string with the most error blocks. The operator monitors error blocks in order to detect components creating bad data blocks. Like problems with data flow, a significant increase in the error block count at one of the terminal points requires operator attention. The mission spigot icons located on the left half of the right screen indicate overall data flow rates and error block counts for each transmitting satellite. Their purpose with a satellite's computer and communication equipment. When a potential problem is suggested, the operator may request more detailed information about an individual equipment string. Three operator commands that display additional information about the current status of components supporting a satellite contact are given in the box below. All three commands will cause additional graphical information to be displayed on the right half of the right screen.

Operator Commands To Obtain More Detailed Information about a Mission
DISPLAY mission-name* STATUS DISPLAY mission-name* FLOW DISPLAY MORE mission-name*
*Table 1 contains a list of abbreviations used for mission-name.

The first command, DISPLAY mission-name STATUS, displays a detailed representation of equipment supporting the satellite contact together with the hardware status fo each. An example is given in Figure 7 Each box in the large icon represents a piece of equipment in the equipment string. The color of the box represents the equipment's status, i.e., a green box indicates that the hardware component is fully operational and a red box indicates that the hardware component is failed.

The second command, DISPLAY mission-name FLOW, displays an icon similar to that used in the STATUS display described above, but rather than giving hardware status it provides information about the data block flow rates and error block counts at each component in the equipment configuration. Like the mission spigot icon, the icon



A Status icon for one of the spacecraft currently engaged in a pass. Correctly operating components are green and failed components are red.

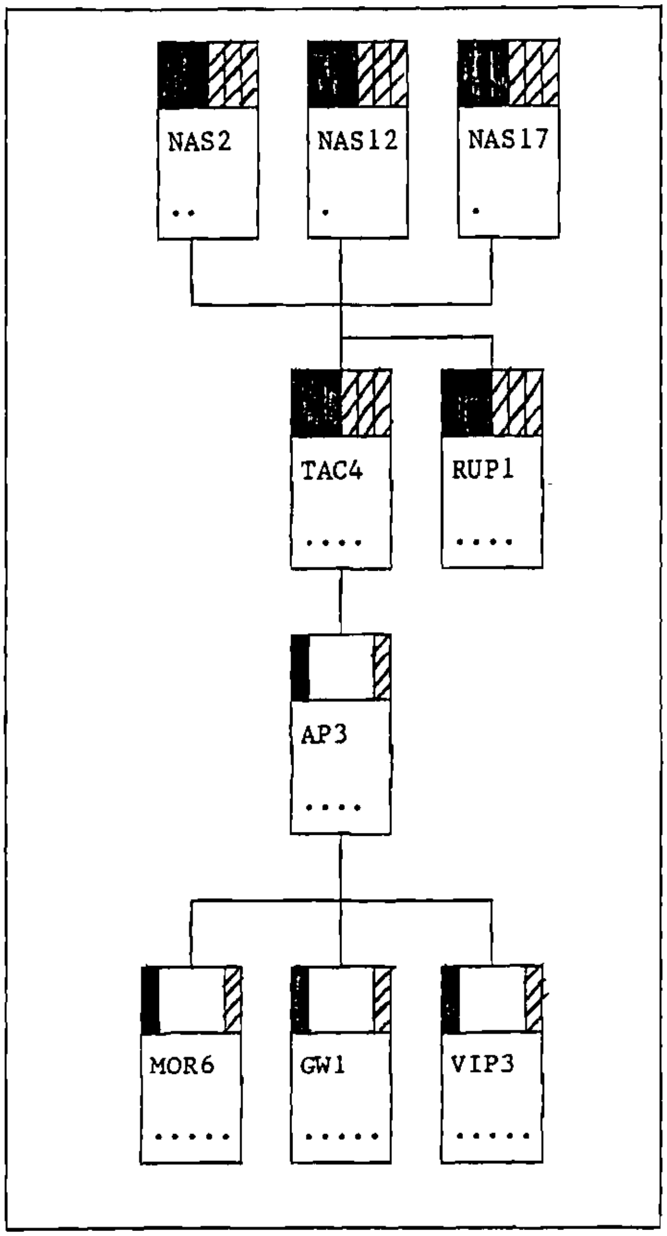
Figure 7

provides qualitative rather than quantitative information. Using the enlarged network of spigots transmitting data, the operator can see if there are problems with data transmission at any component in the equipment string supporting the spacecraft contact. Figure 8 gives and example.

The last command, DISPLAY MORE mission-name, is a higher level command than the others. Before displaying information, the hardware status of each piece of equipment supporting the satellite contact is checked. If a hardware failure is detected, then the equipment STATUS icon appears, otherwise, the FLOW icon appears. The MORE command can be considered "intelligent" in that it will select between the STATUS and FLOW icon; given current system state, this command will choose the display that the operator is more likely to want to see. If a component has a hardware failure, using the detailed STATUS icons, the operator can quickly determine which, if any, component has failed. If there is a problem but not a hardware failure, the FLOW icon provides information that can be used to identify a software failure.

When a mission ends, if its FLOW or STATUS icon is displayed, the icon will be erased automatically. The operator can erase either the FLOW or STATUS icon with an "ERASE BLOWUP" command.

In addition to the iconic display screen, the operator has another CRT available. The left screen provides alphanumeric windows that indicate the status and schedule of system components and aid the operator in various control function. Windows provide schedule



A flow icon for one of the spacecraft currently engaged in a pass. Flow is indicated above each bucket. Full flow takes the entire width of the bucket; here AP3 has a flow problem.

Figure 8

information that includes an overall GT-MSOCC spacecraft pass schedule (DISPLAY MSOCC SCHED) as well as individual schedules for each spacecraft (e.g., DISPLAY ERBE SCHED). Figure 9 gives examples of each type of window. In addition to spacecraft schedules, there are windows with schedules for each piece of GT-MSOCC equipment, (e.g., DISPLAY ERBE SCHED), (see Figure 9b) as well as status windows for each equipment class indicating whether an individual piece of equipment is in use or whether it has failed, (e.g., DISPLAY TAC STATUS). Figure 10 depicts an equipment status window.

There are three window permanently displayed on the left screen, a time window, a user input window, and a system message window. The left screen can support up to thirty other windows whose appearance is operator-controlled. Each operator-controlled window is numbered and the command "ERASE window-number" will erase a specific window. "ERASE ALL" will erase all of the operator-requested windows currently on the screen.

In addition to equipment and mission schedules and equipment status windows, the operator can request windows that provide information needed to perform operator control functions such as replacement of failed equipment or configuration of unscheduled spacecraft contacts. "HELP REPLACE TAC1", for example, provides the operator with all TACs available to replace TAC1. The operator can also request windows to aid in configuring an unscheduled pass. For example if the operator needs to configure a 10 minute unscheduled ERBE pass, "HELP CONFIGURE ERBE 10" displays a window showing what equip-

1 GT-MSOCC Schedule			
msn	up	down	equipment
DE	*	2:42	NAS28,23,21 RUP3 TAC6 AP3,5 MS1 VIP3 MOR13
WS-D	*2:38	2:43	NAS3,17 TAC3 AP5 VIP2 MOR9
AE-QL	2:41	2:47	NAS2,13 TAC2 AP1 CMS2 VIP1 MOR4
ISE	2:47	2:56	NAS2,30,12 RUP1 TAC4 AP3 CMS2 VIP3 MOR8
GEO	2:47	2:51	NAS26,22,15 RUP3 TAC2 AP1 CMS1 VIP2 MOR2
GSAT	2:54	3:02	NAS1,27,33 RUP2 TAC6 AP6 GW1 VIP2 MOR7
LNSAT	2:55	3:01	NAS6,12,31 RUP2 TAC5 AP2 GW2 VIP3 MOR3

This is the overall GT-MSOCC schedule. An asterisk in the "up" column indicates that the mission (msn) is currently engaged in a pass. An asterisk before the "up" time indicates that the mission should be engaged in a pass, but its equipment has not been configured.

Figure 9a

2	AP5	Schedule
msn	up	down
DE	*	2:42
WS-D	2:55	3:01
VENTR	3:01	3:07
WS-D	3:45	3:51
DSEI	4:01	4:09
VENTR	4:11	4:18

Each component in the GT-MSOCC system has a schedule.

Figure 9b

3 DE Schedule		
up	down	equipment
*	2:42	NAS28,23,21 RUP3 TAC6 AP3,5 CMS1 VIP3 MOR13
2:58	3:01	NAS26,24,7 RUP2 TAC8 AP4,3 CMS1 VIP3 MOR13
3:18	3:23	NAS9,25,8 RUP1 TAC6 AP6,7 CMS2 VIP3 MOR13
3:38	3:41	NAS30,11,13 RUP1 TAC6 AP6,7 CMS1 VIP3 MOR13
4:18	4:22	NAS11,24,17 RUP3 TAC8 AP7,6 CMS2 VIP2 MOR13
6:18	6:24	NAS26,5,21 RUP2 TAC1 AP7,3 CMS1 VIP2 MOR13

A sample schedule for DE, one of the spacecraft in the GT-MSOCC system.

Figure 9c

1	Status
TAC1	IDLE
TAC2	IDLE
TAC3	BUSY
TAC4	IDLE-FAILED
TAC5	IDLE
TAC6	IDLE
TAC7	IDLE
TAC8	BUSY

A status window may be called for each class of GT-MSOCC equipment, i.e., NAS, RUP, TAC, AP, GW, VIP, CMS. There is not a status window for MOR's or SPF's, however.

Figure 10

ment is needed to support an ERBE pass as well as window containing available components that the operator can use to configure the spacecraft contact. Figure 11 shows an example of the windows provided to the operator in response to the "HELP CONFIGURE" command.

Information about needed equipment for a specific mission, e.g., ERBE, can be obtained independently. "DISPLAY ERBE TEMPLATE" will provide a window listing the equipment needed to support an ERBE pass without providing information about specific available components. Figure 12 gives an example of this window.

The operator might also want a history of the event and alarm messages that have been sent to the system message window. "DISPLAY EVENTS" displays a window containing this information. This window is updated dynamically and contains messages sent within the last 5 minutes. Figure 13 is an example of the events/alarm window.

Most windows on the left screen are static; their content does not change once the window has been displayed. To obtain updated contents, the window must be requested again. The information in Figures 11 and 12 for example is static; updated information must be requested.

The keyboard and two CRTs display the icons and windows described above constitute the GT-MSOCC operator interface to this system. Details for GT-MSOCC operator supervisory control procedures follows.

6 ERBE Template		
NAS__	NAS__	NAS__
TAC__	RUP__	
AP __		
MOR5	GW__	

1 Free NASs		
NAS1	NAS2	NAS4
NAS5	NAS7	NAS8
NAS9	NAS10	NAS13
NAS17	NAS18	NAS19
NAS21	NAS23	NAS24
NAS25	NAS26	NAS27
NAS29	NAS30	NAS32

2 Free TACs		
TAC1	TAC2	TAC4
TAC7		

3 Free RUPs		
RUP1	RUP2	RUP3

4 Free APs	
AP2	AP6

5 Free GWs
GW1

Sample Response to a "HELP CONFIGURE ERBE 10" Command

Figure 11

1 ERBE Template		
NAS__	NAS__	NAS__
TAC__	RUP__	
AP__		
MOR5__	GW __	

Response to a Request to See a Mission Template
(DISPLAY ERBE TEMPLATE)

Figure 12

30 EVENT LOG	
00:24	Q 7690 Please configure ASTRO for 5 minutes.
00:25	DSEI support ended: deconfiguration complete.
00:25	WS-D configured automatically.
00:28	Unable to configure SOLAR: RUP2 unavailable.

The event and alarm log holds up to seven messages
and erases any message that is more than five minutes old.

Figure 13

GT-MSOCC Supervisory Control Procedures

As stated in the introduction, the GT-MSOCC operator has four major functions:

1) Supervision of spacecraft contacts currently being supported. This function has two subfunctions.

   a) Monitoring the data flow for each currently supported pass to ensure continuity and integrity of the data.

   b) In the event of problems with data flow, identifying and compensating, if possible, for equipment failures.

2) Compensation for automated schedule problems.

3) Response to requests for unscheduled spacecraft contacts.

4) Deconfigure all manually configured or reconfigured equipment strings.

Specific procedures and examples follow.

1. Supervisory Control of Current Spacecraft Contacts

The GT-MSOCC operator ensures that all GT-MSOCC equipment is functioning properly so that information from the satellite reaches the MOR (SPF) and any other terminal points in the equipment string supporting the pass, namely the RUP, CMS, GW and VIP. The operator begins by monitoring the system for problems. On the left terminal the operator may want to call the events/alarm page which keeps a record of important system events and messages to the operator. The command "DISPLAY EVENTS" will call the events/alarm window on the left screen. The left screen is also where the operator enters commands inside the window labeled "COMMAND" and receives messages inside the window labeled "MESSAGE". After the operator receives a message, it is immediately logged onto the events/alarm display page.

On the right terminal the mission icons indicate the data flow rate and quality as they reach terminal points in the GT-MSOCC system. A mission icon is displayed for each satellite that is currently engaged in a pass. Data are collected in a bucket, and bad data blocks, i.e., error blocks, are represented as red dots collecting at the bottom of the bucket.

Recall, the flow for the spigots represent the smallest data flow rate at any of the terminal points, i.e., MOR (SPF), GW, VIP, RUP, CMS, in the equipment string supporting the spacecraft contact. If the icon's flow is significantly reduced or terminated, the operator should immediately suspect a problem and request additional information.

Like the flow rate, the error block counts represented by the red dots at the bottom of the icon are dynamic. At any given time they represent, proportionately, the greatest number of block errors at one of the terminal points of the equipment string. A significant increase in the number of red dots is a warning to the operator to further investigate the data flow through the equipment string.

The mission icon with its qualitative flow and error representation provides the operator with a set of warning signals about possible equipment malfunctions. As long as the equipment supporting a spacecraft contact is operating effectively, the mission icon will have a full flow and a small number of errors. Decreased flow rate or increased red dots at the bottom of the bucket are likely indicators of equipment problems.

As indicated in the overview, GT-MSOCC equipment problems may be separated into two broad types: equipment failures and data transmission degradation. The former, equipment failure, is a situation in which a computer or communications system becomes completely inoperable. The cause of this type of problem is typically hardware failure. A component with failed hardware is usually easy to detect. Failed equipment terminates the data flow recorded at the point of failure and affects the data flow at every subsequent point in the equipment string supporting a pass. The operator should, if possible, immediately replace a component with a hardware failure.

The second type of GT-MSOCC problem, a transmission failure, is much harder to detect. Although these failures may take several forms, all involve a degradation in data transmission even though individual pieces of equipment appear to be functioning adequately. Such failures may be thought of as "soft" failures. The cause is typically a software problem in the mission-specific software at one of the GT-MSOCC pieces of equipment supporting the spacecraft contact.

GT-MSOCC transmission problems occur in one of three ways: full termination of data transmission, decreased data transmission rate, or a significant increase in error block count. The first problem results in a full termination of data transmission. In this situation, software problems at some piece of equipment terminate data processing. As with equipment hardware failure, this type of problem is comparatively easy to detect since data stop arriving at the point of transmission and affect the data flow at all subsequent points in the equipment string.

A related but more subtle problem is decreased rate of data flow. Given the number of NAScom lines supporting a pass, there is an expected data flow rate. A significant decrease in this rate is cause for further examination of related equipment. The operator must monitor displayed data flow through individual components in the equipment string to detect decreased flow rate. If a problem is confirmed, the operator should, if possible, replace the faulty equipment.

The last type of transmission problem is a high error block count in received data. A certain amount of error blocks is expected but a rapid increase in the number of error blocks requires the operator to more closely examine error propagation through the GT-MSOCC equipment string supporting the the pass in order to see if one of the pieces of GT-MSOCC hardware is causing errors. Subsequent sections provide specific operator procedures to detect each of these problems.

Detecting a Hardware Failure. Hardware failures can be detected on the right CRT screen. The mission spigot icons indicate which missions are currently engaged in a pass and give information about the amount and quality of data as they reach terminal points in the configuration strings for each pass. Each icon represents data flowing through the system as liquid flowing out of spigots into a bucket. The blue flow indicates telemetry (science data) flow, yellow indicates nontelemetry (satellite health and safety data) flow. When a component in the equipment supporting the mission fails, the flow coming out of the spigot icon decreases or stops. If flow from the spigot icon stops, the operator may type either "DISPLAY MORE mission-name" or "DISPLAY mission-name STATUS". The first command will automatically check the hardware status of all the equipment in the string supporting the pass. If one or more has a hardware failure, the STATUS icon will be displayed. If there is not a hardware failure, the FLOW icon will appear.

The operator may directly request the STATUS icon with the second command listed above. When the STATUS icon page is displayed, a large icon inside a rectangular box will appear on the right half of the screen showing what equipment is supporting the mission of interest. A failed component will be depicted by a red block. Components that have not failed will be depicted in green on the STATUS page display. A failed component needs offline maintenance and should be replaced if possible. If all components are green and flow is terminated, a component may have a data transmission rate error, rather than a hardware failure.

<u>Detecting a Software Failure</u>: <u>Decreased Data Transmission Rates</u>. Like hardware failures, data flow problems can also be detected by monitoring the mission spigot icon. If there is decreased or terminated flow at the mission icon, the operator may request the FLOW icon depicting data flow rate and error block counts

at each component in the equipment string supporting the spacecraft contact. As in detecting hardware failures, the "DISPLAY MORE mission-name" command can be used to first ensure that a hardware failure is not causing the problem. When all equipment is operating satisfactorily, the MORE command will display the FLOW icon. The FLOW icon can be requested directly with the command "DISPLAY mission-name FLOW".

Given a FLOW icon, to detect a software failure causing data transmission problems the operator should inspect the flow at each terminal point in the equipment string supporting the pass, i.e., RUP, MOR (SPF), GW, CMS, VIP, as well as other components constituting the equipment string. If information is flowing at full rate at all components preceding and including each of the terminal points, then flow from the icon spigots will be the entire width of a spigot. If flow is only the partial width of the a spigot, a data transmission rate problem is indicated. There will be some system jitter, so flow width that is only slightly less than the spigot width may not indicate any problem. When a component has a data transmission rate problem, its data flow will have a thinner width or be nonexistent when compared to flow width at previous components or components at the same level. A data transmission rate problem will propagate through the equipment string, so that a component's data transmission rate error will affect the data transmission rates of all subsequent components. Once a data transmission rate problem has been identified, the operator should attempt to replace the faulty component as quickly as possible.

Detecting a Software Failure: Too Many Error Blocks. The mission spigot icons also represent error block counts. The red dots at the bottom of the bucket indicate bad data blocks that have reached a terminal point of the equipment configuration, i.e., RUP, MOR (SPF), GW, CMS, VIP. The number displayed is the worst case, i.e., the terminal point with the highest relative error block count. A large number of error blocks may indicate a problem with one of the components supporting the satellite. There will be some system jitter, so a small number of slowly increasing error blocks (red dots) may not indicate any problems.

To see the number of error blocks at each component, enter the command "DISPLAY mission-name FLOW". To determine if a component ia generating too many error blocks, the GT-MSOCC operator compares the number of error blocks inside of one component with the number of error blocks inside of previous components and with components at the same level in the equipment string. A faulty component will contain a much higher number of error blocks than the preceding units or units at at the same level. Properly operating components at the same level should contain approximately the same amount of error blocks regardless of the status of previous components in the equipment string.

If multiple NAScom lines are in use, the number of error blocks inside NAS icons must be totaled before a comparison is made with the number of error blocks at the subsequent components in the equipment string. Once the problem has been resolved, the large icon is no longer necessary. To erase the large FLOW icon or the STATUS icon, enter "ERASE BLOWUP".

Replacing Faulty Components.

Once a component has been identified as faulty, it needs to be replaced, if possible, with another component of the same type. A replacement unit needs to be currently free, and also not scheduled for other use during the required time period. The command "HELP REPLACE equip-name" displays a window listing components that are operational and available during the time the faulty component is scheduled for use.

Once the GT-MSOCC operator identifies a replacement component, a command must be given to make the replacement. The command to replace a faulty component with an available component of the same type is given in the box below. Once the operator has manually replaced a component, the equipment string must be manually deconfigured at the conclusion of the pass. The deconfigure command is also

> Operator Command to Replace a Faulty Component and Manually Deconfigure a Mission
>
> REPLACE old-one new-one
> DECONFIGURE mission-name*
>
> *See Table 1 for a list of mission-name abbreviations.

shown in the box below.

For example, suppose AP1 has been detected as a faulty unit. The operator identifies AP5 as an available component by means of the "HELP REPLACE AP1" command; the command displays a window containing replacement equipment that has been provided with the command "HELP REPLACE AP1" and therefore is not in use, does not have a hardware failure, and is not scheduled for use during the needed time period. Given a replacement component, the GT-MSOCC operator makes the replacement with the command "REPLACE AP1 AP5".

A faulty component cannot always be replaced. MORs are spacecraft specific and are not interchangeable. Also, there may be no replacement available for other GT-MSOCC equipment.

If a component has a problem and cannot be replaced, the GT-MSOCC operator enters "ALERT" followed by the message. For example, "ALERT MOR6 IS GENERATING TOO MANY ERROR BLOCKS".

2. Compensation for Automated Schedule Problems

If a pass is scheduled to occur, but one of its needed components has failed, the GT-MSOCC operator receives a message, such as "UNABLE TO CONFIGURE ERBE: TAC3 UNAVAILABLE". The operator must attempt to identify a suitable replacement and manually configure the equipment string with the new component. The command, "HELP CONFIGURE mission-name", displays a window to aid the operator in carrying out this function. It calls up two windows on the left screen. One window displays a template that lists the equipment types that mission-name uses and shows which of the scheduled components is unavailable for use. A second window listing possible replacements for the unavailable component is also displayed.

Once a replacement unit is identified, the operator can manually configure the equipment string to support the scheduled spacecraft contact, specifying the replacement equipment. Since the equipment string has been manually configured, at the end of the pass the operator must also manually deconfigure the equipment. The commands to manually configure and deconfigure equipment are given below. The operator has about 3 minutes to configure and replace a problem with the automated schedule. If a replacement component cannot be found in that time, the pass will be removed from the mission pending list.

3. Responding to Special Requests

The GT-MSOCC operator may also be asked to configure an unscheduled, emergency pass for a given time duration or to respond to requests about the feasibility of immediately scheduling a

Operator Commands to Configure/Replace and Deconfigure Equipment String
CONFIGURE mission-name* REPLACE old-one new-one
DECONFIGURE mission-name*
*See Table 1 for a list of mission-name abbreviations.

particular spacecraft contact. The command, "HELP CONFIGURE mission-name time-duration", will display a window listing what equipment is needed as well as windows listing free components of that type available during time-duration. Moreover, the command will check to ensure that the proposed addition will not increase the total number of concurrently supported missions to more than five. The GT-MSOCC system is not capable of supporting more than five missions simultaneously. Violation of this constraint is considered a serious operator error.

If all of the needed equipment is available, the operator should respond positively to the question referencing the question number and, if the request was to actually configure the equipment, proceed to do so. Several examples of questions and operator answer are given below.

a)	Q	12345	Can ERBE be supported for 5 minutes?
	A	12345	YES, EQUIPMENT IS AVAILABLE.
b)	Q	4123	Please configure ERBE for 4 minutes.
	A	4123	NO, NOT ALL EQUIPMENT IS AVAILABLE.
c)	Q	8179	Can AE-D be supported for 3 minutes?
	A	8179	YES
d)	Q	7431	Please configure SS for 6 minutes.
	A	7431	OK

In the case of questions of type (b) or (d), after answering, if the required equipment is available, the operator should actually configure the equipment to support the request.

The command to configure an unscheduled spacecraft contact is entered in two lines. On the first line the operator enters the mission name and pass duration in minutes; on the second line the operator enters the equipment selected to support the pass. As with a configure-replace command, since the GT-MSOCC operator manually configured the pass, he/she must also manually deconfigure the equipment. When the pass is complete, the operator gives the command, "DECONFIGURE mission-name". The format of the configure and deconfigure commands is given in the box below. The time duration is given in minutes.

An example of a typical command sequence is

```
CONFIGURE ERBE 5 <RETURN>
NAS2 NAS3 NAS4 TAC1 RUP1 AP2 GW1 <RETURN>
DECONFIGURE ERBE
```

Operator Commands to Manually Configure and Deconfigure Equipment for a Special Request

```
CONFIGURE mission-name* duration-time <RETURN>
equip1, equip2, equip3,..., equipn <RETURN>

DECONFIGURE mission-name*
```

*See Table 1 for a list of mission-name abbreviations.

If the operator would like to see what equipment a satellite needs, but not for a specific time duration, "DISPLAY mission-name TEMPLATE" will give a window listing all equipment types needed to configure support for a given satellite.

To erase these windows the operator can use one of two commands. The "ERASE ALL" command can be used to erase all user-controlled windows currently displayed; the "ERASE window-number" command will erase the window identified by that number.

Deconfigure All Manually Configured or Reconfigured Equipment Strings

As stated above any equipment that has been manually configured or reconfigured due to a component failure during a pass or a problem with the automated schedule and control system must be manually deconfigured as quickly as possible. Failure to respond promptly means that configured equipment is unavailable for current use and may cause a subsequent problem with the automated schedule and control system. Operator-caused errors of this type are very serious.

GT-MSOCC Operator Function Priorities

As indicated above, the GT-MSOCC operator role is comprised of several operator functions:

1) monitoring the hardware status and data transmission quality of the components constituting the equipment strings supporting each currently active pass.

2) deconfiguring manually configured equipment.

3) compensating for problems encountered by the automatic schedule when attempting to configure reserved equipment for a previously scheduled pass.

4) responding either to requests for unscheduled mission support or to requests about the feasibility of unscheduled support. There is a priority implicit in these functions.

Under normal circumstancea, the operator is expected to monitor currently active passes. This function is preempted by any of the next three operator functions. The highest priority function is to deconfigure manually configured equipment. The GT-MSOCC operator should quickly deconfigure manually configured equipment. Failure to do so means that the configured equipment is unavailable for use, and if it is scheduled for immediate use, the current state will cause a subsequent automated schedule problem.

The next priority function is the compensation for problems with the automated schedule. When the operator is notified of an automatic schedule problem, other activities should cease until a replacement component is found and the mission is manually configured, or until the operator decides that no replacement component is available.

Compensating for failed equipment, i.e., equipment with either hardware or software failures, is the third priority function. Any time a hardware failure is detected or a software failure is suspected, the operator should attempt to confirm and replace the failed component.

Responding to *ad hoc* requests for special configurations is the lowest priority function. The operator should respond to requests as quickly as possible, but only if all currently supported passes appear to be operating satisfactorily and there are no missions pending that the automatic scheduling system is unable to configure.

A final comment on the overall operator function is needed. Operator-caused automatic schedule problems are considered serious operator errors. The operator should carefully select a component to replace a failed component in a current pass or to replace an unavailable component in a scheduled pass. The operator should ensure that the component is not already reserved for use at some time during the period for which the operator is making the manual replacement or configuration. When reponding to *ad hoc* requests, the operator should ensure that sufficient equipment is currently available and not previously scheduled for the period in question.

The second type of serious operator error is manually configuring a mission so that the total number of concurrently supported missions exceeds five. The GT-MSOCC system is not capable of supporting more than five passes simultaneously and the operator must ensure that this limitation is not violated.

Operator Errors

1. Failing to deconfigure an ended mission so that its equipment is needed, but not available for use, thus causing an automated scheduling problem.

2. Using equipment to
   a) replace a component
   b) configure a scheduled mission having an unavailable component
   c) configure a new pass

   when that equipment is already scheduled, and thus causing an automated scheduling problem.

3. Deconfigure a mission before its pass is completed.

4. Loading more than five missions concurrently. Also, manually configuring a mission without consulting the MSOCC schedule, so that more than five missions are scheduled concurrently. Note that the automatic scheduler never schedules more than five missions at once; this could only occur after the operator manually configures an unscheduled pass.

5. Manually configuring an unscheduled pass over a time when that spacecraft is already scheduled. The spacecraft specific MOR will not be available for the scheduled pass, thus causing an automated scheduling problem.

GT-MSOCC OPERATOR INPUTS

Operator Information Retrieval and Operational Aid Requests

1. Status

   DISPLAY mission-name STATUS
   A complete list of spacecraft abbreviations is given in Table 1.
   example: DISPLAY ERBE STATUS
   (note: mission-name must be currently engaged in a pass)

   DISPLAY component-type STATUS
   component-type = NAS, TAC, RUP, AP, GW, CMS, VIP

2. Flow

   DISPLAY mission-name FLOW
   example: DISPLAY ERBE FLOW
   (note: mission-name must be currently engaged in a pass)

3. More

   DISPLAY MORE mission-name
   example: DISPLAY MORE ERBE
   (note: mission-name must be currently engaged in a pass)

4. Schedules

   DISPLAY MSOCC SCHED

   DISPLAY mission-name SCHED
   examples: mission-name = ERBE, GEO, LNSAT

   DISPLAY component-name SCHED
   A complete list of component names is given on page 4.
   examples: component = TAC7, NAS23, RUP2

5. Events/alarm page

   DISPLAY EVENTS

6. Help Configure a Scheduled Mission and Display Available Equipment

   HELP CONFIGURE mission
   example: HELP CONFIGURE ERBE
   (note: mission must be pending, i.e., the mission must be unable to be scheduled automatically because a reserved component is not available)

7. Help Replace

   HELP REPLACE component-name
   A complete list of component names is given on page 4.
   MORs and SPFs do not have schedules.
   example: HELP REPLACE TAC1
   (note: component must be in use)

8. Help Configure an unscheduled mission and display available equip
    HELP CONFIGURE mission-name duration-in-minutes
    example: HELP CONFIGURE ERBE 10

9. Erase Icon and Window Commands
    ERASE* BLOWUP
    ERASE window-number
    ERASE ALL

*ERASE can be abbreviated with "E", "ER", and "ERA".

Operator Control Commands

1. To configure a scheduled mission and replace:
   CONFIGURE mission-name REPLACE old-component new-component

   Example: CONFIGURE ERBE REPLACE TAC6 TAC3

2. To replace a failed component in an ongoing pass:
   REPLACE old-component-name new-component-name

3. To manually configure an unscheduled mission:
   CONFIGURE mission-name minutes component component... *
   Example: CONFIGURE ERBE 9 <RETURN>
   NAS1 NAS2 NAS3 TAC1 RUP1 AP1...*
   *Do not specify MOR(SPF)
   (see Table 3)

4. To respond to a special request:
   A query-number message
   Example: A 13526 CAN NOT CONFIGURE ERBE
   A 5712 CAN CONFIGURE SS

5. To manually deconfigure a mission:
   DECONFIGURE mission-name
   Example: DECONFIGURE ERBE

6. To send an alert message:
   ALERT msg-string
   Example: ALERT MOR6 TOO MANY ERROR BLOCKS
   ALERT NO REPLACEMENT FOR RUP1

Note: Commands can be written upper or lower case and the words DISPLAY, REPLACE, CONFIGURE, ANSWER and HELP can be abbreviated their first letter or first two or three letters.

EXERCISES

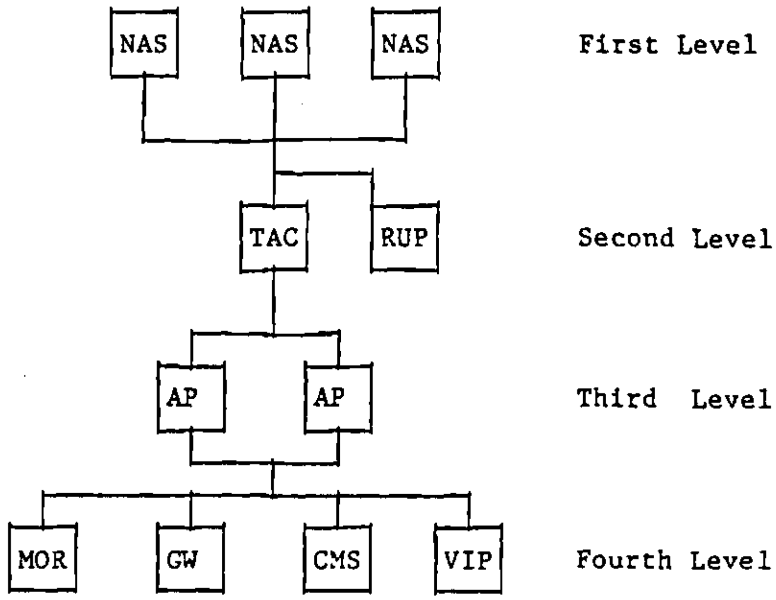


Figure 1

It is important to know the direction of data flow. The order of data flow is as follows. First, data flow through the NAScom lines. Second, data flow through the TAC and RUP processors. Third, data flow through the APs, and finally, data flow to the GW, CMS, VIP and MOR (or SPF).

The icon in Figure 2 shows data flow rate and quality as it reaches terminal points in the equipment string.

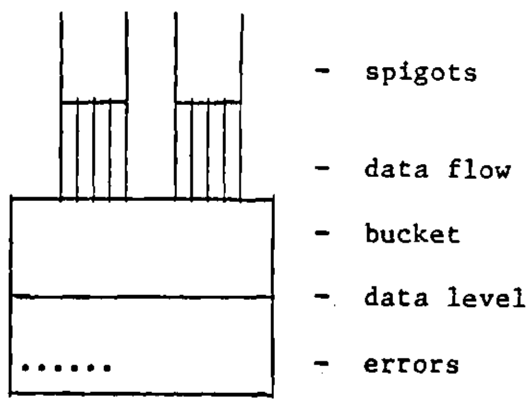
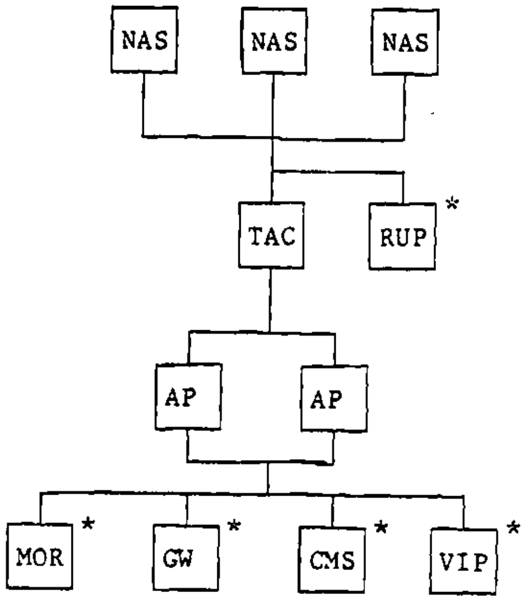


Figure 2

A faulty component may begin updating at one rate, and then start updating at a decreased rate. If data flow through a faulty component at a decreased rate, each subsequent unit will be affected. However, a faulty component which is a terminal point of the data flow stream does not affect previous components. The RUP, MOR, GW, CMS, VIP are terminal points of incoming data flow shown in Figure 3.



*Terminal points on the path of data flow.

Figure 3

Exercise 1

The icon below indicates full data flow at each unit in the configuration under consideration.

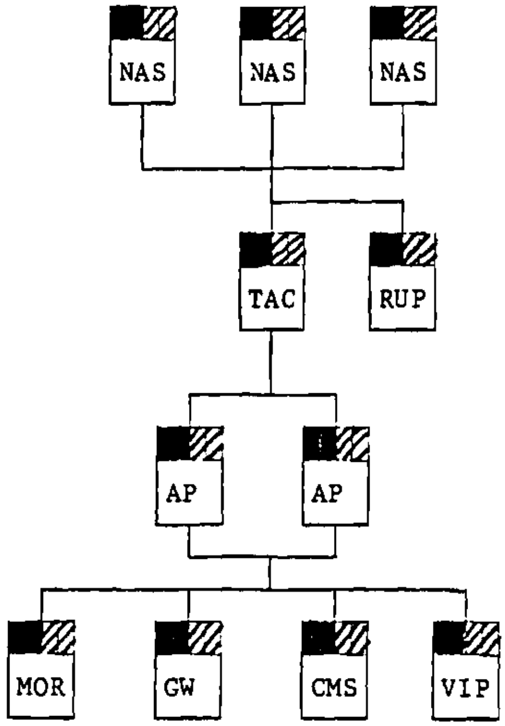


Figure 4

For the configuration above, describe the effect on the flow rates at all subsequent components after the failed component.

Q1: Suppose data begin flowing at a decreased rate at the MOR, which components could be causing the problem?

Q2: Suppose data begin flowing at a decreased rate at the RUP, which components could be causing the problem?

Q3: Suppose data begin flowing at a decreased rate at the CMS, which components could be causing the problem?

Assuming no other components faulty, consider the following questions.

Q1: Suppose the data flow through one NAS is at 50% of the normal rate: describe how the diagram would change.

Q2: Suppose the data flow through the TAC is at 50% of the normal rate: describe how the diagram would change.

Q3: Suppose the data flow through the GW is at 50% of the normal rate: describe how the diagram would change.

Q4: Suppose the data flow through the AP is at 50% of the normal rate: describe how the diagram would change.

Decreased flow rate is only one indication of a faulty unit. A faulty piece of equipment may create error blocks which are data blocks that are garbled or arrive out of sequence. Any error blocks created in a component are passed to subsequent components in the equipment string. For example, suppose a NAScom line is creating error blocks. The following effects could be expected to propagate throughout the system even if all other components are operating satisfactorily.

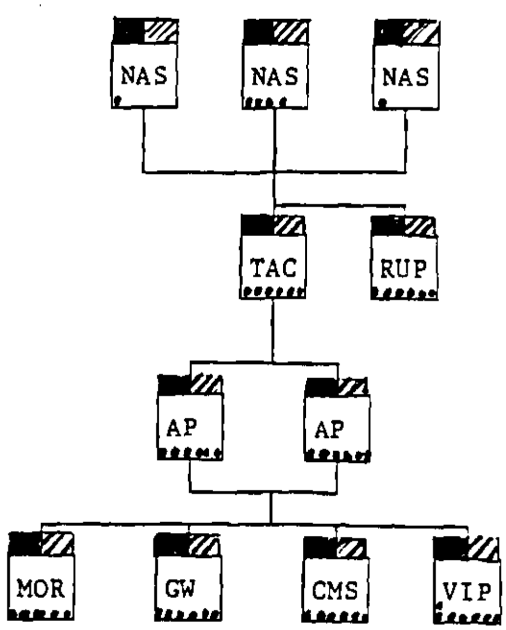


Figure 5

Exercise 2

Given the number of dots corresponding to received error blocks in the components specified below, fill in the following table showing how error blocks propagate through the system. (See Figure 5 for the equipment string).

Q1: 10 error blocks in a NAS

Q2: 10 error blocks in a TAC

Q3: 10 error blocks in an AP, 5 in other AP

Q4: 10 error blocks in a GW

	NAS	NAS	NAS	RUP	TAC	AP	AP	MOR	GW	CMS	VIP
Q1:	0	10	0	__	__	__	__	__	__	__	__
Q2:	0	0	0	__	10	__	__	__	__	__	__
Q3:	0	0	0	0	0	5	10	__	__	__	__
Q4:	0	0	0	0	0	0	0	__	10	__	__

In the above configuration, a more serious rate problem occurs when data flow completely stops at a unit. If data flow stops at a NAS, for example, there would be full flow at the third NAS line. This would result in 2/3 of the expected flow at the RUP, TAC, AP, GW, CMS, VIP and MOR.

Question: Suppose data flow stopped at the MOR. Which components could be responsible? (in the configuration shown in Figure 5)

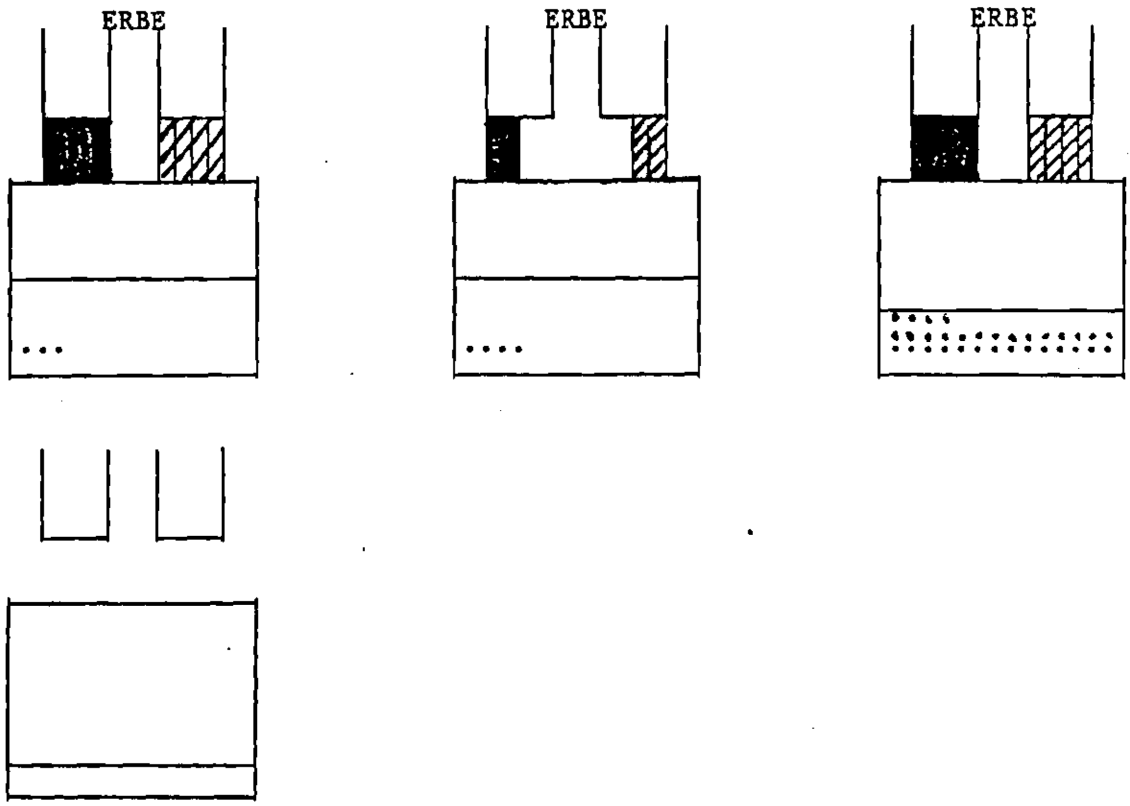
Question: Why couldn't a NAS be responsible?

Question: If the configuration had only one AP, could the AP be responsible?

A low flow rate of a high error block count at a terminal point in the input data stream, such as the MOR, may be an indication of a problem with the MOR, with a preceding unit or it may be no problem at all. There is some system jitter, so if errors are evenly dispersed through the configuration, there probably is not a problem.

Exercise 3

Failure Detection Exercise:
Which of these terminal flow icons indicate potential trouble in the corresponding equipment configuration?



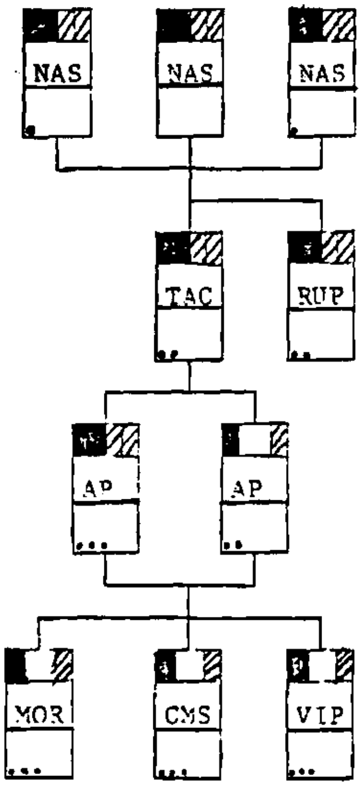
To view a detailed icon representing the satellite's equipment configuration type:

DISPLAY MORE ERBE
(or mission-name of satellite of concern)

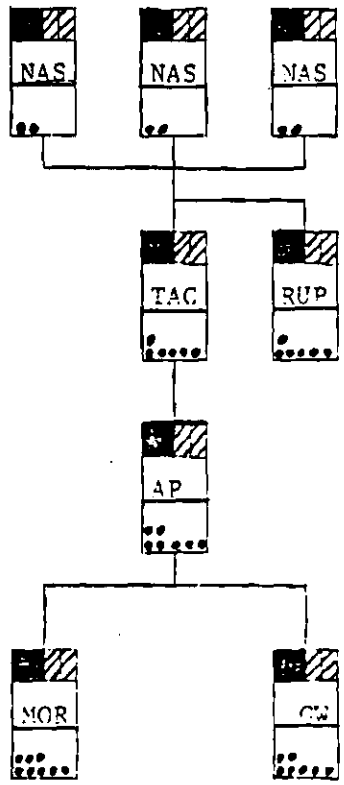
Hint: To find faulty components, compare each component with previous components <u>and</u> components on the same level. Look for significantly more errors or significantly less flow.

Exercise 4

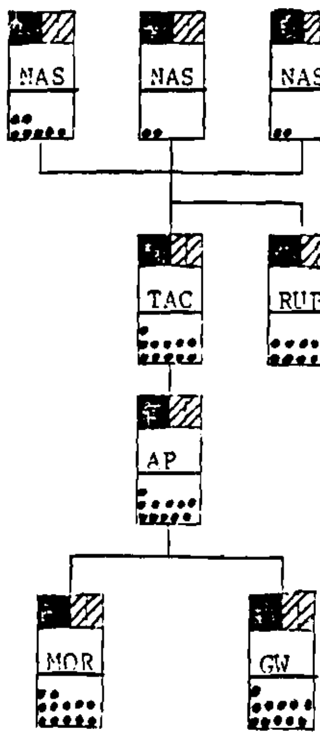
Which component(s) is faulty?



(a)



(b)



(c)

APPENDIX D

Consent and Debriefing Forms

Subject Consent Form
Human-Computer Interaction in Supervisory Control Tasks:
Cognitive Models and Computer Aids
(E24-606)

Date : ____________________

I, ____________________, am voluntarily participating in the NASA sponsored research project supervised by Dr. C. M. Mitchell. I agree to participate in a total of twelve experimental sessions, occurring on consecutive days and lasting approximately one hour each. I understand that I will be paid $ 60 for my participation after the completion of the twelfth session. Finally, I understand that failure to complete all twelve sessions is likely to result in my receiving no payment for any time spent participating in the experiment.

____
Signature   Date

____
Social Security Number

Debriefing Questions for GT-MSOCC

Date ______________________

Subject ______________________

Condition ______________________

1. What did you think of the overall operator tasks?

2. Which tasks or functions were especially hard? Why?

3. Which tasks or functions were especially easy? Why?

4. What attributes of the interface, i.e., information displays and operator commands, made operator tasks more difficult? Why?

5. What attributes of the interface made operator tasks easier? Why?

6. What additional help could be provided to make the operator's job easier?

7. Overall comments or reactions.

APPENDIX E

Procedures to Approximate the F Statistic

Approximating an F Statistic

When exact tests do not exist, an F statistic can be approximated using linear combinations of mean squares chosen such that

$$MS' = a\ MS_r + \dots + b\ MS_s$$

$$MS'' = y\ MS_u + \dots + z\ MS_v$$

where the coefficients and mean squares are chosen so that $MS' - MS''$ is equal to the effect considered in the null hypothesis (Montgomery, 1984). The test statistic is then

$$F = MS' / MS'' \sim F_{p,q}$$

which is distributed as $F_{p,q}$ where

$$p = (a\ MS_r + \dots + b\ MS_s)^2 / (a\ MS_r)^2 + \dots + (b\ MS_s)^2$$

$$q = (y\ MS_u + \dots + z\ MS_v)^2 / (y\ MS_u)^2 + \dots + (z\ MS_v)^2$$

A numerical example is provided below that demonstrates procedures to develop the approximate F statistic to test the effect primarily of of display condition. Expected mean squares are provided by the GLM procedure of SAS statistical software. These expected mean squares are used to develop MS' and MS''.

For this example, suppose the SAS output provides the following information:

SOURCE	EXPECTED MEAN SQUARE	MEAN SQUARE
COND	VAR(ERROR) + 1.35*VAR(SESSxSUBJ(COND)) + 9.25*VAR(CONDxSESS) + 6.34*VAR(SUBJ(COND)) + Q(COND)	1,868,469
SESS	VAR(ERROR) + 1.74*VAR(SESSxSUBJ(COND)) + 15.40*VAR(CONDxSESS) + 30.78*VAR(SESS)	1,950,062
SUBJ(COND)	VAR(ERROR) + 1.66*VAR(SESSxSUBJ(COND)) + 9.54*VAR(SUBJ(COND))	648,252
COND*SESS	VAR(ERROR) + 1.74*VAR(SESSxSUBJ(COND)) + 15.37*VAR(CONDxSESS)	312,550
SESS* SUBJ(COND)	VAR(ERROR) + 2.05*VAR(SESSxSUBJ(COND))	129,995
ERROR	VAR(ERROR)	125,637

Since the effect of condition is being analyzed in this example, Q(COND) is the model parameter considered in the null hypothesis. The goal is to develop MS' and MS'', such that $MS' - MS'' = Q(COND)$.

<u>Step 1</u>: Begin by including $1*MS_{COND}$ in MS'.

Thus all terms in the E(MS) for COND (shown above) are in MS'.

<u>Step 2</u>: Put the same VAR(SUBJ(COND)) term in MS' and MS''.

Currently, 6.34*VAR(SUBJ(COND)) is in MS', and there is no VAR(SUBJ(COND)) term in MS''. To put 6.34*VAR(SUBJ(COND)) in MS'', multiply 6.34/9.54 (or .66) by $MS_{SUBJ(COND)}$ and include in MS''.

Note: .66*VAR(ERROR) + .66*1.66*VAR(SESSxSUBJ(COND)) are now also terms included in MS''.

Step 3: Put the same VAR(CONDxSESS) term in MS' and MS''.
Currently, 9.25*VAR(CONDxSESS) is in MS', and there is no VAR(CONDxSESS) term in MS''. To put 9.25*VAR(CONDxSESS) in MS'', multiply 9.25/15.37 (or .60) by $MS_{CONDxSESS}$ and add this to MS''.
Note: .60*VAR(ERROR) + .60*1.74VAR(SESSxSUBJ(COND)) are now also added to MS''.

Step 4: Put the same VAR(SESSxSUBJ(COND)) term in MS' and MS''.
Currently, 1.35*VAR(SESSxSUBJ(COND)) is in MS' (Step 1), and 2.15*VAR(SESSxSUBJ(COND)) is in MS'' (Step 2 and Step 3). To put 2.15*VAR(SESSxSUBJ(COND)) in MS', multiply .80/2.05 (or .39) by $MS_{SESSxSUBJ(COND)}$ and add this to MS'.
Note: this step also add .39*VAR(ERROR) to MS'.

Step 5: Put the same VAR(ERROR) term in MS' and MS''. Currently, 1.39*VAR(ERROR) is in MS' (Step 1 and Step 4), and 1.26*VAR(ERROR) is in MS'' (Step 2 and Step 3). To put 1.39*VAR(ERROR) in MS'', multiply 1.39–1.26 (or .12) by MS_{ERROR} and add this to MS''.
Note: no terms other than .12*VAR(ERROR) are added to MS''.

The final result is:

$$MS' = MS_{CONDITION} + .39 * MS_{SESSxSUBJ(COND)}$$

$$= 1919167$$

$$MS'' = .66 * MS_{SUBJ(COND)} + .60 * MS_{CONDxSESS} + .12 * MS_{ERROR}$$

$$= 630452$$

These linear combinations of mean squares result in:

$MS' =$

1.39 * VAR(ERROR) + 2.15 * VAR(SESSxSUBJ(COND))

+ 9.25 * VAR(CONDxSESS) + 6.34 * VAR(SUBJ(COND)) + Q(COND)

$MS'' =$

1.39 * VAR(ERROR) + 2.15 * VAR(SESSxSUBJ(COND))

+ 9.25 * VAR(CONDxSESS) + 6.34 * VAR(SUBJ(COND))

Note that MS' and MS'' are identical except for the Q(COND) term. Thus, $MS' - MS'' = Q(COND)$ as desired. Thus, the test statistic is

$$F = MS' / MS'' = 1919167/630452 = 3.04$$

REFERENCES

Bury, K. F., Davies, S. E. and Darnell, M. J. Window Management: A Review of Issues and some Results from User Testing. IBM Human Factors Center, San Jose, California, June 1985.

Card, S. K., Pavel, M. and Farell, J. E. Window-Based Computer Diaglogues. Conference papers from INTERACT'84 - First IFIP Conference on Human Computer Interaction, Vol. 1, 1984, pp. 355-359.

Foley, J. D. and Van Dam, A. Fundamentals of Interactive Computer Graphics. Reading, Massachusetts: Addison-Wesley Publishing Company, 1982.

Goldberg, A. and Kay, A. Smalltalk-72 Instruction Manual. Xerox PARC, Palo Alto, California, 1976.

Goldberg, A. and Robson, D. SMALLTALK-80 The Language and its Implemtation. Addison-Wesley, 1983.

Goodstein, L. P. Discriminative Display Support for Process Operators. In J. Rasmussen and W. B. Rouse (Eds.), Human Detection and Diagnosis of System Failures. New York, New York: Plenum Press, 1981, pp. 433-449.

Gould, L. and Finzer, W. A Study of TRIP: A Computer System for Animating Time-Rate-Distance Problems. International Journal of Man-Machine Studies, Vol. 17, 1982, pp. 109-126.

Kantowitz, B. H. and Sorkin, R. D. Human Information Processing. New York: John Wiley & Sons, 1983, pp 164-191.

Loftus, G. R. and Loftus, E. F. Human Memory: The Processing of Information. Hillsdale, New Jersey: Lawrence Erlbaum Associates, Inc., 1976.

Meyrowitz N. and Moser, M. BRUWIN: An Adaptable Design Strategy for Window Manager/Virtual Terminal Systems. ACM SIGOPS Conference, 1981, pp. 180-189.

Miller, G. A. The Magic Number Seven, Plus or Minus Two. Psychological Review, 1956, Vol. 63, pp. 81-97.

Mitchell, C. M. An Approach to Modeling Human-Computer Interaction in Supervisory Control Systems. Center for Man-Machine Research Technical Report, GIT, Allanta, GA, Contract number NAS5-28575, September 1985.

Mitchell, C. M. and Miller, R. A. A Discrete Control Model of Operator Function: A Methodology for Information Display Design, IEEE Transactions on Systems, Man, and Cybernetics, in press.

Montgomery, D. C. Design and Analysis of Experiments (2cd ed.). New York, New York: John Wiley & Sons, 1984, pp. 254-259.

Moray, N. The Role of Attention in the Detection of Errors and the Diagnosis of Errors in Man-Machine Systems. In J. Rasmussen and W. B. Rouse (Eds.), Human Detection and Diagnois of System Failures. New York, New York: Plenum Press, 1981, pp. 185-198.

Murray, J. T., Hakkinen, M. T. and Mackraz, J. D. Window Design for a Professional Workstation Design in the Office Environment. Proceedings of the Human Factors Society, 28th Annual Meeting, 1984, p. 423.

NASA, Multisatellite Operations Control Center (MSOCC) Data Operations Control System (DOCS) Operator Interface Design Review, Sponsored by NASA Goddard Space Flight Center, Greenbelt, Maryland. Contract NAS 5-27555, July 1983.

Pike, R. Graphics in Overlapping Bitmap Layers. ACM Transactions on Graphics, Vol. 2, No. 2, April 1983, pp. 135-160.

Purvy, R., Farrell, J. and Klose, P. The Design of Star's Records Processing: Data Processing for the Noncomputer Professional. ACM Transactions on Office Information Systems, Vol. 1, No. 1, January 1983, pp. 3-24.

Rasmussen, J. Strategies for State Identification, and Design of Computer-Based Support Systems. In W. B. Rouse (Ed.), Advances in Man-Machine Systems Research, Vol. 1, Greenwich, Connecticut: JAI Press Inc., 1984, pp. 139-193.

Rassmussen, J. and Lind, M. Coping with Complexity (Riso-M-2293). Riso National Laboratory, Roskilde, Denmark: Electronics Department, June 1981, pp. 70-91.

Satterwaitte, F. E. An Approximate Distribution of Estimates of Variance Components. Biometrics Bulletin, Vol. 2, pp. 110-114.

Sheridan, T. B. Toward a General Model of Supervisory Control. In T. B. Sheridan and G. Johannsen (Eds.), Monitoring Behavior and Supervisory Control, New York, New York: Plenum Press, 1976, pp. 271-281.

Sheridan, T. B. Supervisory Control of Remote Manipulators, Vehicles and Dynamic Processes: Experiments in Command and Display Aiding. In W. B. Rouse (Ed.), Advances in Man-Machine Systems Research, Vol. 1, Greenwich, Connecticut: Jai Press Inc., 1984, pp. 49-137.

Spector, P. C., Goodnight, J. H., Sall J. P. and Sarle W. S. The GLM Procedure. SAS User's Guide: Statistics, Version 5 Edition. Cary, North Carolina: SAS Institute Inc., 1985, pp. 433-506.

Stallman, R., Weinreb, D and Moon, D. Lisp Machine Window System Manual. Lisp Machine Inc., Culver City, CA., August, 1983.

Streveler, D. J. and Wasserman, A. I. Quantitative Measures of the Spatial Properties of Screen Designs. INTERACT'84 - First IFIP Conference on 'Human Computer Interaction', Vol 1, 1984, 125-133.

Teitelman, W. Interlisp Reference Manual. Xerox Corporation, Palo Alto, California, 1974.

Weiser, M., Torek, C., Trigg, R. and Lyle, J. The Maryland Window System, The University of Maryland, 1983.

Wickens, C. D. Memory. Engineering Psychology and Human Performance, Columbus, Ohio: Charles E. Merrill Publishing Company, 1984, pp. 226-229.

Williams, G. The Lisa Computer System. BYTE, Vol. 8, No. 2, 1983, pp. 35-50.

Woods, D. D. Visual Momentum: A Concept to Improve the Cognitive Coupling of Person and Computer. International Journal of Man-Machine Studies, Vol. 21, 1984, pp. 229-244.

Wong, P. C. S. and Reid, E. R. FLAIR - User Interface Dialogue Design Tool. Computer Graphics, Vol. 16, No. 3, July 1982, 87-98.