NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

DEVELOPMENT OF A PROTOTYPE
MULTICHANNEL COMMUNICATION NETWORK
MAINTENANCE EXPERT SYSTEM

by

Joseph S. Yavorsky

March 1987

Thesis Advisor Carl R. Jones

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b RESTRICTIVE MARKINGS
2a SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited
2b DECLASSIFICATION/DOWNGRADING SCHEDULE	
4 PERFORMING ORGANIZATION REPORT NUMBER(S)	5 MONITORING ORGANIZATION REPORT NUMBER(S)
6a NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b OFFICE SYMBOL (If applicable) Code 54	7a NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000	7b ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000
8a NAME OF FUNDING/SPONSORING ORGANIZATION US Army Signal Center 8b OFFICE SYMBOL (If applicable) ATZH-CD	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c ADDRESS (City, State, and ZIP Code) Directorate of Combat Developments Ft. Gordon, Georgia 30905	10 SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO / PROJECT NO / TASK NO / WORK UNIT ACCESSION NO

11 TITLE (Include Security Classification)
DEVELOPMENT OF A PROTOTYPE MULTICHANNEL COMMUNICATIONS NETWORK MAINTENANCE EXPERT SYSTEM

12 PERSONAL AUTHOR(S)
Yavorsky, Joseph S.

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year Month Day)	15 PAGE COUNT
Master's Thesis	FROM ____ TO ____	1987 March	84

16 SUPPLEMENTARY NOTATION

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Multichannel Communications, CACSMAM, Communications Maintenance, Knowledge Expert System Engineering

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

This thesis involves the development of a small prototype microcomputer-based expert system to aid the Battalion Maintenence Officer (or staff) of a division signal battalion allocate resources when a communications node fails. This decision aid is designed to "fill the gap" between those automated systems designed to reroute circuits and those designed to diagnose equipment failures.

The system models a multichannel network as employed by a division signal battalion. It is limited to only the multichannel equipment itself and not any other network components (patch panels, switchboards, etc.). The assumption is that troubleshooting has taken place and the system failure is due to multichannel equipment failure in an AN/TRC 145 Radio Terminal.

This system is conceived as part of an integrated automated management system to aid the controlling node in managing the battlefield communications network more effectively. It is called the Computer Aided Communication System Maintenance Manager (CACSMAM). CACSMAM consists of approximately 185 production rules, written using the M.1 Knowledge System Development Tool (version 2.0) by Teknowledge, Inc and requires 338K bytes of RAM on an IBM PC compatable computer running PC DOS 2.1 or higher.

20 DISTRIBUTION/AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a NAME OF RESPONSIBLE INDIVIDUAL Carl R. Jones	22b TELEPHONE (Include Area Code) 408-646-2767 22c OFFICE SYMBOL Code 54JS

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete. — SECURITY CLASSIFICATION OF THIS PAGE

Approved for public release; distribution is unlimited.

Development of a Prototype Multichannel Communication Network Maintenance Expert System

by

Joseph S. Yavorsky
Captain, United States Army
B.S., United States Military Academy, 1978

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS TECHNOLOGY
(Command, Control and Communications)

from the

NAVAL POSTGRADUATE SCHOOL
March 1987

ABSTRACT

This thesis involves the development of a small prototype microcomputer-based expert system to aid the Battalion Maintenence Officer (or staff) of a division signal battalion allocate resources when a communications node fails. This decision aid is designed to "fill the gap" between those automated systems designed to reroute circuits and those designed to diagnose equipment failures.

The system models a multichannel network as employed by a division signal battalion. It is limited to only the multichannel equipment itself and not any other network components (patch panels, switchboards, etc.). The assumption is that troubleshooting has taken place and the system failure is due to multichannel equipment failure in an AN/TRC 145 Radio Terminal.

This system is conceived as part of an integrated automated management system to aid the controlling node in managing the battlefield communications network more effectively. It is called the Computer Aided Communication System Maintenance Manager (CACSMAM). CACSMAM consists of approximately 185 production rules, written using the M.1 Knowledge System Development Tool (version 2.0) by Teknowledge, Inc and requires 338K bytes of RAM on an IBM PC compatable computer running PC DOS 2.1 or higher.

THESIS DISCLAIMER

The reader is cautioned that computer programs developed in this research may not have been exercised for all cases of interest. While every effort has been made, within the time available, to ensure that the programs are free of computational and logic errors, they cannot be considered validated. Any application of these programs without additional verification is at the risk of the user.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to sincerely thank CPT Richard Routh, Ph.D., of the Artificial Intelligence Center for Training at Ft. Gordon, Georgia, for taking two weeks to teach me most of what I know about M.1 and expert systems. Also, without his invaluable assistance, the funding for this research, as well as the research itself would most likely not have been possible.

I would also like to extend my thanks to Captain Don Howard of the Directorate of Combat Development, Captain John Schoedel of the Office of the Chief of Signal, and Captain Paul Rossbach, instructor in the Senior Leadership Department of the USA Signal School, all of whom volunteered their own valuable time to assist me immensely in the verification of the knowledge base in CACSMAM.

I. INTRODUCTION

A. GENERAL

As technology advances, so does the art of warfare. Increasingly complex weapon systems and an ever-increasing pace of battle are placing a premium on fast, well-informed decisions by those in charge. In order for sound decisions to be made and dessiminated, a good command and control (C2) system must be in place. The official definition of a C2 system taken from JCS Publication 1 is:

> A Command and Control System consists of the facilities, equipment, communications, procedures, and personnel essential to a commander for planning, directing, and controlling operations of assigned forces pursuant to the missions assigned. [Ref. 1]

The overall structure of a C2 system, once emplaced, tends to be static. However, the personalities, equipment, and environment within and around the system tend to be dynamic. Therefore, the C2 system must be adjusted continually to account for changes.

At the Army division level, the division commander is in charge of the overall C2 system within the division and as such is responsible for making sure it works. The members of the division staff have the delegated responsibilities of making sure individual parts of the C2 system work by themselves and in conjunction with the rest of the system. This thesis concerns itself primarily with the maintenance of the communications network supporting the division commander's C2 system.

Maintenance of the facilities and equipment which provide the communications is crucial in maintaining the ability of the division commander to control his forces. The Division Signal Officer, who is also the signal battalion commander, is responsible for maintaining the divisional communications network. That responsibility incorporates training, frequency management, cryptological material management, communications asset management, as well as pure logistical considerations of supply, maintenance and transport of the personnel and equipment essential to the communications networks [Ref. 2: p. 114].

Given the complexity of the battlefield environment, the signal battalion commander must make sound, fast decisions in order to maintain those communications essential to the tactical commanders. However, because of the large amount of information needed to make a good decision, it is almost essential for the human decision maker to have assistance in choosing a course of action if he is to take advantage of all the data available. One of the most promising means of aiding him involves the use of the artificial intelligence technique known as an ***expert system*** [Ref. 3].

The signal battalion commander maintains the division communications system with the aid of the Battalion Maintenance Officer (BMO), who is on the battalion staff. A prototype expert system known as the *C*omputer *A*ided *C*ommunication *S*ystem *Ma*intenance *M*anager (CACSMAM) has been developed as part of this thesis to assist the BMO fulfill his/her responsibilities.

B. APPLICATION OF CACSMAM

The Computer Aided Communication System Maintenance Manager (CACSMAM) is designed as a prototype microcomputer-based expert system to aid the Battalion Maintenance Officer (or his staff) of the divisional signal battalion make the best decision as to what action to take when a multichannel communication node fails. This decision aid is envisioned as a part of an integrated automated management network combining that system designed to reroute the communication circuits around the failed node, that system designed to diagnose the cause of the equipment failure, and CACSMAM, which will determine how to reestablish the node through the redistribution of available assets. Figure 1.1 illustrates how the three expert systems might interact to help manage the multichannel network when the receiver fails at a communications node.

The division communications system consists of several sites, each with several communications shelters interconnected to such a network as shown in Figure 1.2 [Ref. 4: p. 20]. At the center of the site lies the AN/TSC 76 Patching Communications Center which routes all the connections between the various other assemblages on the site (and is the location of the Communications Electronics System Element (Patch), to be discussed later). The other assemblages shown in the figure are the AN/TSC 58, Telecommunications Center, the AN/GSQ 80 Message Center, the AN/TTC 23 Telephone Central Office (Manual), the AN/GRC 142 Radio Teletypewriter, the AN/VRC 49 Radio-Wire Integration Center, the AN/MSC 31 Operations Center (the location of the Communication System Control Element, also to be discussed later),

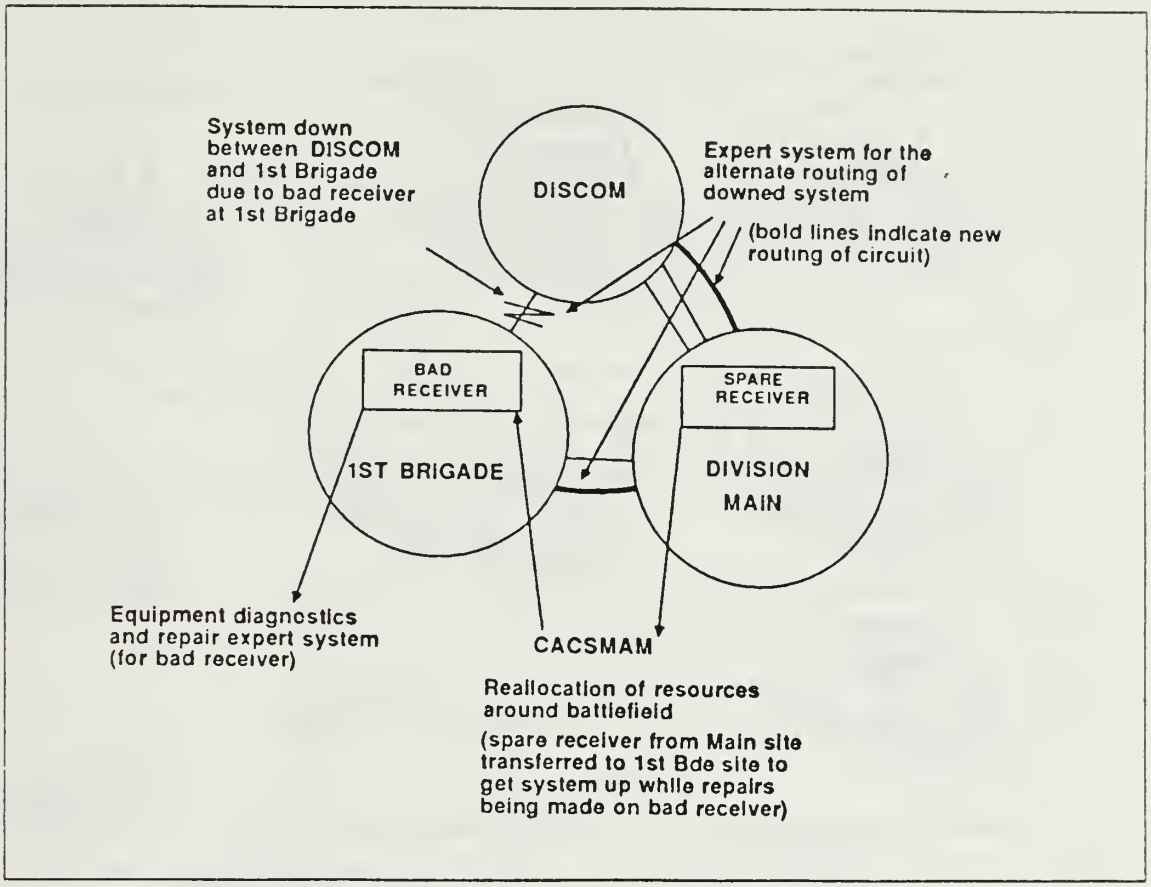


Figure 1.1 Interaction of CACSMAM with Other Expert Systems.

and the AN/TRC 145 Radio Terminal. The AN/TRC 145 is the multichannel radio van which is the backbone of the division communications system and the maintenance concern of CACSMAM.

When a problem occurs within the network, finding the cause can involve complex troubleshooting techniques. As shown in Figure 1.2, several pieces of equipment can be involved in the problem. After the troubleshooting has been done by the operator and his/her supervisors, actions are then taken to rectify the problem, either by repair or replacement of components. CACSMAM was designed on the assumption that troubleshooting has taken place and the fault has been isolated to the multichannel terminal and cannot be fixed solely through operator adjustment or repair of the equipment within the shelter but requires replacement of a component.

Chapter II provides an introduction to the Army Division Tactical Multichannel Communications Network and the system set up to manage it. Also included is a brief

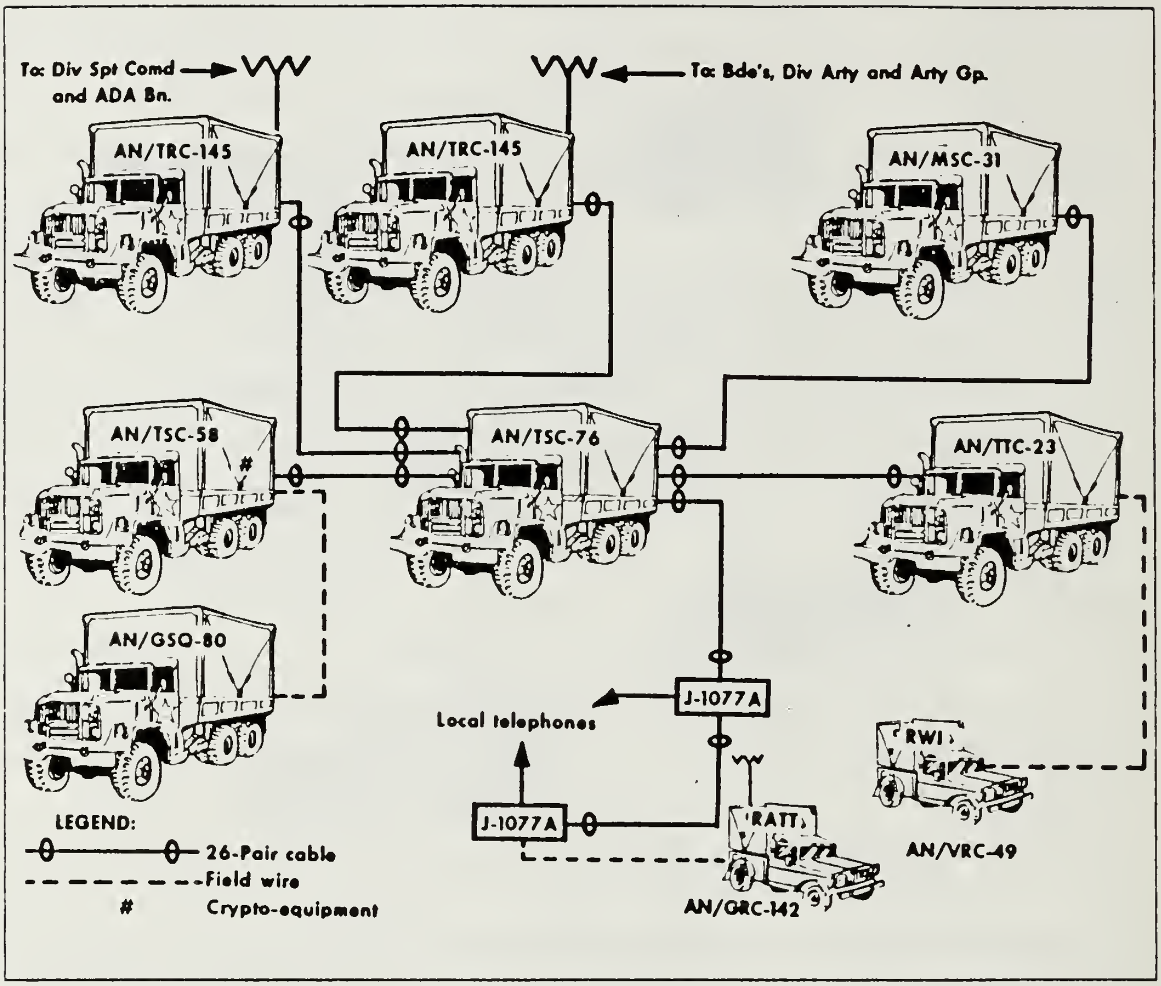


Figure 1.2 Typical Site Configuration of a Communications Node.

discussion of expert systems and the M.1 Knowledge System Development Tool in particular. In Chapter III the author discusses the sequence of events used to develop CACSMAM and some of the key points in its development. Chapter IV provides parts of a sample session of CACSMAM and gives the author's conclusions and recommendations with regards to the development of CACSMAM, expert systems and their fielding, and offers some suggestions for further work.

II. BACKGROUND

Prior to outlining the development of CACSMAM itself, the reader must understand enough about the system used by the signal battalion commander to manage the division communications system in order to get a feel for where CACSMAM fits in. Also, the reader must understand enough about expert systems and M.1 itself in order to follow the developmental procedures used for producing CACSMAM. This chapter explains the communications system and it's management, expert systems and M.1 in preparation for Chapter III which details the development of CACSMAM itself.

A. COMMUNICATIONS ELECTRONICS MANAGEMENT SYSTEM (CEMS)

The signal battalion commander has four primary managerial echelons of mission planning and execution which make up the Communications Electronics Management System (Figure 2.1). The Communication System Planning Element (CSPE) consists of the division C-E officer (the battalion commander), and the assistant division C-E officer and his staff. The CSPE is responsible for performing the communications requirements analysis and system planning for the tactical operations of the division. These functions are performed in close coordination with the signal battalion operations officer, the S3. The requirements generated by the CSPE form the basis of the commander's directives and orders to the signal battalion. [Ref. 4: p. 17]

The second echelon is the Communications System Control Element (CSCE), commonly referred to as simply the SYSCON. The CSCE consists of the signal battalion S3 and his operations staff and is responsible for the design, modification, and management of the division communications system which is installed, operated and maintained by the signal battalion. The functions of the CSCE include network system and circuit design, engineering, records keeping, reporting and supervision. The second major set of functions include the allocation and control of the signal battalion resources, and the monitoring of the operational status of systems and circuits. The information on the status of equipment, spare parts stockage, location of parts, maintenance teams, and fuel status are all maintained by a separate staff element known as the Battalion Logistics Operations Center (BLOC). The information provided by the BLOC enables the CSCE to allocate resources and supervise the system. [Ref. 4: p. 17]

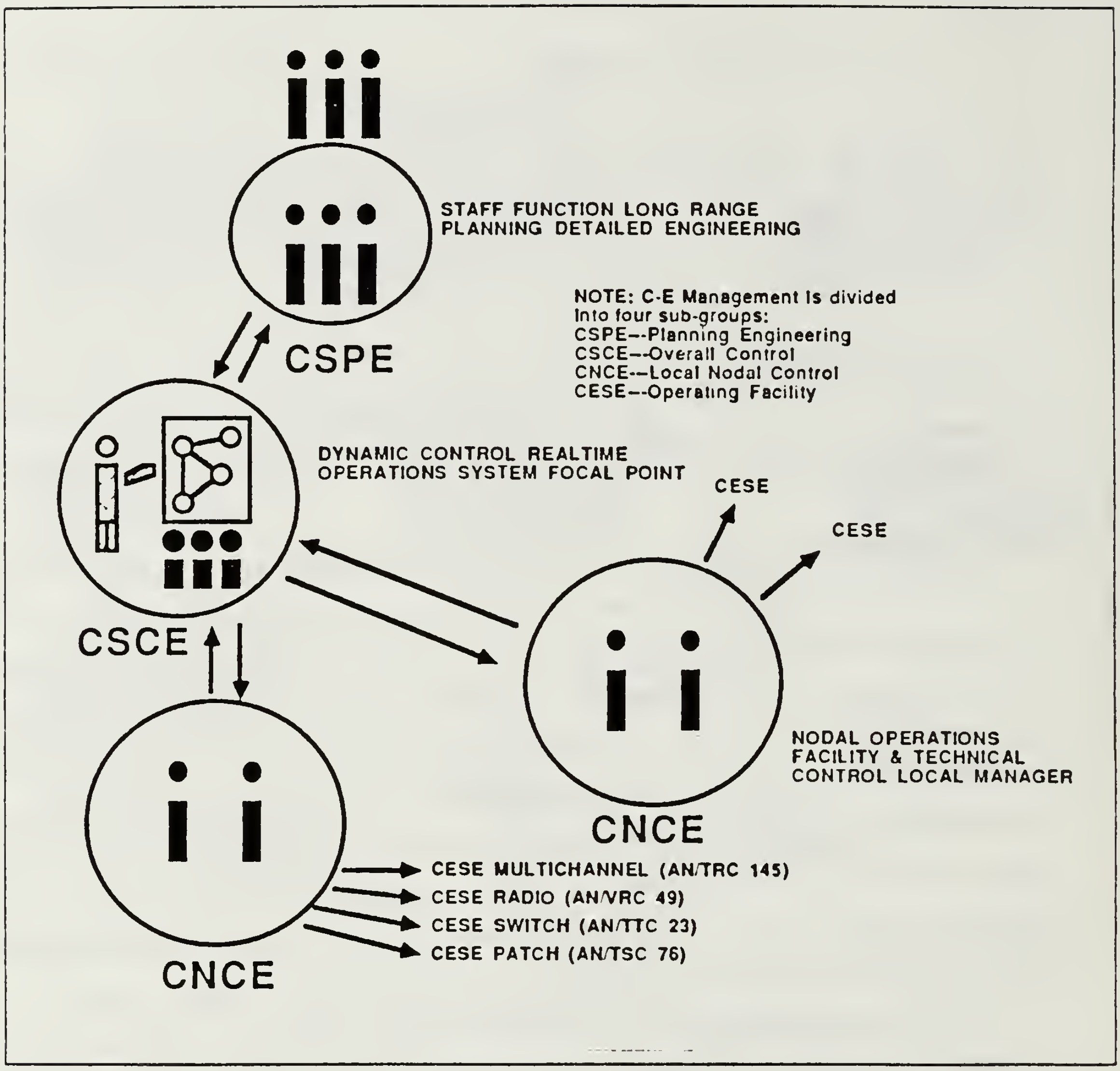


Figure 2.1 Communications-Electronics System Management.

The third echelon is the Communications Nodal Control Element (CNCE) which is the action arm of the CSCE. The CNCE's are located at the various nodes of the communications system and perform the local management and technical control of the node. Functions of the CNCE include monitoring, testing, reporting, patching (routing and rerouting) and local supervision.

The final echelon is that of the Communications Electronics System Element (CESE) which is actual communications assemblage and its operating personnel. The personnel of the CESE are responsible for the installation, operation, and low level maintenance of the communications equipment. They are also responsible for circuit testing and status reporting to the CNCE's. In the figure, the CESE's for the AN/TSC 76 Patching Communications Center (CESE Patch), the AN/TTC 23 Telephone Central Office (Manual) (CESE Switch), the AN/VRC 49 Radio Wire Integration (CESE Radio), and the AN/TRC 145 Radio Terminal (CESE Multichannel) are indicated. [Ref. 4: p. 18]

B. THE TACTICAL MULTICHANNEL COMMUNICATIONS NETWORK

A typical armor, infantry, or mechanized division has organic to its division base a signal battalion whose mission is to provide communication links between units of the division as well as terminate the lateral and higher-to-lower links. The signal battalion has several assets to provide different and redundant means of communications to the various commanders within the division including radio teletype (RATT), FM radio, AM radio, radio-wire integration (RWI), telecommunications center operations, and tactical multichannel system [Ref. 5]. The most secure and easiest system for the subscriber to use is the multichannel system. CACSMAM is designed to help in managing the multichannel communications network.

1. Multichannel System

The tactical multichannel system involves the use of communications equipment which provides the capability of full duplex operations for 6, 12, or 24 channels using only two frequencies (one transmit, one receive). The equipment used provides secure line-of-sight communications in the VHF or UHF frequency band. Antennas used are highly directional which decreases the probability of enemy intercept. The capability also exists to terminate cable multichannel circuits within the same van as the radio circuits. However, cable systems are seldom used due to the logistics of supporting such circuits, as well as the long time it takes to install and break down such a system in a rapidly moving tactical environment [Ref. 6].

Several modes of communications can be supported by the multichannel system. Teletype, facsimile, TACFIRE, and numerous other data type communications as well as voice can utilize the network. Since the multichannel system is secure and very versatile to the user, it is almost always the communications means of choice for personnel wishing to pass communications traffic. As such, the signal battalion commander generally places his highest priority of installation, operation, and maintenance (IOM) on the multichannel network.

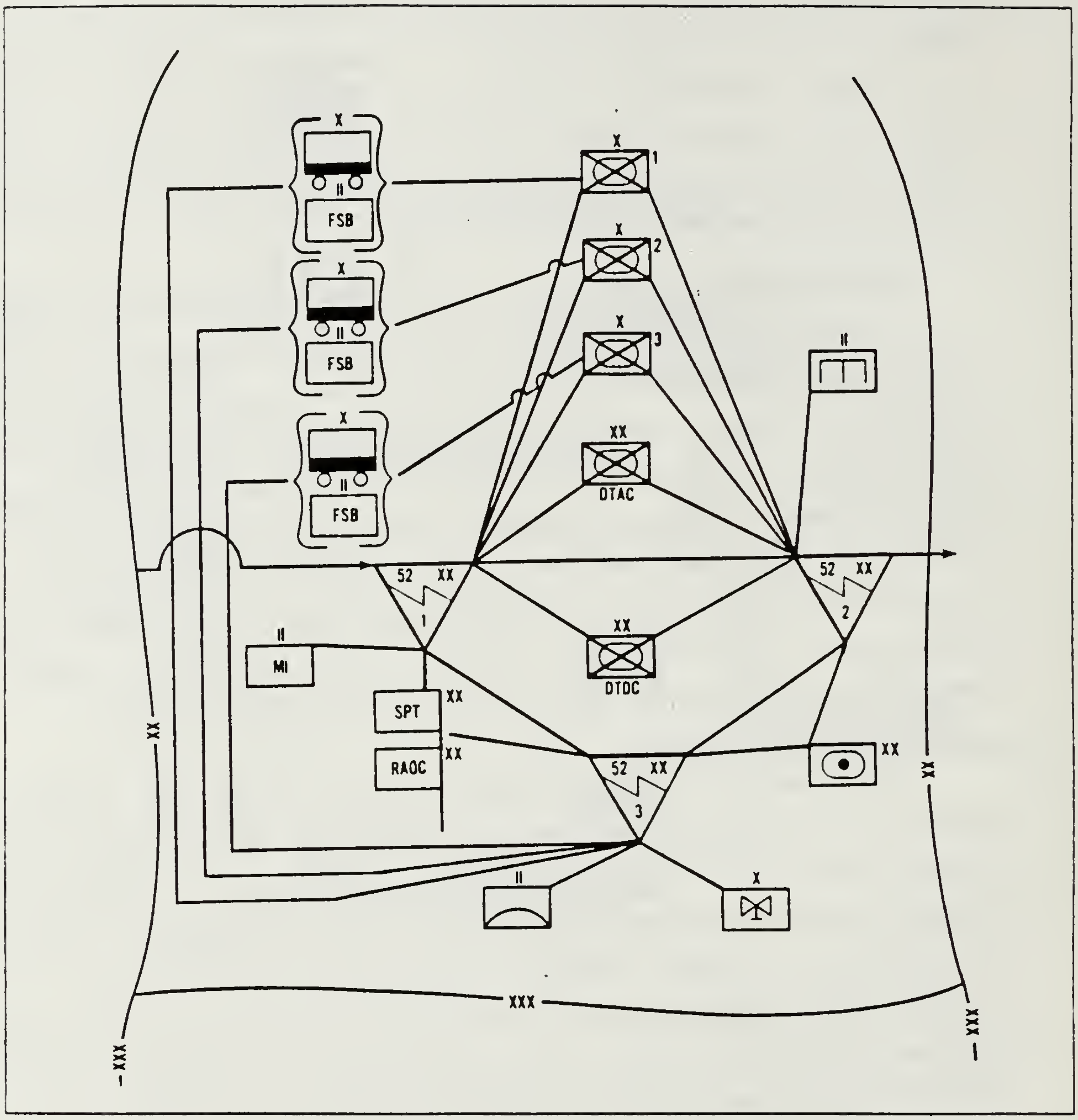


Figure 2.2 Division Multichannel System Diagram.

The doctrinal divisional multichannel network is shown in Figure 2.2 [Ref. 2: p. 116]. Each of the lines connecting the units shown in the figure represents a 12 channel multichannel system. Connectivity is provided between the various nodes by the signal battalion. However, seldom do signal battalion commanders employ their assets as such due to personnel, equipment and tactical considerations, as well as the wishes of the division commander. Therefore the signal battalion commander configures the multichannel system to provide the best communications possible given the current situation [Ref. 2: p. 117]. Figure 2.3 is a diagram of the multichannel system used by the 13th Signal Battalion at Fort Hood, Texas in support of the 1st Cavalry Division from 1981 to 1984. The site and system designators were the Standard Operational Procedure for the unit [Ref. 7].

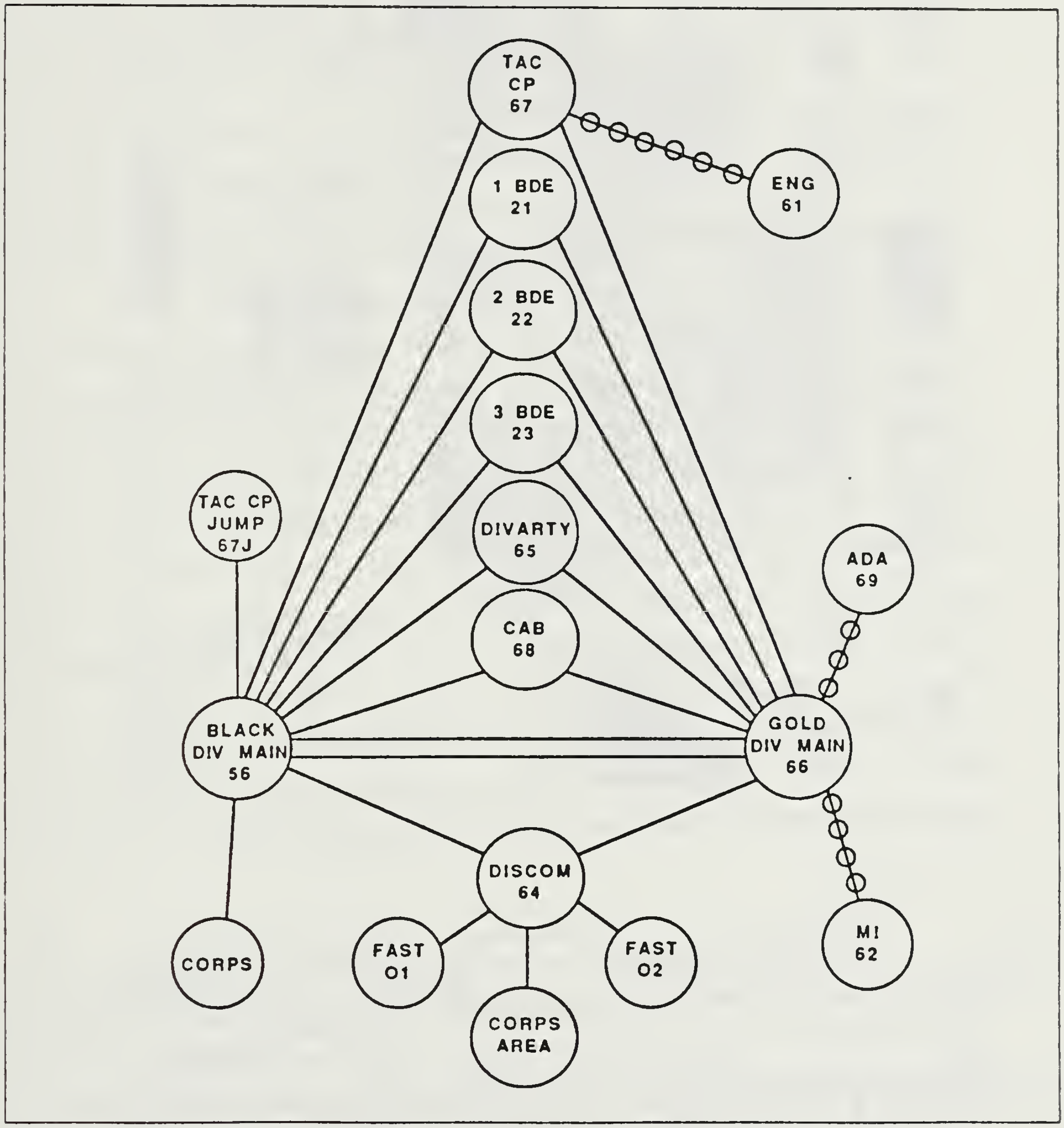


Figure 2.3 Multichannel Network Diagram/13th Signal Battalion.

2. Multichannel Equipment

Within a typical multichannel communications van, several items of equipment are included as components of a functional "stack". The stack consists of all the devices necessary to terminate a radio or cable communications path. A typical stack consists of a receiver, transmitter, multiplexer, signal converter, cable-combiner, antenna, crypto gear, and associated cabling. There are several different generations and configurations of multichannel equipment in the Army inventory. Each has similar devices which perform the necessary functions. Figure 2.4 [Ref. 2: p. 149]. shows the interior of an AN/TRC 145 Radio Terminal, which is the multichannel van used by the 13th Signal Battalion [Ref. 8]. The stack of equipment shown in the figure consists of the RT-773 Receiver-Transmitter (also known as the Order Wire), the R-1329 Receiver and T-983 Transmitter, the TD-754 Cable Combiner, the TS-EC KG-27 Secure Device, the TD-660 Multiplexor, and the CV-1548 Signal Generator.

In order for a multichannel system to remain operational, all the components of the stack must be operational (on both sides of the communications path). The components, and the circuit path through the components are described in Figure 2.5 . The user (up to 12) originates a signal (using a teletype, telephone, etc.) which enters the CV-1548 Signal Generator. The signal is then multiplexed with the other users by the TD-660 Multiplexer and encrypted by the KG-27 Secure Device. The signal then passes through the TD-754 Cable Combiner (if the signal is to be transmitted via cable), or through the T-983 Transmitter (if the signal is to be transmitted via radio). The receive path is the same except it involves the R-1329 Receiver instead of the T-983 Transmitter. The RT-773 Receiver Transmitter sends and receives a signal (which is neither multiplexed nor encrypted) which is designed for the operator of the multichannel vans at connecting nodes to troubleshoot any problems in the circuits between them.

3. Scenario

CACSMAM is designed to assist the Battalion Maintenance Officer in managing the multichannel assets for a division level multichannel communications network. All of the initial information that he/she needs to make a decision on the allocation of those assets is stored in the fact cache (to be explained more fully later) of CACSMAM. The site designators and system designators used in the system are those depicted in Figure 2.3 . The equipment types stored in CACSMAM are those shown in Figure 2.4 as part of the AN/TRC 145. All of the rules and procedures used in

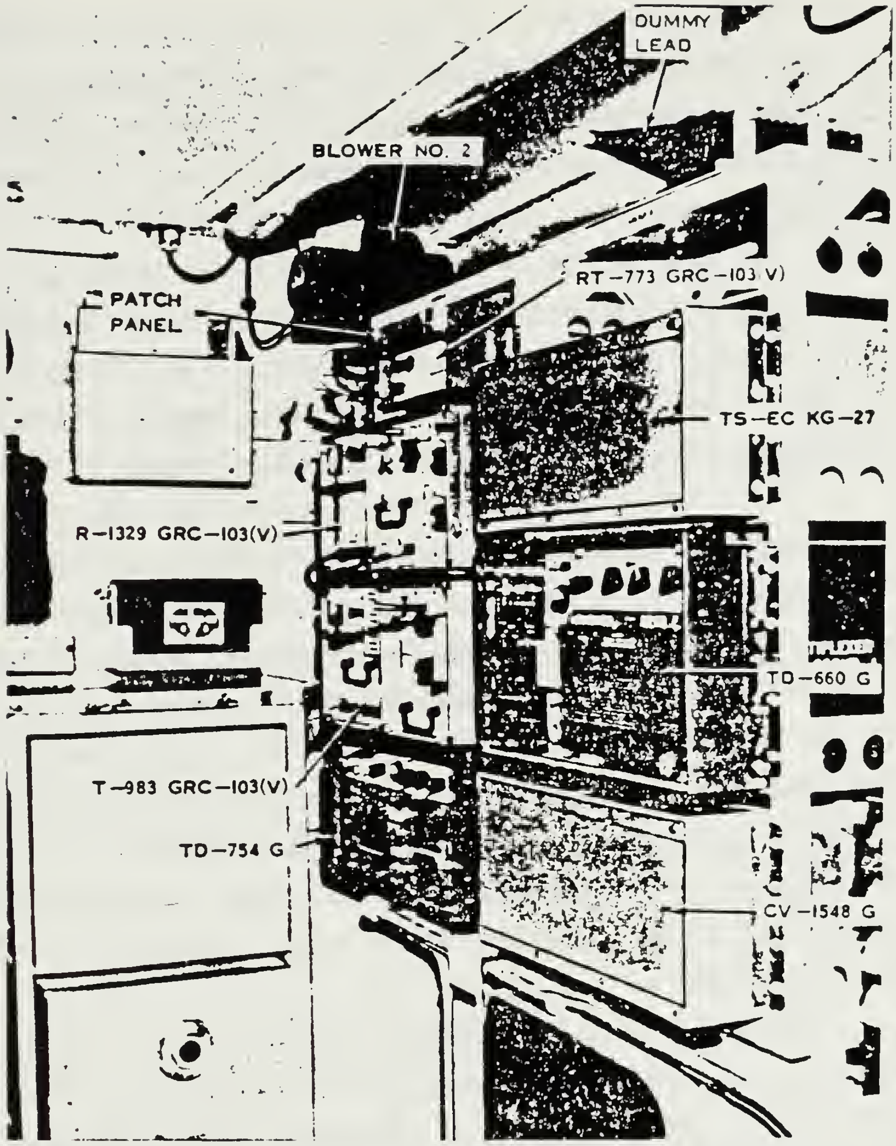


Figure 2.4 AN/TRC 145 Multichannel Equipment.

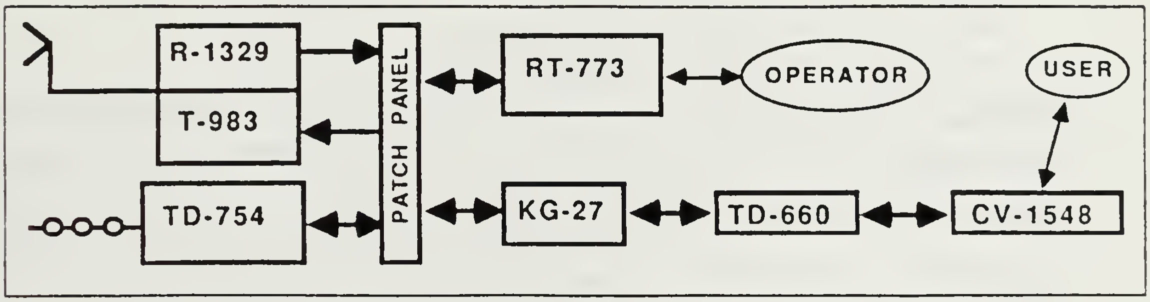


Figure 2.5 Circuit Path Through Components of AN/TRC 145.

CACSMAM are taken from the author's past experience with the 13th Signal Battalion and were subjected to verification, the process of which will be discussed in Chapter III. Prior to determining the rules to be used however, the situation in which CACSMAM would be used had to be defined; in other words, what assumptions were designed into the system.

The simple scenario which was used in the development of CACSMAM is that of a signal battalion deploying to the battlefield, installing a multichannel communications network as depicted in Figure 2.3 , and the BLOC receives a call from one of the multichannel operators at one of the sites informing the Battalion Maintenance Officer that a system is experiencing problems.

While several possible problem situations can exist within a malfunctioning multichannel system, all can be classified under the two general categories of circuit degradation and circuit failure. Circuit degradation (static, fading, loss of signal strength, etc.) can be caused by weather conditions, progressive clogging of equipment filters, and components beginning to fail. Circuit failure can be caused, for example, by total loss of power (loss of a generator), extreme weather, or failure of a component. CACSMAM is designed on the assumption that the circuit has failed, and it is due to a component failure within the AN/TRC 145. It is acknowledged that electronic countermeasures employed by the enemy can cause circuit degradation and failure of our systems. However, electronic warfare effects were not considered to be part of CACSMAM.

Generally, most multichannel operators and supervisors recognize the symptoms of a circuit degradation and begin diagnosing the causes prior to total failure. So prior to total failure, or very soon after, the personnel at the site of the malfunctioning equipment are in communications with the BLOC, through the CSCE, seeking assistance from the Battalion Maintenance Officer.

C. BATTALION MAINTENANCE OFFICER

The CSCE, in fulfilling it's responsibilities for the management of the Tactical Multichannel Communications Network of the division, requires continual input from the BLOC in order to keep current on the status of the readiness of the equipment within the network. The person within the BLOC with the responsibility of maintaining the equipment status within the signal battalion and providing the information to the CSCE is the Battalion Maintenance Officer.

1. Duties

The Battalion Maintenance Officer (BMO) is responsible for the overall equipment maintenance posture of the battalion. The BMO supervises : 1) the direct support maintenance of all organic communications-electronic (C-E) equipment, 2) the organizational maintenance of all wheeled vehicles, power generation, and C-E equipment, and 3) the preparation of all required records and reports in support of the battalion equipment maintenance program. He coordinates closely with the other members of the staff and the company commanders to enhance the battalion maintenance and training program. He is also the primary advisor to the commander on all matters pertaining to the maintenance of C-E equipment, wheeled vehicles, and power generators. [Ref. 9]

The BMO is given the authority to allocate resources (available in the unit) in order to maintain the communications systems. CACSMAM is designed to assist the BMO in the allocation of these resources. The assets which are available to him for consideration in his decision are discussed below.

2. Assets Available

Several options exist when a component of a multichannel rig fails. The two broad categories of action are repair and replacement. Repair of the equipment can be done either by adjustment of some internal circuits or by the replacement of minor parts, or both. Replacement involves finding another servicable like item of equipment, either from stockage or through controlled exchange--"cannibalization". Normally, other like items of equipment are available, but in limited amounts and thus priorities must be established. If two systems are down for the same reason, and only one spare piece of equipment is available, an allocation choice must be made. The sources of multichannel assets for the BMO of are spares, operational readiness floats, backup stacks, and "jump" stacks.

a. Spare Equipment

Spares are those items of equipment which are not considered major components or end items of a communications assemblage. Cables, light bulbs, fuses, filters, circuit cards, microphones and thousands of others are considered in the category of spares. While not considered major components, some spares are nevertheless expensive and in limited supply.

b. Float Equipment

Operational Readiness Float items (ORF) are those items of materiel authorized to be maintained on hand at a maintenance activity for the replacement of like items evacuated for maintenance from the using units. Servicable replacements from ORF assets are provided when like items of equipment from supported activities cannot be repaired or modified in time to meet operational requirements. [Ref. 10]

c. Backup Equipment

In a multichannel network (such as the one in Figure 2.3), not all of the multichannel assets are committed. For each AN/TRC 145 there are two stacks of multichannel gear capable of terminating two multichannel systems. If, for example, the requirement exists to provide 26 systems (52 stacks, one stack at each end of the system), and 31 AN/TRC 145's (62 stacks) were available in the signal battalion, then 10 stacks would be unused. These uncommitted stacks are considered "backups".

d. Jump Equipment

For those sites which tactically relocate ("jump") often, such as a forward brigade headquarters, the signal battalion commander may want to commit an entire "backup" multichannel rig as an asset to be used for the tactical displacements. This is often done. In these cases, the "jump" rig is considered a committed asset, even though it is not active and terminating a system. It is no longer considered a "backup".

During tactical relocations, a cell consisting of some operations personnel from the supported unit (a "quartering party") along with the communications personnel with the "jump" rig, moves to the new location and establishes the site. The communications are set up and the old site breaks down. The old "jump" rig is now operational, and the old operational rig becomes the new "jump" rig. This "leap-frogging" of the rigs allows for a more rapid displacement of a headquarters since communications are already operational at the new site.

Now that the reader understands how the CEMS is set up, the role of the BMO in the CEMS, and the assets management decisions which the BMO must make when dealing with system failures, the next step is to understand how the decision process is modeled in an expert system. The next section explains what an expert system is, how it works, and discusses M.1 in particular.

D. EXPERT SYSTEMS

A definition of an expert system is given by Professor Edward Feigenbaum of Stanford University who is one of the leading researcher's in expert systems [Ref. 11: p. 5]:

> . . . an intelligent computer program that uses knowledge and inference procedures to solve problems that are difficult enough to require significant human expertise for their solution. Knowledge necessary to perform at such a level, plus the inference procedures used, can be thought of as a model of the expertise of the best practitioners of the field.
>
> The knowledge of an expert system consists of facts and heuristics. The "facts" constitute a body of information that is widely shared, publicly available, and generally agreed upon by experts in a field. The "heuristics" are mostly private, little-discussed rules of good judgment (rules of plausible reasoning, rules of good guessing) that characterize expert-level decision making in the field. The performance level of an expert system is primarily a function of the size and the quality of a knowledge base it possesses.

Expert systems, in order to be useful and perform at a significant level of expertise must include several items [Ref. 12: p. 81]:

- facts about the domain
- hard and fast rules or procedures
- problem situations and what might be good things to do when you are in them (heuristics)
- global strategies (methods of approaching any problem within the overall domain)
- differential diagnoses (methods of breaking specific large problems into smaller ones to solve)
- possibly theories about the domain itself (how and why the domain is the way it is).

Many of the facts and rules which must be included in the expert system can be obtained from such sources as textbooks, regulations, and technical manuals. Many of the global strategies and theories about the domain can be obtained from doctrinal publications. However, most of the heuristics must be provided by the expert(s) themselves.

Through training and experience, experts gain their abilities to solve problems. Relating this ability to a knowledge engineer (the one who is building the expert system) is often quite painstaking and difficult. Experts often cannot convey the reasoning they use to come up with solutions [Ref. 13: p. 48], which is why the best experts to use in developing a system have the following characteristics [Ref. 14]:

- Introspective---adept at picking out their own reasoning for actions
- Articulate---can communicate their thought processes well

- Willing to help---willing to put forth considerable effort to help develop the system
- Is available---can take the time from their current jobs
- Ego and self-identification not wrapped up in their jobs---not threatened by a machine taking their place
- Honest---not going to sabotage the system with bad information if they feel threatened.

For CACSMAM, the domain expert is the author based on his experience and training. As the programmer and developer of the system, the author was also the knowledge engineer.

> Knowledge engineers acquire knowledge from a human expert and then imbed it in an expert system. They are specialists in getting the information from the expert, prototyping an expert system that contains the knowledge, and then working with the expert to improve the system. [Ref. 11: p. 195]

The domain expert as the knowledge engineer puts a premium on the first two characteristics of introspectiveness and articulateness as there are no third party personnel "looking in on" the knowledge acquisition and system development to spot errors in logic or procedure. This also makes the verification of the knowledge base more important since the verification step is the first time anyone outside of the expert system development process has seen the product.

E. DESCRIPTION OF M.1.

M.1 is a sophisticated knowledge engineering tool, suitable for expert systems (generally about 1000 rules [Ref. 15: p. 1-1]), and is designed to seek a goal defined by the programmer and can be set to present a single solution or all possible solutions. It will accept "UNKNOWN" as an answer, answer questions about its reasoning during a consultation, and will calculate certainty factors for its conclusions. It has sophisticated user interface and windowing capabilities which eases the development of a system. [Ref. 13: p. 113]

M.1 requires an IBM PC, XT, AT, or compatible computer, running PC DOS 2.0 or later with a minimum RAM of 512K bytes and two disk drives. As M.1 is capable of color, a color monitor is highly recommended. [Ref. 13: p. 113] CACSMAM was developed using a series of microcomputers (IBM, COMPAQ, TELEVIDEO, WYSE), all with hard disk drives and color monitors.

M.1 (version 2.0 and later) is written in the C programming language (version 1.0 was implemented in PROLOG) and thus is capable of accessing other programs (such as database and calculating programs) through C language patches. This capability was not investigated during the course of this project. M.1 can, of course, access other M.1 programs. The code which is written by the developer can be written in a standard word processor using a language similar to, but much less complex, than PROLOG, and much more like written English. [Ref. 13: p. 114]

1. Inference Engine

The inference engine will seek values for expressions by methodically considering previously stored conclusions (cached values), relevant knowledge base entries, and information supplied by the user. Previously stored conclusions can be those facts that never change that are resident in the program, or values that have been determined previously during the run of the program. These conclusions are stored in what is known as the cache. Relevant knowledge base entries are the rules and processes in the program which will determine through inferencing, the values for the expression. If values have not been determined by either the search through the cache, or by inferencing, then M.1 asks the user:

What is the value of: Expression?

The reference manual for M.1 gives a succinct example of the order in which a value is sought for an expression [Ref. 15: p. 4-2 - 4-3]:

> As an example, consider the simplest possible knowledge base, consisting of a single knowledge base entry:
>
> **goal = advice**
>
> When you begin a consultation using this knowledge base, the following events take place:
>
> 1. The inference engine identifies the goal expression of the consultation and begins to seek a value for **advice.**
>
> 2. M.1 checks to see if **advice** is an arithmetic expression for which it can simply compute the value. It is not.
>
> 3. M.1 searches the cache for a prior conclusion for **advice.** As no such conclusion yet exists, the search is unsuccessful.
>
> 4. M.1 searches through the knowledge base for an entry that can help conclude a value for **advice.** No such entry exists, so again the search fails.
>
> M.1 asks a question:
>
> **What is the value of: advice?**
>
> to which you may respond:
>
> **> > sell.**

6. The system has found a value for its goal expression. M.1 displays the conclusion, along with its justification, and returns you to the top-level interpreter.

> **advice = sell (100%)**
> **because you said so**
>
> **M.1 >**

The method M.1 uses to seek values for expressions via knowledge base entries is called backward chaining. Backward chaining seeks to satisfy the stated goal by seeking rules in which the THEN portion of the rule matches the goal, then seeks other rules whose THEN portion matches the IF portion of the rule which satisfies the goal [Ref. 13: p. 165]. Again, a very good example is provided in the reference manual [Ref. 15: p. 4-11 - 4-13]:

. . . consider the following simple knowledge base:

```
kb-1: goal = best-color.
kb-2: if main-component = fish
         then best-color = white.
kb-3: if day-of-week = friday
         then main-component = fish.
kb-4: question(day-of-week) =
         'What is the day of the week?'.
kb-5: if best-color = white
         then wine = chablis.
```

When a consultation is run with this knowledge base, the following takes place:

1. M.1 begins seeking the goal expression, **best-color.** After first checking the cache, the inference engine tries to find a knowledge base entry that might conclude a value for **best-color.**

2. Finding **kb-2,** the inference engine then tests the premise of that rule by trying to find a value for **main-component**.

3. After checking the cache and finding no conclusion mentioning **main-component**, the inference engine locates **kb-3** and tries to use it. **kb-3** causes M.1 to seek **day-of-week.**

4. The only knowledge base entry that can help find a value for **day-of-week** is **kb-4,** so you are asked the question:

> **What is the day of the week?**
> **> >**

5. If you answer **friday,** M.1 concludes that **day-of-week** is equal to **friday**, and notes that fact in the cache.

6. This causes **kb-3** to succeed, and M.1 notes that **main-component = fish** in the cache.

7. This causes **kb-2** to succeed, and the inference engine notes that **best-color = white.** Since this is the goal of the consultation, M.1 displays its conclusions and returns you to the top-level interpreter:

> **best-color = white (100%)**
> **because kb-2**
>
> **M.1 >**

Had you answered anything other that **friday**, all the rules would have failed and M.1 would have indicated that it could not find a value for the goal expression:

**best-color was sought, but
no value was concluded.**

Note that M.1 does not invoke **kb-5** even though logically it could use **kb-5** to infer that **wine = chablis** after the last conclusion was noted. It does not do so because nothing caused it to seek the value of **wine**. M.1 never invokes a rule unless its conclusion provides a value for the expression currently being sought. An expression is never sought unless it is explicitly declared to be a goal or is sought as a result of backward chaining from a goal.

A limited forward chaining capability is available in M.1 also. Forward chaining seeks to identify all the rules whose IF portions are true, then uses the THEN portions of the rules to find other rules which are also true [Ref. 13: p. 166]. In M.1, the command **whenfound** or **whencached** in the form

whenfound (EXPRESSION = VALUE) = LIST

will cause the LIST of expression to be solved for when the EXPRESSION is found to have the VALUE specified. The **whenfound** command is read "if EXPRESSION = VALUE, then LIST is true". [Ref. 15: p. 7-10 - 7-11]

2. **Uncertainty**

Uncertainty is involved with any decision made by the BMO. This is not only due to his inability to be ***absolutely certain*** that his action is the correct one, but also due to the inexactness of any knowledge upon which he may base his decision. The method M.1 represents this uncertainty is through the use of certainty factors [Ref. 15: p. 4-15 - 4-16].

Certainty factors indicate the degree to which a fact is believed as indicated by an integer from -100 to +100 where

- +100 represents complete certainty
- 20 represents a minimum threshold of belief
- 0 represents no evidence for or against
- negative numbers indicate belief that the fact is false
- -100 represents complete certainty that the fact is false.

Within M.1 certainty factors less than 100 (the default value) may arise because:

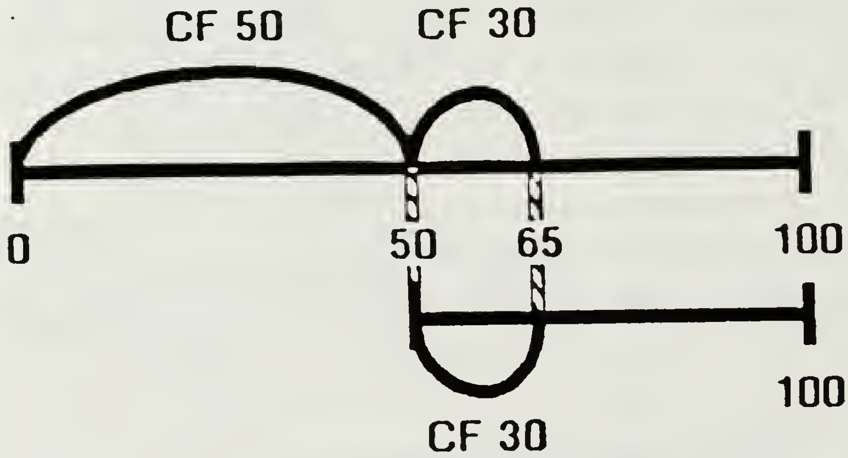
- the answer to a question is qualified by a certainty factor, or
- a resident fact in the cache has an attached certainty factor, or
- the conclusion of a rule in the knowledge base contains a certainty factor [Ref. 15: p. 4-16].

As evidence accumulates during a consultation, certainty factors must be combined to come up with a single level of confidence for the final conclusion. In combining two positive certainty factors, the formula used is:

CF-noted = CF1 + (CF2)% of (100 - CF1).

An example is shown in Figure 2.6 [Ref. 15: p. 4-17]. Certainty Factor 1 (CF1) = 50 and Certainty Factor 2 (CF2) = 30. So the Certainty of the conclusion (CF-noted) = 65 or;

$$65 = 50 + (.30) * (100 - 50).$$



kb-1: if main-component = meat
then best-color = red cf 50.
kb-2: if preferred-color = red
then best-color = red cf 30.
kb-3: main-component = meat.
kb-4: preferred-color = red.
kb-5: goal = best-color.

best-color = red cf 65
because kb-1 and kb-2.

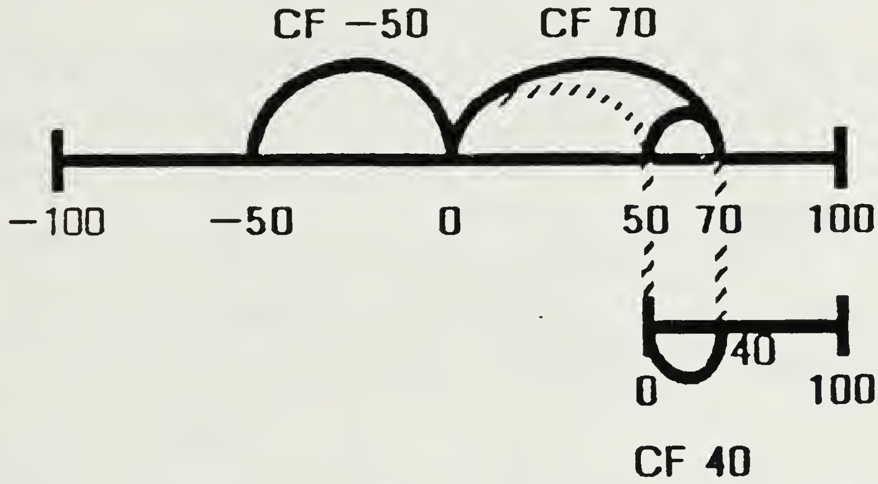
Figure 2.6 Combining Two Positive Certainty Factors.

The combination of two pieces of negative evidence is the same as that for two pieces of positive evidence, with the exception that after the calculation, the negative is taken of the result. The formula is thus:

$$\text{CF-noted} = -(\,|CF1| + |CF2|\,\%\ \text{of}\ (100 - |CF1|))$$
$$= CF1 + CF2\%\ \text{of}\ (100 + CF1).$$

For example, for a Certainty Factor 1 (CF1) = -50 and a Certainty Factor 2 (CF2) = -30, the Certainty Factor concluded (CF-noted) = -65.

$$-65 = -50 + (-30) * (100 + (-50))$$



kb-1: if main-component = fish
 then best-color = red cf -50.
kb-2: if sauce = tomato
 then best-color = red cf 70.
kb-3: main-component = fish.
kb-4: sauce = tomato.
kb-5: goal = best-color.

best-color = red cf 40
 because kb-1 and kb-2.

Figure 2.7 Combining Positive and Negative Certainty Factors.

To combine both positive and negative evidence, the two certainty factors are added, then the result multiplied by a scaling factor of 100/(100 - A) where A is the smaller of the absolute values of the two factors. The formula:

$$\text{CF-noted} = (CF1 + CF2) * 100/(100 - A),$$
$$A = \min(|CF1|, |CF2|).$$

An example is shown in Figure 2.7 [Ref. 15: p. 4-18]. Certainty Factor 1 (CF1) = -50 and Certainty Factor 2 (CF2) = 70. A = min(|-50|, |70|) = 50. So the certainty of the conclusion (CF-noted) = 40 or;

40 = (-50 + 70) * (100/(100 - 50)).

The method used to calculate the combination of certainty factors leads to some noteworthy consequences [Ref. 15: p. 4-19]:

- The final certainty factor is independent of the order in which evidence is found.
- accumulating positive or negative evidence will never lead to a conclusion that has a certainty beyond 100 or -100.
- Once a certainty factor for a conclusion reaches 100 or -100 (stored in the cache) it cannot be lowered or changed by additional evidence.
- Any certainties below 100 or above -100 cannot combine to produce certainties of 100 or -100 respectively.
- A certainty factor of 0 does not change the certainty of a fact.
- With the exception of 100 and -100, equal positive and negative evidence will cancel each other out.

This chapter has covered the domain of CACSMAM and the asset management problem addressed by the BMO in the maintenance of the Tactical Multichannel Communications Network. It also provided the reader with insight into expert systems and M.1 in particular. In the next chapter, the author discusses the strategy used in modeling the BMO's decision process (within the identified domain) as a knowledge system using M.1.

III. DEVELOPMENT STRATEGY

The author used the strategy for building small expert systems as outlined by Harmon and King:

1. Select a tool and implicitly commit yourself to a particular consultation paradigm.
2. Identify a problem and then analyze the knowledge to be included in the system.
3. Design the system. Initially this involves describing the system on paper. It typically involves making flow diagrams and matrices and drafting a few rules.
4. Develop a prototype of the system using the tool. This involves actually creating the knowledge base and testing it by running a number of consultations.
5. Expand, test, and revise the system until it does what you want it to do.
6. Maintain and update the system as needed [Ref. 11: p. 178].

Due to the time and asset limitations on this project, the majority of effort was directed toward the steps two through four, as explained in the following sections.

A. SELECTING A TOOL

This expert system development is sponsored by the Directorate of Combat Development (DCD), United States Army Signal Center at Fort Gordon, Georgia. The DCD directed the author to use the knowledge system development tool marketed by Teknowledge, Inc. as M.1. M.1 was chosen initially to develop prototype systems for the U. S. Army and the Signal Center due to its ease of use for inexperienced programmers, as well as the maintainability of a system implemented using M.1.

The word processing package used by the author to produce the text of the rule base was SIDEKICK by Borland International, Inc. While any IBM PC compatible word processor can be used, SIDEKICK allowed the author to alternate between the running M.1 program and the text editor in SIDEKICK, to make changes as needed. This facilitated rapid changes to the rule base.

The method used by the BMO to gather information about a failed node is by asking questions of the operator or person in charge of the failed node. The answers obtained, combined with the BMO's knowledge of the status of the rest of the network, the availability of supplies and maintenance teams, and his own past experience in similar situations (his "expertise"), lead to his decision of the appropriate course of action. This method of consultation and reasoning leads to an ideal implementation of the question and answer format for an automated system such as M.1.

B. PROBLEM IDENTIFICATION AND KNOWLEDGE ANALYSIS

In identifying the problem, the author used a three step approach:

1. Define an area for which an automated aid might be useful---in this case the Battalion Maintenance Officer's asset management process.
2. Test the problem area to see if an expert system is indeed feasible, or even the right approach.
3. Define the scope and bounds of the problem to narrow the domain of consideration for the system---define the knowledge base.

The author's past experience in tactical communications (signal platoon leader, company executive officer, battalion S4, and company commander) has shown that the Communications System Control Element with its Battalion Logistics Operation Center operates at a very high pace throughout a field exercise. There is nothing to indicate that during an actual conflict the pace would be any less. The personnel making key decisions in those staff elements are few in number and often fatigued. The author felt in conjunction with the sponsor of the project, the DCD, that investigation into an automated aid for the BMO was warranted.

The author then had to insure that the problem was appropriate for an expert system. The checklist used was that outlined by Williamson as a list of characterisitics for a "good" problem for an expert system [Ref. 13: p. 47]:

1. A relevant body of knowledge exists and is available.
2. The skill involved is one which could be taught to a new employee.
3. The knowledge can be expressed in bite-sized pieces that make sense standing alone.
4. Applying the knowledge does not require common sense.
5. Solving the problem without a computer takes someone who is good at it less than a few minutes and no more than a few hours.
6. The benefit that will come from developing the system is sufficient to justify the cost involved.

For the purposes of this thesis, the author fulfills the first characteristic as the domain expert. The managerial skills needed by the BMO are taught by peers, senior non-commisioned officers and experience and therefore the second criteria is fulfilled. Upon initial investigation, the author did not envision any situation which was so complicated as to not be able to be broken down into small enough pieces for a small rule-based system. The knowledge application is straight forward enough and easily understood, thus inconsistencies within the domain can be solved within the rule base; "common sense" interpretations by the user would not be necessary. Normally, reestablishment of a failed communications node is a critical, high priority task and thus must be acted upon immediately; the decision by the BMO is made quickly. Again, for the purposes of this thesis, and in the opinion of the author and the DCD, the knowledge and experience gained from the development of CACSMAM justified its development.

When a node of a multichannel communications network fails, experience has shown the author that there are three major actions which must immediately happen. First, those channels which passed through the node must be rerouted through other nodes; second, the equipment which failed must be fixed and; third, the node must be brought back on-line and the original network re-established as soon as possible.

An automated system, known as the Communications Planning Assistant (COMPASS) is currently being designed to reroute circuits and manage the frequency changes involved. [Ref. 16] Another expert system, known as TAGERS, is currently available (as a training aid) to diagnose downed multichannel equipment. [Ref. 17] CACSMAM is envisioned to fulfill the need for determining how to reestablish the node.

Determination of the knowledge which must be included in CACSMAM was done by the author. Drawing on past experience, the author reviewed different problems in the multichannel network which were solved by resource allocation, and analyzed each to determine the necessary bits of information needed by the BMO to make a decision. The knowledge which must be included in the system consists of several items:

1. The site designators and system designators must be resident in the program.
2. The assets available to replace a failed component must be resident.
3. The location and identification of the failed piece of equipment must be resident.
4. The system that the is down due to the failed equipment must be known.
5. All the possible corrective actions which can be taken must be known (the goal of the BMO).
6. The conditions which must be evaluated to determine a course of action must be resident.
7. The combinations of conditions leading to the possible actions must be resident (the rules).

C. DESIGNING THE SYSTEM

The first step in designing the system was to identify the goal of the expert system. What decision is the BMO going to make? The decision is that corrective action (reallocation of resources) which will reestablish the failed node in the multichannel network. The goal state for CACSMAM was thus defined as the **appropriate-action.** The action(s) defined as the the best allocation(s) of resources to reestablish the multichannel node.

Once the goal state was defined, all of the actions which could be considered as decisive (all of the BMO's options), were written down in a matrix similar to a decision table. The first conditions that came to mind which would lead to the various actions were also written down. It soon became apparent that several conditions overlapped for different actions, and that several actions could actually solve the problem. Since several actions could be taken in response to like sets of conditions, and the decision table is designed to render unique solutions, the decision table is invalidated as an appropriate aid [Ref. 18]. It is however a tool that the author used to organize his approach. Figure 3.1 is an example of what the author used.

To solve for the goal state, all of the conditions leading up to the goal state must be sought. These are the antecedents to the rules for which the appropriate action is the conclusion. Some of the antecedents leading directly to an appropriate action conclusion, are in and of themselves conclusions in rules which have several antecedents, and so on. Thus, a tree is formed for the backward chaining process as discussed in Chapter II. Ultimately, those conclusions which cannot be reached through the inference process must be found through facts obtained either from the database resident to the system or from the user through his responses to questions posed to him by the system.

D. DEVELOPING THE PROTOTYPE

To develop the prototype, the author took one **appropriate-action** (one goal state) and analyzed the sets of conditions which would lead the BMO to take that action. Using these conditions as goals by themselves, criteria to solve these intermediate goals were sought, and so on until only facts were needed, either provided in the cache or by the user.

All of the conditions were written down in the production rule format acceptable to M.1. Once the first **appropriate-action** (such as **repair-system-X-with-spare-PART**)

RULES / ACTIONS	A	B	C	D
repair with spare part	X			X
cannibalize backup rig	X	X	X	X
replace with backup stack	X	X		
CONDITIONS				
spare part is available	X			X
backup stack is operational	X	X		
repairs will be made before jump	X	X	X	X
backup stack is not operational			X	X
part in backupstack is operational	X	X	X	X

Figure 3.1 Example of Decision Table Matrix.

was solved (one branch of the tree), a second **appropriate-action** (such as **replace-system-X-with-backup-stack**) was solved (next branch of the tree), and the process continued for the defined **appropriate-action(s).**

Once the first iteration of rules were entered into M.1, the system was run, and glitches (such as illogical flow of questions or unexpected solutions to a given set of conditions) in the program were noted for future revision.

Once the rules were written, and the system considered finished (before any testing), the knowledge base was checked for redundancy of rules. This was done manually and systematically by checking the conclusions of the rules to see if sets of antecedents were identical for identical conclusions. The impact of redundant rules is not critical, yet it does slow down execution of the program.

The rules were then reconciled. Every antecedent of every rule was checked for solution; in other words, each antecedent had to be solved for by the system in order for conclusions to be reached. This process was also done manually, checking through the rule base. However, the M.1 inference engine helps by asking the user:

What is the value of: (antecedent)?

when the value for the antecedent is searched for but not found in the resident knowledge base. This indicates that no rule had fired, no fact was stored, or no question was asked of the user in order to solve the antecedent. Rules were then made to solve for those antecedents which were missed.

E. EXPANDING, TESTING, AND REVISING THE SYSTEM

1. Expansion

Expansion of the system involves either redefining the system boundaries, thus expanding the domain considered in the system, or by user-friendly additions, or both. Expansion of the boundaries of the domain (such as including system troubleshooting as well as corrective action decisions) involve expanding the breadth and depth of the rule tree. User-friendly additions, while adding rules to the knowledge base, are not necessarily designed to lead directly to goal state conclusions, but rather make the system easier to use or understand.

a. Breadth

Expanding the breadth of the system involves adding more solutions to the goal state of CACSMAM; in other words, more **appropriate-actions** would need to be identified. This could be done be either finding more unique solutions or by splitting the current solutions into more specific or detailed ones with differing sets of criteria for implementation.

Expanding the breadth of the system increases the number of rules dramatically, for each new solution has unique sets of criteria, and thus a new set of branches for the tree. Due to time and asset limitations, the author chose to bound the domain and limit the scope of the solution set to the one currently in CACSMAM.

b. Depth

To expand the depth of the system would involve more rules to lead a more ignorant user to a solution. For instance, if the user did not know the answer to a question, or possibly where to go to find the answer, the system could provide guidance, or a series of questions to lead the user to the proper answer, or both. This involves getting deeper into the tree to more rudimentary facts and procedures, which if known by the user, are bypassed at the higher levels of the tree. Thus, more depth is added to the system.

CACSMAM as currently implemented requires a certain level of communications expertise by the user in the form of some knowledge of a multichannel system and its terminology. This level was chosen on purpose in order to limit the depth of the system for this thesis. However, expansion of the system's depth is certainly worthwhile, especially if CACSMAM is to be used in a classroom environment for students who may not have the required level of expertise.

2. Testing

The testing of an expert system is difficult at best. Since one of the key attributes in choosing an expert system for a particular application is the ease in which the system can be changed (in order to remain "expert"), by definition the system is dynamic. Traditionally, expert systems have been tested by verification and validation, neither of which use traditional statistical methods primarily because if the system does not perform as well as an expert, the rules are changed until it does.

a. Verification

Verification consists of two parts: verification of the inference engine and verification of the rule base. Verification of the M.1 inference engine is beyond the scope of this project, therefore the assumption is that the engine is correct and verified.

The verification of the rule base involves allowing outside experts to look at the knowledge base and comment on its accuracy. This outside expertise can spot factual, procedural or logical errors as well as point out different methods of reaching solutions, or possibly even different solutions.

For the verification of CACSMAM, the author gave hard copies of the knowledge base to three officers who have had division level signal experience and are considered knowledgable in the management of AN/TRC 145 mulitchannel assets. Captain Donald Howard, of the Directorate of Combat Development, Captain John Schoedel, of the Office of the Chief of Signal, and Captain Paul Rossbach, instructor in the Senior Leadership Department of the USA Signal School, were asked to review each rule individually as a single logical entity, and comment on its accuracy of logic, procedure and fact. Over a one week period each submitted comments to the author who proceeded to make changes to the knowledge base until the three had no further comments on the accuracy of the rules. CACSMAM was then briefly demonstrated to the USA Signal Precommand Course students, all of whom were attending the course in preparation for taking commands of signal battalions or brigades (lieutenant colonel and colonel rank). CACSMAM provided proper solutions to given problems.

At this point the knowledge base was considered verified due to the rules being verified and the program itself being able to run properly and initially giving proper solutions. However, CACSMAM cannot be considered validated as discussed in the next section.

b. Validation

One method of validating the system is done by using case studies of known problems and their solutions and posing the problem to the system. If the system comes up with the same answers as the experts who solved the problem in the case study, then the system can be considered validated.

A second approach is called a "Turing Test". This involves using a set of identical problems posed to both the experts and to CACSMAM, without either knowing the proper solution. Each solves the problems and the solutions are compared side by side by an impartial panel of experts who are unaware of which system came up with which solution. If the panel cannot tell which solutions were provided by the experts and which were provided by CACSMAM, then the system is indeed an "expert" and thus passes the "Turing Test".

c. Other Tests

A different test of an expert system is proposed by Rook and Donnell. They tested

> . . . user/expert system interaction as a function of two variables, control strategies and user's conceptual model, by investigating: (1) the degree to which similarity between an expert system's and its user's control strategies (i.e., forward or backward chaining) affects the expert system's utility; and (2) the extent to which a user's accurate conceptual model of the expert system's inference mechanisms influences combined user/system performance [Ref. 19].

What Rook and Donnell tested is the effectiveness of the user/system interaction; 1) when the user's control strategy was changed from forward to backward chaining, either matching the strategy used by the system or not and, 2) when the user had an accurate conceptual model of the expert system and the interface between the user and system, or not. They found that an accurate understanding (conceptual model) of the system facilitated the user/expert system interaction and was more significant to performance of the user/system team than was the matching of control strategies between the user and the expert system. The results were interesting and the reader is encouraged to read their paper. [Ref. 19]

3. Revision

Several revisions could be made to CACSMAM, all of which fall into the categories of expansion (breadth or depth), or user friendly additions. The capabilities of M.1 (which will not be discussed here) allow for a very pleasant work environment. User friendly additions would actually be expanding the depth of the rule base by giving more detailed explanations, or allowing the user to investigate the possible ramifications of a particular course of action. The addition of a database management tool which would keep account of assets would be an outstanding revision to CACSMAM.

F. MAINTENANCE AND UPDATING OF THE SYSTEM

The maintenance of the system is quite easy which is one of the primary attractions of a rule-based expert system. When a situation occurs for which CACSMAM does not provide a recommendation, or gives an unacceptable one, the user can call upon the resident expert in the area to look at the system. CACSMAM has the capability of showing its reasoning which allows the expert to find the rule which is at fault, or to find where an ommision in the rules did not account for a particular situation. The rule base can then be entered and modifications to the rules made.

Updating of the system can be done the same way, yet may be more involved. Major revisions might have to be made to the system if major changes in maintenance policy required changes in the terms used in CACSMAM, or if echelons of maintenance changed, or some other unforeseen change occurs in the way business is done. However, even major revisions are relatively simple (yet quite possibly tedious), since all the rules are written in a language very similar to English and changes are easily made from rule to rule.

G. SUMMARY

The steps outlined in this chapter are by no means the only ones that can be used to develop a system. However, the author feels reasonably confident that CACSMAM is a fairly "complete" system (given the limitations of the domain), but probably more importantly understands where the boundaries of the domain lie well enough to expand the system if warranted. The development strategy used ensured that all of the steps taken left an audit trail for further development by the author. In the next chapter, the author offers some conclusions reached as a result of the development of CACSMAM and offers some recommendations for further research.

IV. CONCLUSIONS AND RECOMMENDATIONS

CACSMAM does what it was designed to do by providing courses of action for the BMO to take when a multichannel system fails, limited only to the asset allocation. CACSMAM was narrowly bounded and its application limited to only multichannel equipment. However, several things could be done to improve CACSMAM as it is currently, without changing its scope. Expansion of the scope of the system to access databases and tie in to other systems which aid in the management of the communications around the battlefield represent additional possible improvements. This chapter will detail conclusions and recommendations concerning CACSMAM as it stands, its possible future, expert systems in general, and recommendations for future investigations.

A. SPECIFICALLY ABOUT CACSMAM AND M.1 (VERSION 2.0)

The following suggested improvements concerning CACSMAM are possible using the current capabilities of M.1, and are well within the capabilities of the author. However, due primarily to time limitations the author did not implement them.

1. Storage Requirements and Memory Usage

One of the unstated goals of the author was to keep CACSMAM limited to the storage capacity of one 5 1/4" floppy diskette (to insure physical portability). The storage capacity available proved to be sufficient for this system, but more efficient use of the space is possible and would allow for a more comprehensive system. By using storage space more efficiently, expansion of CACSMAM is possible within the one diskette limitation. However, it must be noted that an expanded system, even though it may fit on one floppy diskette, would more than likely involve more variable manipulation and thus require additional RAM.

CACSMAM currently uses approximately 338K bytes of disk space. Since it is configured as an executable file, it is loaded completely into RAM, and with the variable storage space required by the program, CACSMAM takes up virtually all of the available RAM on a 512K byte RAM capacity microcomputer. This is obviously a drawback if no more RAM is available and if CACSMAM is to be expanded with more capabilities. Three methods of reducing the memory requirements of CACSMAM are possible; use of forward chaining, better variable manipulation, and modularized programming.

a. Use of Forward Chaining

The author wrote nearly all the rules for a backward chaining control strategy. Backward chaining is the default M.1 control strategy, tending to "eat up" more memory than necessary by seeking values of antecedents which may not be relevant to the final solution. Since M.1 has a forward chaining capability, judicious use of that capability would reduce the memory needed, as well as speed up the program (by asking fewer questions).

b. Use of Advanced Variable Manipulation of M.1

M.1 has commands and some advanced syntax available allowing for a more condensed style of coding and thus lessoning memory usage. These methods enable the program to pass variable values more efficiently between rules and the cache. Currently, CACSMAM uses one antecedent to instantiate one variable in a large percentage of the rules. By consolidating these several antecedents with several variables into one antecedent using list structure syntax and fact tables, less memory space is used [Ref. 14: p. 10-23].

c. Modularized Programming

By breaking down CACSMAM into modules, with only the driver module being the executable file, the amount of RAM capacity used up would be significantly less. The driver module would be the only one resident in RAM, with the other M.1 modules called from the diskette as needed. Example modules could be: the driver module, the user preparation module (checks for the system and site designations, location of maintenance personnel, etc.---all the administrative overhead data), the asset availability module (checks for availability of spares, backup stacks, etc.---this would be the one to most likely access a data base), and a decision module.

While modularized programming would use less RAM, it would use up more storage space (due to overhead rules and whatever patches may be necessary). Just to modularize CACSMAM as it currently is written would have caused the program to be too large to fit on one floppy diskette. Modularization of the program would probably tend to slow down the program a bit also. But the author hypothesizes that the change would not be significant because the modules would only be called as necessary---they would not be used in every consultation necessarily. Currently, M.1 passes through the entire CACSMAM rule base several times during a consultation to find values. Modularizing the rule base would eliminate several passes through the same rules. However, time saved by not passing through the rules is offset by overhead rules and the physical accessing of the secondary storage device to find the needed module for consultation. Hence, the speed of operation probably would not significantly change.

2. Ease of Use

a. Readability

CACSMAM currently asks simple and fairly straightforward questions of the user. However, improvements can continually be made to the wording of the questions and the visual presentation of information to the user. In the instance of the extremely weary user, a complicated display of screens, or unclear questions can make the system hard to use or tolerate. The next obvoius style of visual presentation is graphics.

Currently M.1 does not have a graphics capability, although it can access graphics software packages. This of course requires more memory and storage capability. Connecting some sort of graphics display (to depict the multichannel network) with a visual display of where all the assets are located within the network, and enabling the user to use a mouse or light pen to point out where the problems and assets are, and then letting CACSMAM reach a decision, would be a great stride in simplicity for the BMO. The decision made would then be made based on all the information available (stored in databases, in the cache, etc.) rather than relying solely on the memory of the BMO as to where all the assets may be located. As stated in Chapter I, the complexity of the environment is very high, and the possibility of the BMO not considering all the data available can lead to less than optimum decisions.

b. Explanation of CACSMAM Reasoning

One of the biggest advantages to an expert system such as CACSMAM is the inherent ability to provide reasoning for the solution presented, an audit trail. However, the usefulness of the reasoning process used by the system is only as good as the medium in which it is presented to the user. There are a few commands within M.1 which allow the user to see the reasoning process of the running program. The two most useful and significant are the **why** and **show** commands.

The **why** command causes M.1 to display the reason why a particular question is being asked or why a particular value is being sought. The way M.1 answers the **why** query is shown in the following example (taken from the sample consultation in Appendix B):

```
Even though the whole backup stack is not operational is the 11A23 panel
out of the backup stack operational?
```

```
        1. yes
        2. no
        3. glossary
>>why
        M.1 is trying to determine whether the following rule is
        applicable in this consultation:

        kb-124:
              if component = '11A23 panel' and
                    site-designator = '56' and
                    backup-stack is on-site- '56' and
                    backup-stack is not-operational and
                    not('11A23 panel'-in-backup-stack is operational)
              then '11A23 panel'-in-backup-stack is not-operational.

        The following entries are also under consideration:

              kb-125(a rule)
              kb-65(a rule)
              kb-4(a goal)
```

As the reader can see, M.1 just provides a copy of the rule being fired. as it was written by the programmer, along with the number of any other rules under consideration. Dependent on the programmer's style, and the particular rule being fired, the response to a **why** query can be fairly cryptic to the user.

The **show** command merely displays to the user all the values stored in the cache up to that point in the consultation which provides the audit trail for the user. Again, as seen in the example below (also from Appendix B), depending on what the programmer decides to call things, the results can be confusing:

```
        appropriate-action = cannibalize-11A23 panel-from-backup-stack (72%)
        because kb-70.
CACSMAM->>show
        system-designator = 5666PBA (100%) because you said so.
        site-designator = 56 (100%) because you said so.
        system-site-check-ok = yes (100%) because kb-42.
        system-site-test = yes (100%) because kb-40.
        item = 11A23 panel (100%) because you said so.
        component = 11A23 panel (100%) because set by user.
        minor-part is identified = yes (100%) because kb-52.
        maintenance-team is required = no (100%) because you said so.
        appropriate-action = cannibalize-11A23 panel-from-backup-stack (72%)
        because kb-70.
        spare-11A23 panel is on-site-56 = yes (100%) because you said so.
        spare-11A23 panel is available = yes (100%) because kb-99.
        backup-stack is on-site-56 = yes (100%) because you said so.
        backup-stack is operational = no (100%) because you said so.
        opposite(no) = yes (100%) because kb-27.
        backup-stack is not-operational = yes (100%) because kb-128.
        11A23 panel-in-backup-stack is not-operational was sought, but no
        value was concluded.
        11A23 panel-in-backup-stack is operational = yes (100%) because you
        said so.
        opposite(yes) = no (100%) because kb-26.
        backup-stack is not-on-site-56 = no (100%) because kb-127.
        11A23 panel-in-backup-stack is available = yes (100%) because kb-123.
        11A23 panel-in-backup-stack is not-available = no (100%) because
        kb-130.
        time-to-repair-system-5666PBA-on-site-56-with-spare-11A23 panel =
```

```
    1.0 (85%) because kb-106.
    maintenance-team is not-required = yes (100%) because kb-60.
    repair-time-for-system-5666PBA-on-site-56 = 1.0 (85%) because you
    said so.
    time-until-repairs-can-begin was sought, but no value was concluded.
    spare-11A23 panel is not-on-site-56 = no (100%) because kb-141.
    time-to-cannibalize-a-stack = 0.7 (80%) because you said so.
    a-tactical-jump-of-site-56 is planned = no (100%) because you said
    so.
    a-tactical-jump-of-site-56 is not-planned = yes (100%) because
    kb-140.
    replacement-can-be-made-before-jump was sought, but no value was
    concluded.
    spare-11A23 panel is not-available was sought, but no value was
    concluded.
    backup-stack is not-available was sought, but no value was concluded.
CACSMAM-> >
```

In both instances of the **why** and **show** commands, M.1 provides the capability to tailor the output to the user in plain text. This capability is valuable because it takes full advantage of the expert system's capability to display its reasoning. The author did not pursue the tailoring of the screens for the commands for it involves several more rules and hundreds of lines of text explaining each rule (in response to the **why** queries). Time and the self-imposed limit to the capacity of one floppy diskette were the main limitations to the author.

3. Expansion of the System

a. Use of Unknown

M.1, and therefore CACSMAM, has the capability to accept "unknown" as an answer. The inferencing process will continue even though a value has not been concluded. This capability is valuable because while the system may not be able to continue reasoning down a particular branch of the knowledge base, it can shift to another line of reasoning and continue to seek a conclusion, just as an expert would.

To exploit this capability, the system could be expanded to include some forward chaining rules to lead the user (who doesn't know the information needed for a particular chain of reasoning to continue) to the information needed. The system could provide the user information about where and when the information might be, or give the user clues on how to derive the information needed. Thus, the system may ultimately gain the information needed from the user, but if not, at least the user would know more about how to get necessary information.

b. Access to Databases

Modularizing the program, and including databases within the modules could significantly ease the record keeping burden of the BMO. C language patches could be written to access databases which include site designators, system designators, and the equipment, maintenance and supply assets on each site. Each time a decision was made, and assets reallocated, the database could be updated by CACSMAM, thus eliminating a repetitive chain of questioning. By not repeating the same questions in every consultation, a significant reduction in time in consultation with CACSMAM could be realized.

The author feels that accessing databases with CACSMAM would be the most significant improvement which could be made. The majority of time spent in consultation with CACSMAM is establishing which assets are available and where they are. Currently the BMO uses some sort of manual accounting system and his memory to keep track of assets. The database module for CACSMAM would eliminate the need for this manual accounting, lesson the RAM requirement for CACSMAM, lesson the time required for consultation, and thus increase the efficiency of the system overall.

c. *Inclusion of Other Network Equipment*

The domain of CACSMAM could be expanded to include communications equipment of other types which must be managed by the BMO. Just about any major piece of communications equipment consists of components which may be replaced at the battalion level (under the current maintenance concept). Therefore, the skeleton of CACSMAM could be used to develop other systems which may be used for radio teletypewriters, telecommunications centers, switchboards, etc. Later they could be combined with CACSMAM to form a comprehensive expert maintenance asset control system for the BMO.

B. MICROCOMPUTER BASED EXPERT SYSTEMS

1. General

One of the major benefits of M.1 and other microcomputer based expert systems is the ease with which people who have little or no computer programming experience learn to use them to develop *useful systems.* The instructors at the US Army Signal School at Ft. Gordon, Georgia have found (and subsequently designed a two week training course on the premise) that personnel with expertise in particular areas, but none in programming, can learn to use M.1 and develop a simple working system within 2 weeks. A comprehensive, usable system can be implemented within months. With this in mind, several systems can be developed using M.1 (or tools like it) for the military.

Maintenance procedures for troubleshooting and repair are prime candidates for expert systems. Administrative and logistical procedures could be set into rule-based format fairly readily for implementation by an expert system. Control procedures which are unique to a particular network of management could be set into a expert system. Using the checks outlined in Chapter III, expert systems can be developed for just about every military workspace to relieve some of the personnel workload.

2. Expert System as Trainers

A significant amount of effort is spent by experienced personnel training others who are new or less experienced in an organization (especially in the military with frequent personnel moves). Standard Operating Procedures (SOP) of the organization, as well as simple knowledge needed to perform at an acceptable level could be provided to the new member in the form of a expert system to be consulted on his/her desktop as needed. This would free up the other personnel to take care of other business.

Not only would an expert system relieve workload, but could also provide a means to preserve the corporate memory of the organization. Every organization dreads the loss of prized knowledgeable employees, especially in the military where tours of duty tend to be short and required training time tends to be getting longer. Expert systems can capture the knowledge of resident expert employees, and thus "replace" them when they leave the organization. Since expert systems are also very easily maintained, as new experts develop with new procedures, the expert systems can be updated and the knowledge base expanded continually. Thus, eventually it can be foreseen that the expert system, with the combined knowledge of several experts contained within its knowledge base (derived over several iterations), could become as good an expert in its domain than any individual human expert.

3. Fielding of Expert Systems

With all the capabilities of microcomputer based expert systems, their ease of development and the benefits which can be realized, fielding of these systems must be made as rapid as possible. A rapid prototyping and fielding system should be implemented to get expert systems to the field as soon as possible after development where the benefits can be realized.

Since rule based expert systems such as CACSMAM are so easy to change and modify, the traditional high cost of software maintenance is drastically reduced. Now the user (or at least the resident domain expert) can modify the software without being a computer programmer. This is very valuable since now a computer sofware product can be sent to the field without the normal extensive, time consuming testing. Some testing prior to fielding is necessary of course (to ensure the standardized baseline expert system is working), but the majority of small, idiosyncratic changes (local SOP's, unit peculiar regulations, etc.) can be made by the using unit and the product can be put to use immediately. The using unit would not find it necessary to return the system to the developer for modification.

Without addressing the legal issues involved, the author feels that copies of expert systems, such as CACSMAM, which have been developed for general use, should be made available immediately to units by sending the products to the units without waiting for requests (all too often, units don't even know what is available). In this manner, expert system technology can be seen, used, and then understood by the rank-and-file, and may ultimately become a major factor in the command and control of small units who may never have access to larger systems.

C. SUMMARY

Expert system technology is now available to people who do not have a computer programming background, or those who feel that computers cannot do anything for them other than word processing or spreadsheets. Some feel that all too often, programs which are designed to do analysis and decision aiding are not useful because they either have too many "bugs" in the program, or were developed in isolation and are not applicable to the "real world". Yet unit commanders, especially those involved in communications, need automated help in managing their part of the complex battlefield. With the proliferation of microcomputers in army units, and unit personnel becoming more computer literate, the potential of microcomputer-based expert systems should not be wasted.

This thesis has shown that microcomputer-based expert systems can be useful (or if they are not "quite right", they can be easily modified), small, mobile, adaptable, "expert" in their fields, and get better with age as more experts place knowledge into the systems. It is the responsibility of those who have access to this technology now, to advertise it and get it into the hands of those in the military who need the help the most, the small units. It will be the responsibility of the small units to realize the potential of the expert systems and make them an integral part of the unit command and control.

However, as with any labor saving device, a risk of dependency is present. The author strongly feels that while expert systems should be developed and used wherever possible, training for decision makers (for whom the expert system is to aid) should never stop. An inherent danger associated with the prolonged use of a comprehensive expert system is that failure of the system may cripple the organization dependent on it. Therefore, while the system is in use, and taking full advantage of the expert system's ability to explain its reasoning, the system can train individuals at the same time as it is rendering decisions. Even though we design expert systems to possibly "take the place of an expert" we must continually train personnel to "take the place of the expert system".

In conclusion, the author offers a quote from Dr. Lawrence J. Korb, Assistant Secretary of Defense for Manpower, Reserve Affairs and Logistics, in which he summarizes quite nicely the relationship which must be present between man and machine on the future battlefield.

> There is no substitute for man's judgement when the computer is without eyes and lacks the information to recognize a new and untried situation, and there is no substitute for man's calculations of the consequences of his actions. However, in the present and coming electronic battle there will not be time enough for man to do it all and still accomplish his mission and survive. The thinking machine, some sort of created intelligence, is going to be necessary. In fact, it is necessary already, if man is going to have the information available to him to make those decisions only a man can and should make. [Ref. 21]

APPENDIX A
GLOSSARY OF KEY TERMS AND ACRONYMS

ADA - Air Defense Artillery Battalion

AI - Artificial Intelligence

Antecedent - the "IF" part of a production rule

1 BDE - First (1) Brigade

2 BDE - Second (2) Brigade

3 BDE - Third (3) Brigade

Backward Chaining - A control mechanism that seeks to satisfy a stated goal by seeking rules in which the THEN portion matches the goal, then seeking other rules whose THEN portions match the IF portion of the rule which satisfies the goal [Ref. 13: p. 165]

BLACK - Designated name for a division main (DIVMAIN) site

BLOC - Battalion Logistics Operations Center

Breadth - In an hierarchy of rules, breadth refers to all of the rules which are on the same level of the hierarchy. This is in contrast with depth. [Ref. 11: p. 257]

CAB - Combat Aviation Brigade

Cache - name given the memory space allocated to the expert system for storing conclusions

CEMS - Communications-Electronics Management System

Certainty - the degree of confidence one has in a fact or relationship [Ref. 11: p. 258]

Certainty factor - A numerical weight (integers from -100 - 100 in M.1) given to a fact or relationship to indicate the confidence one has is the fact or relationship. These numbers behave differently than probability coefficients. In general, methods for manipulating certainty factors are more informal than approaches to combining probabilities [Ref. 11: p. 258]

CESE - Communications-Electronics System Element

CNCE - Communications Nodal Control Element

Conclusion - the "THEN" part of a production rule

CSCE - Communications System Control Element (also known informally as SYSCON)

CSPE - Communications System Planning Element

Depth - In an hierarchy of rules, depth refers to the rule on the highest level and all the rules immediately below that one in the hierarchy. This is in contrast to breadth. [Ref. 11: p. 259]

DISCOM - Division Support Command

DIVARTY - Division Artillery

DIVMAIN - Division Main (site)

Domain - a topical area or region of knowledge

ENG - Engineer Battalion

FASC - Forward Area Signal Center

Forward Chaining - a control mechanism that seeks to identify all rules whose IF portions are true, then uses the THEN portions of those rules to find other rules which are also true [Ref. 13: p. 166]

GOLD - Designated name for a division main (DIVMAIN) site

Heuristics - rules of thumb and educated guesses that an expert uses in solving problems in his or her domain [Ref. 13: p. 166]

Inference Engine - the part of a knowledge based system that contains the procedures for reaching a conclusion

Instantiate - the process of assigning specific values to variables

MSE - Mobile Subscriber Equipment

Production Rule - the term used to describe an IF-THEN rule

Rule - a conditional statement of two parts. The first part, comprised of one or more IF clauses, establishes conditions that must apply if a second part, comprised of one or more THEN clauses, is to be acted upon [Ref. 11: p. 265]

SYSCON - System's Control (informal name for the CSCE)

TAC CP - Tactical Command Post

TACFIRE - Tactical Fire Control System

Validation - The process of establishing the fitness or worth of a software product for its operational mission. Informally, "Are we building the right product?" [Ref. Boehm: p. 37]

Verification - The process of establishing the truth of the correspondance between a software product and its specification. Informally, "Are we building the product right?" [Ref. 20: p. 37]

APPENDIX B
USER'S GUIDE TO CACSMAM

1. INTRODUCTION

CACSMAM is a small prototype expert system designed to aid the Battalion Maintenance Officer (BMO), or those personnel working with the BMO, in making asset management decisions. CACSMAM is only concerned with AN/TRC 145 Radio Terminal multichannel asset management for the maintenance of the divisional tactical multichannel communications network. It will not currently provide solutions for any other hardware system. It also does not include any system troubleshooting.

This user's guide is written on the assumption that the reader is familiar with the following:

1. use of a microcomputer,
2. communication terminology used in the management of a multichannel system,
3. the M.1 Knowledge Development Tool (version 2.0), and
4. the unique terminology which is associated with expert systems.

This guide will not go into detail on the use of a computer, the specific operating system used, nor any of the terms used in the program which are communications or M.1 specific. It is only intended to be a brief guide for the user for CACSMAM alone. Any references mentioned in this guide are documented at the end of the thesis of which this guide is an appendix.

a. Purpose of CACSMAM

CACSMAM is designed to help the BMO (or the BLOC personnel in the BMO's absence) decide what action to take when an AN/TRC 145 Radio Terminal is down due to a component failure (such as a receiver, multiplexer, crypto, etc.). It will ask questions of the user as to what system is down and at what site the problem is. It will continue to ask questions to determine what assets are available and where they are, as well as where the nearest maintenance team is to the problem (if one is needed).

Once the user has input enough information for CACSMAM to reach a conclusion, an **appropriate-action** will be presented to the user. The **appropriate-action** will be one which possibly recommends a reallocation of assets between stacks, rigs or possibly sites.

b. Specifications for System

CACSMAM requires an IBM-PC compatable microcomputer with at least 512K bytes of RAM running PC-DOS 2.0 or later. Also required is one 5 1/4" floppy disk drive. A color monitor is recommended but not necessary.

CACSMAM consists of only one executable file called **CACSMAM.EXE** on one floppy disk.

2. USING CACSMAM

a. Initial Setup

After turning on your microcomputer and its monitor, and "booting" up the system with PC-DOS 2.0 or later, the screen should be displaying the characteristic DOS prompt

A>

or the designation of the current default drive.

Insert the CACSMAM diskette into the floppy disk drive designated in the prompt and type CACSMAM (lower or upper case). The initial full screen of text welcoming you to CACSMAM will be displayed (see the sample session in Section 3.1). Underneath the welcoming text is a prompt to strike any lowercase key to continue.

The default setting for CACSMAM is for a color monitor. If you have a monochrome monitor, type

> >colors off

instead of a single lowercase key. This will improve the contrast of the text on the screen for easier reading.

To continue with CACSMAM, just answer each question in the proper format, of which there are three:

- a menu listing --- select the number of the appropriate answer and follow with a <carriage return>
- ***any lower-case expression*** --- select any lowercase key on the keyboard to continue the consultation and follow with a <carriage return>
- ***a number*** --- select any number (integer or real/positive or negative) and follow with a <carriage return>

While several commands are available for use in M.1, most have been disabled in the CACSMAM program for purposes of simplicity. The only allowable commands are explained in the following section.

b. **Commands Available**

All allowable commands within M.1 and CACSMAM are to be entered in *lowercase letters only*!!

1. abort

The **abort** command can be used at any time during the consultation. It will return the user to the

CACSMAM > >

prompt. This is particularly useful if the user has answered a question incorrectly as CACSMAM does not give the user the capability to change wrong answers. This command terminates the consultation.

2. colors on/off

The **colors on/off** command allows for the use of either color or monochrome monitors. The default setting for CACSMAM is **colors on.**

3. go

Typing go at the

CACSMAM > >

prompt begins a consultation with CACSMAM. This command is only used at the top level CACSMAM prompt. However, the go command is not needed on the initial consultation after the CACSMAM diskette has been loaded and the WELCOME screen has appeared. The system automatically starts a consultation at that point. The go command is only needed for subsequent consultations.

4. help/COMMAND

Typing **help COMMAND** (where COMMAND is the one for which the user needs help) will cause the appropriate M.1 help message to appear on the screen. For example:

> > help go

5. list

Typing **list** at the

CACSMAM > >

prompt will provide the user with a listing of the entire knowledge base used in CACSMAM. Typing **list RULE** (where RULE is the number of the rule of interest) will provide the user with a listing of the rule (example: **list kb-165**). This command is often used in conjunction with the **why** command.

6. quit/exit

Typing **quit** or **exit** at any time will return the user to the DOS system and the

A >

prompt.

7. why

Typing **why** at any time in answer to a question will provide the user with the reasoning that CACSMAM is using at the current time in asking the particular question. The reasoning is shown as a listing of the rules under consideration at the time the question is asked. Quite often more than one rule is under consideration, but only one rule is fully listed, with only the numbers of the others under consideration are given. The **list** command is useful in conjunction with this command in this case.

8. show

Typing **show** at anytime will provide the user with a listing of all the items of information CACSMAM is using to reach a conclusion (which are those currently stored in the cache). The values shown will have either been obtained from the user (responses to questions), through reasoning (conclusions to rules in the knowledge base which prove true), or through facts permanently stored in the CACSMAM knowledge base.

9. CTRL BREAK

Typing **<CTRL BRK>** (actually hitting the **CONTROL** and **BREAK** keys at the same time) at any time will return the user to a command level one higher than the one the user was in when the command was typed. For example, within a consultation, at the

> >

prompt, the user types

> > <CTRL BRK>

The system will "pop" the user up to the top level to the

CACSMAM > >

prompt.

c. **Using Certainty Factors**

To allow for the user to communicate uncertainty to the system ("I'm *pretty sure* that the backup stack is good."), CACSMAM accepts certainty factors along with its answers. Certainty factors range from -100 (absolutely sure that the answer is false) to 100 (absolutely sure that the answer is true) and are input to CACSMAM in the format

answer cf XX

where **answer** is the response to the question, **cf** is the required keyword for inputting a certainty factor, and **XX** is the number representing the certainty the user has in his answer. For example, CACSMAM asks the following question

Is the backup stack operational?

1. yes
2. no
3. glossary

to which the user may respond (for a "pretty sure" answer, say 80% sure)

> > 1 cf 80 .

If no certainty factor is input by the user, CACSMAM assumes that the answer is totally true and acts as if a certainty factor of 100 was assigned to the answer.

Only a few questions in CACSMAM explicitly ask the user to input certainty factors. However, the user may input certainty factors in response to any question posed by the system, if the user has any uncertainty whatsoever in the response.

d. **Errors**

If at any time the user enters something that is either not in the proper format, or something that CACSMAM, or more correctly, M.1, does not recognize as correct, the system will respond with an error message. The error message presented is the one programmed into the M.1 system shell, and not in CACSMAM. The user may refer to the M.1 reference manual for further details.

Once M.1 has displayed the appropriate error message however, the last question asked will be redisplayed and the user will be reprompted for an answer.

3. SAMPLE CACSMAM CONSULTATION

C> cacsmam

WELCOME TO CACSMAM!!!

COMPUTER AIDED COMMUNICATIONS SYSTEM MAINTENANCE MANAGER!!!!

An investigation into the the usefulness of an expert system as an aid to systems maintenance managers in the field.

This system is designed to be a prototype microcomputer based expert system to aid the Battalion Maintenance Officer (BMO) or his staff of the divisional signal battalion make the best decision as to what action to take when a multichannel communication node fails. This decision aid will make recommendations as to the reallocation of available assets from within the signal battalion.

This system currently assumes that the operator of the multichannel communications shelter (rig) is competent enough to troubleshoot the problem accurately enough to determine that a component has failed and thus requires replacement (either with or without maintenance personnel support).

Strike any lowercase key to continue (follow with carriage return).

```
      any lower-case expression.
>> f
```

Has the operator done a complete operational check of his rig?

```
      1. yes
      2. no
>> 1
```

Has the operator had his supervisor check out the problem and verify it?

```
      1. yes
      2. no
>> 1
```

Have you reported the verified outage to SYSCON?

```
      1. yes
      2. no
>> 2
```

Get someone to submit a status report ASAP
while you consult this program.

Strike any lowercase key to continue (follow with carriage return).

any lower-case expression.
>> s

What is the system designator of the troubled system?

1. 5666PAA
2. 5666PBA
3. 5667PAA
4. 5621PAA
5. 5622PAA
6. 5623PAA
7. 5665PAA
8. 6568PAA
9. 5664PAA
10. 6667PAA
11. 6621PAA
12. 6622PAA
13. 6623PAA
14. 6665PAA
15. 6668PAA
16. 6669PAA
17. 6662PAA
18. 6664PAA
19. 6401PAA
20. 6402PAA
21. other

>> 20

At what site is the troubled system located?

1. 01
2. 02
3. 21
4. 22
5. 23
6. 56
7. 61
8. 62
9. 64
10. 65
11. 66
12. 67
13. 68
14. 69
15. other

>> 1

You have made an error in inputting either your system or your site designators. You input the system as 6402PAA and the site as 01. Please reinput at the appropriate prompts.

Strike any lowercase key to continue (follow with carriage return).

any lower-case expression.

> > 20

What is the system designator of the troubled system?

1. 5666PAA
2. 5666PBA
3. 5667PAA
4. 5621PAA
5. 5622PAA
6. 5623PAA
7. 5665PAA
8. 6568PAA
9. 5664PAA
10. 6667PAA
11. 6621PAA
12. 6622PAA
13. 6623PAA
14. 6665PAA
15. 6668PAA
16. 6669PAA
17. 6662PAA
18. 6664PAA
19. 6401PAA
20. 6402PAA
21. other

> > 2

At what site is the troubled system located?

1. 01
2. 02
3. 21
4. 22
5. 23
6. 56
7. 61
8. 62
9. 64
10. 65
11. 66
12. 67
13. 68
14. 69
15. other

> > 6

What is the primary component in the system which is suspected of causing the system problems?

1. transmitter
2. receiver
3. multiplexer
4. crypto
5. transmitter case
6. receiver case
7. cable combiner
8. converter
9. other

> > 9

You have indicated that a major item of equipment does not need to be replaced. Please type in the part that you need. (Please use enclose your reply in single (') quotes, e.g. 'TSA2 panel').

any lower-case expression.
> > '11A23 panel'

Do you require personnel from the signal maintenance section to assist in replacing the failed component?

1. yes
2. no
> > 2

Is a spare 11A23 panel on site 56?

1. yes
2. no
3. glossary
> > sdf
Error 224. The response entered is not a legal value. Please re-enter.

Is a spare 11A23 panel on site 56?

1. yes
2. no
3. glossary
> > 1

Is a backup stack available on site 56?

1. yes
2. no
3. glossary
> > 1

Is the backup stack operational?

1. yes
2. no
3. glossary

> > 3

THIS IS THE GLOSSARY:

backup stack = A TRC 145 system stack which is not currently committed as an active system or in a jump rig.
jump = relocation (either planned or hasty) of a communications site due to tactical considerations as determined by the subscriber or customer.
jump rig = A TRC 145 which is committed to any upcoming jumps of the system.
operational = The part, component, stack, or rig actually works.
spare part = The spare part or float item of equipment which is needed to bring the system up.
stack = The set of equipment inside the TRC 145 for one system consisting of the antenna, receiver, transmitter, crypto, etc.
system = The actual network multichannel communications system engineered by SYSCON, delineated from terminal to terminal, NOT end-user to end-user.
up = The same as operational.

Strike any lowercase key to continue (follow with carriage return).

any lower-case expression.

> > d

Is the backup stack operational?

1. yes
2. no
3. glossary

> > 2

Even though the whole backup stack is not operational is the 11A23 panel out of the backup stack operational?

1. yes
2. no
3. glossary

> > why

M.1 is trying to determine whether the following rule is applicable in this consultation:

```
kb-124:
    if component = '11A23 panel' and
        site-designator = '56' and
        backup-stack is on-site- '56' and
        backup-stack is not-operational and
        not('11A23 panel'-in-backup-stack is operational)
    then '11A23 panel'-in-backup-stack is not-operational.
```

The following entries are also under consideration:

kb-125(a rule)
kb-65(a rule)
kb-4(a goal)

Even though the whole backup stack is not operational is the 11A23 panel out of the backup stack operational?

1. yes
2. no
3. glossary

> > list kb-125
kb-125:
if site-designator = Y and
component = PART and
backup-stack is on-site-Y and
backup-stack is not-operational and
PART-in-backup-stack is not-operational
then PART-in-backup-stack is not-available.

Even though the whole backup stack is not operational is the 11A23 panel out of the backup stack operational?

1. yes
2. no
3. glossary

> > 1

Approx. how many hours will it take to repair the system 5666PBA given that the spare component is available at site 56?
Please include a certainty factor in your reply.
(In the form 'answer cf number')

a number.
> > 1.0 cf 85

How long would it take to remove the appropriate part from another stack, put it in the system, and begin logging in channels? Please include a certainty factor. (In the form 'answer cf factor')

a number.
> > .7 cf 80

Is a tactical jump planned for site 56 that you know of?

```
          1.  yes
          2.  no
          3.  glossary
>> 2
          appropriate-action = cannibalize-11A23 panel-from-backup-stack (72%)
          because kb-70.
CACSMAM->>show
          system-designator = 5666PBA (100%) because you said so.
          site-designator = 56 (100%) because you said so.
          system-site-check-ok = yes (100%) because kb-42.
          system-site-test = yes (100%) because kb-40.
          item = 11A23 panel (100%) because you said so.
          component = 11A23 panel (100%) because set by user.
          minor-part is identified = yes (100%) because kb-52.
          maintenance-team is required = no (100%) because you said so.
          appropriate-action = cannibalize-11A23 panel-from-backup-stack (72%)
          because kb-70.
          spare-11A23 panel is on-site-56 = yes (100%) because you said so.
          spare-11A23 panel is available = yes (100%) because kb-99.
          backup-stack is on-site-56 = yes (100%) because you said so.
          backup-stack is operational = no (100%) because you said so.
          opposite(no) = yes (100%) because kb-27.
          backup-stack is not-operational = yes (100%) because kb-128.
          11A23 panel-in-backup-stack is not-operational was sought, but no
          value was concluded.
          11A23 panel-in-backup-stack is operational = yes (100%) because you
          said so.
          opposite(yes) = no (100%) because kb-26.
          backup-stack is not-on-site-56 = no (100%) because kb-127.
          11A23 panel-in-backup-stack is available = yes (100%) because kb-123.
          11A23 panel-in-backup-stack is not-available = no (100%) because
          kb-130.
          time-to-repair-system-5666PBA-on-site-56-with-spare-11A23 panel =
          1.0 (85%) because kb-106.
          maintenance-team is not-required = yes (100%) because kb-60.
          repair-time-for-system-5666PBA-on-site-56 = 1.0 (85%) because you
          said so.
          time-until-repairs-can-begin was sought, but no value was concluded.
          spare-11A23 panel is not-on-site-56 = no (100%) because kb-141.
          time-to-cannibalize-a-stack = 0.7 (80%) because you said so.
          a-tactical-jump-of-site-56 is planned = no (100%) because you said
          so.
          a-tactical-jump-of-site-56 is not-planned = yes (100%) because
          kb-140.
          replacement-can-be-made-before-jump was sought, but no value was
          concluded.
          spare-11A23 panel is not-available was sought, but no value was
          concluded.
          backup-stack is not-available was sought, but no value was concluded.
CACSMAM->>quit

C>
```

APPENDIX C
KNOWLEDGE BASE FOR CACSMAM

```
configuration(banner) =('

                        WELCOME TO CACSMAM!!!

      COMPUTER AIDED COMMUNICATIONS SYSTEM MAINTENANCE MANAGER!!!!

      An investigation into the the usefulness of an expert system
         as an aid to systems maintenance managers in the field.

   This system is designed to be a prototype microcomputer based expert
system to aid the Battalion Maintenance Officer (BMO) or his staff of
the divisional signal battalion make the best decision as to what action
to take when a multichannel communication node fails.  This decision aid
will make recommendations as to the reallocation of available assets
from within the signal battalion.
   This system currently assumes that the operator of the multichannel
communications shelter (rig) is competent enough to troubleshoot the
problem accurately enough to determine that a component has failed and
thus requires replacement (either with or without maintenance personnel
support).

').

/*------------------INITIALIZE THE SYSTEM-----------------------------*/

configuration(startup) = go.
configuration(prompt) = 'CACSMAM->>'.

goal = appropriate-action.
multivalued(appropriate-action).

initialdata = [screen-change,status-verification-1,
status-verification-2,
status-verification-3,system-designator,site-designator,
system-site-test,component,maintenance-team is required].

nocache(paginate).
nocache(operator-check).
nocache(supervisor-check).
nocache(status-report-check).
nocache(status-verification-1).
nocache(status-verification-2).
nocache(status-verification-3).
nocache(screen-change).
nocache(glossary is printed).
nocache(system-site-match).

nolist(system-designator).
nolist(site-designator).
nolist(opposite(yes)).
nolist(opposite(no)).
nolist(system-site-test).

disable(adda).
disable(add).
disable(core).
disable(find).
disable(loadcache).
disable(options).
disable(panels).
```

```
disable(remove).
disable(savecache).
disable(trace).
disable(traced).
disable(uses).

whenfound(X = glossary) = [glossary is printed, do(reset X), X].

  if display([nl,nl,'
                                 THIS IS THE GLOSSARY:
    backup stack = A TRC 145 system stack which is not currently committed
                                 as an active system or in a jump rig.
    jump = relocation (either planned or hasty) of a communications site
                 due to tactical considerations as determined by the subscriber
                 or customer.
    jump rig = A TRC 145 which is committed to any upcoming jumps of the
                          system.
    operational = The part, component, stack, or rig actually works.
    spare part = The spare part or float item of equipment which is needed
                             to bring the system up.
    stack = The set of equipment inside the TRC 145 for one system
                   consisting of the antenna, receiver, transmitter, crypto, etc.
    system = The actual network multichannel communications system
                     engineered by SYSCON, delineated from terminal to terminal,
                     NOT end-user to end-user.
    up = The same as operational.
      ',nl,nl])
     and paginate is sought
  then glossary is printed.

  if paginate is sought
  then screen-change.

question(paginate) = ['
Strike any lowercase key to continue (follow with carriage return).',
nl].

opposite(yes) = no.
opposite(no) = yes.


/*--------------END INITIALIZATION-----------------------------------*/


/*-----------CHECK USER PREPAREDNESS TO USE SYSTEM--------------------*/

automaticmenu(ALL).
enumeratedanswers(ALL).

question(system-designator) = [nl,nl,'
What is the system designator of the troubled system?',nl].

legalvals(system-designator) = ['5666PAA','5666PBA','5667PAA',
          '5621PAA','5622PAA',
          '5623PAA','5665PAA','6568PAA','5664PAA','6667PAA',
          '6621PAA','6622PAA',
          '6623PAA','6665PAA','6668PAA','6669PAA','6662PAA',
          '6664PAA','6401PAA','6402PAA','other'].

whenfound(system-designator = other) = [new-system is input].

if system-number is sought
   and system-number = X
   and do(set system-designator = X)
then new-system is input.

question(system-number) = [nl,nl,nl,'
Since the system designator of the troubled system was not listed,
please input the system designator enclosed in single ('') quotes, e.g.
```

```
''5675PAA''.',nl,nl,nl].

question(site-designator) = [nl,nl,'
At what site is the troubled system located?',nl].

legalvals(site-designator) = ['01','02','21','22','23','56',
'61','62','64','65','66','67','68','69',other].

whenfound(site-designator = other) = [new-site is input].

if site-number is sought
  and site-number = X
  and do(set site-designator = X)
then new-site is input.

question(site-number) = [nl,nl,nl,'
Since the site of the trouble was not listed, please input it here
enclosed in single ('') quotes, e.g. ''75''.',nl,nl,nl].

 if system-designator is sought
   and site-designator is sought
   and (system-site-match or
        system-site-check-ok)
 then system-site-test.

 if system-designator is sought
   and site-designator is sought
   and system-designator = X
   and site-designator = Y
   and ((substring(0,2,X)) = Y or
        (substring(2,2,X)) = Y)
 then system-site-match.

 if not(system-site-match)
   and system-designator = X
   and site-designator = Y
   and display([nl,nl,nl,'
       You have made an error in inputting either your system or
       your site designators.  You input the system as ',X,' and
       the site as ',Y,'.  Please reinput at the appropriate
       prompts.',nl,nl,nl])
   and paginate is sought
   and do(reset(system-designator))
   and do(reset(site-designator))
   and system-site-match
 then system-site-check-ok.

question(operator-check) = [nl,nl,nl,'
Has the operator done a complete operational check of his rig?',nl,nl,
nl].

legalvals(operator-check) = [yes,no].

question(supervisor-check) = [nl,nl,nl,'
Has the operator had his supervisor check out the problem and verify
it?',nl,nl,nl].

legalvals(supervisor-check) = [yes,no].

question(status-report-check) = [nl,nl,nl,'
Have you reported the verified outage to SYSCON?',nl,nl,nl].

legalvals(status-report-check) = [yes,no].

question(component) = [nl,nl,nl,'
What is the primary component in the system which is suspected of
causing the system problems?',nl,nl,nl].

legalvals(component) = [transmitter,receiver,multiplexer,crypto,
```

```
'transmitter case','receiver case','cable combiner',converter,other].

whenfound(component = other) = [minor-part is identified].

 if item is sought
  and item = X
  and do(set component = X)
 then minor-part is identified.

question(item) = [nl,nl,nl,'
You have indicated that a major item of equipment does not need
to be replaced.  Please type in the part that you need. (Please use
enclose your reply in single ('') quotes, e.g. ''18A2 panel'').
',nl,nl,nl].

question(maintenance-team is required) = [nl,nl,nl,'
Do you require personnel from the signal maintenance section to assist
in replacing the failed component?',nl,nl,nl].

legalvals(maintenance-team is required) = [yes,no].

whenfound(maintenance-team is required = yes) = [
site-location-of-maintenance-team, report-time-for-maintenance-team].

question(site-location-of-maintenance-team) = [nl,nl,nl,'
At what site location are available maintenance personnel located?
Please input your answer in single ('') quotes.',nl,nl,nl].

question(report-time-for-maintenance-team) = [nl,nl,nl,'
How long will it take for the maintenance personnel to arrive at the
site?',nl,nl,nl].

legalvals(report-time-for-maintenance-team) = number.

 if not(maintenance-team is required)
 then maintenance-team is not-required.

 if maintenance-team is required = A
   and opposite(A) = B
 then maintenance-team is not-required = B.

 if operator-check = no
   and display([nl,nl,nl,'Get the operator to check his system and then
      consult this system. GOODBYE!',nl,nl])
   and paginate is sought
   and do(reset)
   and do(restart)
 then  status-verification-1.

 if supervisor-check = no
   and display([nl,nl,nl,'Get a supervisor to check the system out then
     consult this system.  GOODBYE!',nl,nl,nl])
   and paginate is sought
   and do(reset)
   and do(restart)
 then status-verification-2.

 if status-report-check = no
   and display([nl,nl,nl,'Get someone to submit a status report ASAP
      while you consult this program.',nl,nl,nl])
   and paginate is sought
  then status-verification-3.

/*-------------- END USER PREPAREDNESS CHECK -------------------------*/

/*--------- CHECK ALL OPTIONS FOR APPROPRIATE ACTIONS ----------------*/

          /*******  REPAIR WITH SPARE PARTS **************/
```

```
if system-designator = X
  and site-designator = Y
  and component = PART
  and spare-PART is available
  and PART-in-backup-stack is not-available
  and PART-in-jump-rig is not-available
  and (a-tactical-jump-of-site-Y is not-planned or
    replacement-can-be-made-before-jump)
then appropriate-action = repair-system-X-on-site-Y-with-
                                       spare-PART cf 90.

if system-designator = X
  and site-designator = Y
  and component = PART
  and spare-PART is available
  and backup-stack is on-site-Y
  and backup-stack is operational
  and time-to-repair-system-X-on-site-Y-with-spare-PART = C
  and time-to-replace-system-X-with-another-stack = A
  and A >= C
  and (a-tactical-jump-of-site-Y is not-planned or
    replacement-can-be-made-before-jump)
 then appropriate-action = repair-system-X-on-site-Y-with-
                                       spare-PART cf 90.

if system-designator = X
  and site-designator = Y
  and component = PART
  and spare-PART is available
  and PART-in-backup-stack is available
  and time-to-repair-system-X-on-site-Y-with-spare-PART = C
  and time-to-cannibalize-a-stack = B
  and B >= C
  and (a-tactical-jump-of-site-Y is not-planned or
    replacement-can-be-made-before-jump)
then appropriate-action = repair-system-X-on-site-Y-with-
                                       spare-PART cf 90.

if system-designator = X
  and site-designator = Y
  and component = PART
  and spare-PART is available
  and PART-in-backup-stack is not-available
  and a-jump-rig is on-site-Y
  and PART-in-jump-rig is available
  and time-to-repair-system-X-on-site-Y-with-spare-PART = C
  and time-to-cannibalize-a-stack = B
  and B >= C
  and (a-tactical-jump-of-site-Y is not-planned or
    replacement-can-be-made-before-jump)
then appropriate-action = repair-system-X-on-site-Y-with-
                                       spare-PART cf 90.

if system-designator = X
  and site-designator = Y
  and component = PART
  and spare-PART is available
  and PART-in-backup-stack is not-available
  and PART-in-jump-rig is available
  and time-to-repair-system-X-on-site-Y-with-spare-PART = C
  and time-to-replace-system-X-with-another-stack = A
  and A >= C
  and (a-tactical-jump-of-site-Y is not-planned or
    replacement-can-be-made-before-jump)
then appropriate-action = repair-system-X-on-site-Y-with-
                                       spare-PART cf 90.

          /********** CANNIBALIZE PART FROM BACKUP STACK ********/
```

```
if system-designator = X
   and site-designator = Y
   and component = PART
   and spare-PART is available
   and backup-stack is on-site-Y
   and backup-stack is not-operational
   and PART-in-backup-stack is available
   and time-to-repair-system-X-on-site-Y-with-spare-PART = C
   and time-to-cannibalize-a-stack = B
   and B < C
   and (a-tactical-jump-of-site-Y is not-planned or
     replacement-can-be-made-before-jump)
then appropriate-action = cannibalize-PART-from-backup-stack cf 90.

if system-designator = X
   and site-designator = Y
   and component = PART
   and spare-PART is not-available
   and backup-stack is on-site-Y
   and backup-stack is not-operational
   and PART-in-backup-stack is available
   and (a-tactical-jump-of-site-Y is not-planned or
     replacement-can-be-made-before-jump)
then appropriate-action = cannibalize-PART-from-backup-stack cf 90.

if system-designator = X
   and site-designator = Y
   and component = PART
   and spare-PART is not-available
   and backup-stack is on-site-Y
   and backup-stack is operational
   and time-to-cannibalize-a-stack = A
   and time-to-replace-system-X-with-another-stack = B
   and A < B
   and (a-tactical-jump-of-site-Y is not-planned or
     replacement-can-be-made-before-jump)
then appropriate-action = cannibalize-PART-from-backup-stack cf 90.

              /********** REPLACE SYSTEM WITH BACKUP STACK ***********/

if system-designator = X
   and site-designator = Y
   and component = PART
   and spare-PART is available
   and backup-stack is on-site-Y
   and backup-stack is operational
   and time-to-repair-system-X-on-site-Y-with-spare-PART = C
   and time-to-replace-system-X-with-another-stack = A
   and A < C
   and (a-tactical-jump-of-site-Y is not-planned or
     replacement-can-be-made-before-jump)
then appropriate-action = replace-system-X-with-backup-stack cf 90.

if system-designator = X
   and site-designator = Y
   and component = PART
   and spare-PART is not-available
   and backup-stack is on-site-Y
   and backup-stack is operational
   and time-to-replace-system-X-with-another-stack = A
   and time-to-cannibalize-a-stack = B
   and A <= B
   and (a-tactical-jump-of-site-Y is not-planned or
     replacement-can-be-made-before-jump)
then appropriate-action = replace-system-X-with-backup-stack cf 90.

                  /************ CANNIBALIZE PART FROM JUMP RIG **********/
```

```
 if system-designator = X
   and site-designator = Y
   and component = PART
   and spare-PART is not-available
   and PART-in-backup-stack is not-available
   and a-jump-rig is on-site-Y
   and a-tactical-jump-of-site-Y is not-planned
   and jump-stack is not-operational
   and PART-in-jump-rig is available
 then appropriate-action = cannibalize-PART-from-jump-rig cf 90.

/****** CANNIBALIZE JUMP RIG AND REPAIR JUMP RIG WITH SPARE ****/

 if system-designator = X
   and site-designator = Y
   and component = PART
   and spare-PART is available
   and PART-in-backup-stack is not-available
   and a-jump-rig is on-site-Y
   and a-tactical-jump-of-site-Y is planned
   and jump-stack is not-operational
   and PART-in-jump-rig is available
   and time-to-repair-system-X-on-site-Y-with-spare-PART = A
   and time-to-cannibalize-a-stack = B
   and time-to-site-Y-jump = Z
   and B < A
   and B < Z
 then appropriate-action = cannibalize-PART-from-jump-rig-
                                  and-repair-jump-rig-with-spare-PART cf 90.

              /*********** REPLACE SYSTEM WITH JUMP STACK **********/

 if system-designator = X
   and site-designator = Y
   and component = PART
   and spare-PART is not-available
   and PART-in-backup-stack is not-available
   and a-jump-rig is on-site-Y
   and a-tactical-jump-of-site-Y is not-planned
   and jump-stack is operational
   and time-to-repair-system-X-on-site-Y-with-spare-PART = C
   and time-to-replace-system-X-with-another-stack = A
   and A <= C
 then appropriate-action = replace-system-X-with-jump-stack cf 90.

 if system-designator = X
   and site-designator = Y
   and component = PART
   and spare-PART is not-available
   and PART-in-backup-stack is not-available
   and a-jump-rig is on-site-Y
   and a-tactical-jump-of-site-Y is not-planned
   and jump-stack is operational
   and time-to-replace-system-X-with-another-stack = A
   and time-to-cannibalize-a-stack = B
   and A <= B
 then appropriate-action = replace-system-X-with-jump-stack cf 90.

/**************** REPAIR SYSTEM AT JUMP SITE *************************/

 if system-designator = X
   and site-designator = Y
   and component = PART
   and maintenance-team is not-required
   and spare-PART is available
   and backup-stack is not-available
   and a-tactical-jump-of-site-Y is planned
   and replacement-cannot-be-made-before-jump
 then appropriate-action = replace-PART-at-jump-site cf 90.
```

```
if system-designator = X
   and site-designator = Y
   and component = PART
   and maintenance-team is not-required
   and spare-PART is available
   and backup-stack is on-site-Y
   and backup-stack is not-operational
   and PART-in-backup-stack is not-available
   and a-tactical-jump-of-site-Y is planned
   and replacement-cannot-be-made-before-jump
then appropriate-action = replace-PART-at-jump-site cf 90.

    /******* CANNIBALIZE BACKUP AT JUMP SITE ******************/

if system-designator = X
   and site-designator = Y
   and component = PART
   and maintenance-team is not-required
   and spare-PART is not-available
   and backup-stack is on-site-Y
   and PART-in-backup-stack is available
   and a-tactical-jump-of-site-Y is planned
   and replacement-cannot-be-made-before-jump
then appropriate-action = cannibalize-backup-stack-at-jump-site-
                                 and-designate-stack-as-new-backup cf 85.

   /******* SEND MAINTENANCE TEAM TO JUMP SITE/REPLACE PART *****/

if system-designator = X
   and site-designator = Y
   and component = PART
   and maintenance-team is required
   and spare-PART is available
   and backup-stack is not-available
   and a-tactical-jump-of-site-Y is planned
   and replacement-cannot-be-made-before-jump
then appropriate-action = send-maintenance-team-to-jump-site-and-
                                 replace-PART-there cf 90.

if system-designator = X
   and site-designator = Y
   and component = PART
   and maintenance-team is required
   and spare-PART is available
   and backup-stack is on-site-Y
   and backup-stack is not-operational
   and PART-in-backup-stack is not-available
   and a-tactical-jump-of-site-Y is planned
   and replacement-cannot-be-made-before-jump
then appropriate-action = send-maintenance-team-to-jump-site-and-
                                 replace-PART-there cf 90.

/***** SEND MAINTENANCE TEAM TO JUMP SITE/CANNIBALIZE BACKUP ***/

if system-designator = X
   and site-designator = Y
   and component = PART
   and maintenance-team is required
   and spare-PART is not-available
   and backup-stack is on-site-Y
   and PART-in-backup-stack is available
   and a-tactical-jump-of-site-Y is planned
   and replacement-cannot-be-made-before-jump
then appropriate-action = send-maintenance-team-to-jump-site-and-
                                 cannibalize-backup-stack-there-
                                 and-designate-stack-as-new-backup cf 85.
```

```
/****************** REPLACE RIG WITH RIG FROM DISTANT SITE **********/

 if component = PART
   and system-designator = X
   and site-designator = Y
   and spare-PART is not-available
   and PART-in-backup-stack is not-operational
   and PART-in-jump-rig is not-available
   and other-backup is available-for-site-Y
   and a-tactical-jump-of-site-X is not-planned
   and approval-of-syscon-has-been-received
 then appropriate-action = replace-rig-with-backup-rig-from-
                                     other-site cf 80.

/*************** DELETION OF SYSTEM *********************************/

 if component = PART
   and system-designator = X
   and site-designator = Y
   and spare-PART is not-available
   and PART-in-backup-stack is not-available
   and PART-in-jump-rig is not-available
   and other-backup is not-available-for-site-Y
   and the-approval-of-syscon-has-been-obtained
 then appropriate-action = delete-system-X-until-further-directions-
                                      from-syscon cf 90.

 if system-designator = X
   and site-designator = Y
   and component = PART
   and spare-PART is not-available
   and PART-in-backup-stack is not-available
   and a-jump-rig is on-site-Y
   and jump-stack is not-operational
   and PART-in-jump-rig is available
   and a-tactical-jump-of-site-Y is planned
   and approval-of-syscon-has-been-obtained
 then appropriate-action = delete-system-X-until-further-directions-
                                     from-syscon cf 90.

/*********** CONSULT WITH SYSCON FOR OTHER ALTERNATIVES **************/

 if component = PART
   and system-designator = X
   and site-designator = Y
   and spare-PART is not-available
   and PART-in-backup-stack is not-available
   and PART-in-jump-rig is not-available
   and other-backup is not-available-for-site-Y
   and the-approval-of-syscon-has-not-been-obtained
 then appropriate-action = get-with-syscon-to-find-other-
                                     alternatives cf 90.

 if system-designator = X
   and site-designator = Y
   and component = PART
   and spare-PART is not-available
   and PART-in-backup-stack is not-available
   and a-jump-rig is on-site-Y
   and jump-stack is not-operational
   and PART-in-jump-rig is available
   and a-tactical-jump-of-site-Y is planned
   and approval-of-syscon-has-not-been-obtained
 then appropriate-action = get-with-syscon-to-find-other-
                                     alternatives cf 90.

/*--------END APPROPRIATE ACTION CHECK---------------------------------*/

/********* CHECK INTERMEDIATE ANTECEDENTS **********************/
```

```
/*---- REPLACEMENT CAN BE MADE BEFORE JUMP --------------------*/

 if system-designator = X
   and site-designator = Y
   and component = PART
   and backup-stack is on-site-Y
   and backup-stack is operational
   and a-tactical-jump-of-site-Y is planned
   and time-to-site-Y-jump = Z
   and time-to-replace-system-X-with-backup-stack = B
   and B <= Z
 then replacement-can-be-made-before-jump cf 90.

 if system-designator = X
   and site-designator = Y
   and component = PART
   and spare-PART is available
   and a-tactical-jump-of-site-Y is planned
   and time-to-site-Y-jump = Z
   and time-to-repair-system-X-on-site-Y-with-spare-PART = A
   and A < Z
 then replacement-can-be-made-before-jump cf 90.

 if system-designator = X
   and site-designator = Y
   and component = PART
   and backup-stack is on-site-Y
   and PART-in-backup-stack is available
   and a-tactical-jump-of-site-Y is planned
   and time-to-site-Y-jump = Z
   and time-to-cannibalize-the-backup-stack = C
   and C < Z
 then replacement-can-be-made-before-jump cf 90.

 if system-designator = X
   and not(replacement-can-be-made-before-jump)
 then replacement-cannot-be-made-before-jump.

/*----------- TIME TO CANNIBALIZE STACKS ----------------------*/

 if system-designator = X
   and site-designator = Y
   and component = PART
   and maintenance-team is not-required
   and PART-in-backup-stack is available
   and time-to-cannibalize-a-stack = A
 then time-to-cannibalize-the-backup-stack = A.

 if system-designator = X
   and site-designator = Y
   and component = PART
   and maintenance-team is required
   and report-time-for-maintenance-team = B
   and PART-in-backup-stack is available
   and time-to-cannibalize-a-stack = A
 then time-to-cannibalize-the-backup-stack = A + B.

 if system-designator = X
   and site-designator = Y
   and component = PART
   and maintenance-team is not-required
   and a-jump-rig is on-site-Y
   and PART-in-jump-rig is available
   and time-to-cannibalize-a-stack = A
 then time-to-cannibalize-the-jump-stack = A.

 if system-designator = X
   and site-designator = Y
```

```
   and component = PART
   and maintenance-team is required
   and report-time-for-maintenance-team = B
   and a-jump-rig is on-site-Y
   and PART-in-jump-rig is available
   and time-to-cannibalize-a-stack = A
 then time-to-cannibalize-the-jump-stack = A +B.

/*----------- TIME TO REPLACE STACK WITH OTHER STACKS ---------*/

 if system-designator = X
   and site-designator = Y
   and component = PART
   and backup-stack is on-site-Y
   and backup-stack is operational
   and time-to-replace-system-X-with-another-stack = B
 then time-to-replace-system-X-with-backup-stack = B.

/*------------- AVAILABILTY OF SPARE PARTS --------------------*/

 if site-designator = Y
    and component = PART
    and spare-PART is on-site-Y
    or spare-PART is available-at-another-site
  then spare-PART is available.

 if site-designator = Y
    and component = PART
    and spare-PART is not-on-site-Y
    and spare-PART is not-available-at-another-site
  then spare-PART is not-available.

/*----------- TIME UNTIL REPAIRS CAN BEGIN --------------------*/

 if component = PART
   and site-designator = Y
   and spare-PART is on-site-Y
   and maintenance-team is required
   and report-time-for-maintenance-team = A
 then time-until-repairs-can-begin = A.

 if component = PART
   and site-designator = Y
   and maintenance-team is required
   and spare-PART is not-on-site-Y
   and spare-PART is available-at-another-site
   and site-location-of-maintenance-team = A
   and location-of-spare-PART = B
   and A = B
   and report-time-for-maintenance-team = C
 then time-until-repairs-can-begin = C.

 if component = PART
   and site-designator = Y
   and maintenance-team is required
   and spare-PART is not-on-site-Y
   and spare-PART is available-at-another-site
   and site-location-of-maintenance-team = A
   and location-of-spare-PART = B
   and (A < B or A > B)
   and report-time-for-maintenance-team = C
   and time-to-get-spare-PART-to-site-Y = D
   and C >= D
 then time-until-repairs-can-begin = C.

 if component = PART
   and site-designator = Y
   and maintenance-team is required
   and site-location-of-maintenance-team = A
```

```
   and spare-PART is not-on-site-Y
   and location-of-spare-PART = B
   and (A < B or A > B)
   and report-time-for-maintenance-team = C
   and time-to-get-spare-PART-to-site-Y = D
   and D > C
 then time-until-repairs-can-begin = D.

 if component = PART
   and site-designator = Y
   and maintenance-team is not-required
   and spare-PART is not-on-site-Y
   and spare-PART is available-at-another-site
   and time-to-get-spare-PART-to-site-Y = A
 then time-until-repairs-can-begin = A.

/*--------- TIME TO REPAIR SYSTEM WITH SPARE PART -------------*/

 if site-designator = Y
   and component = PART
   and system-designator = X
   and maintenance-team is not-required
   and spare-PART is on-site-Y
   and repair-time-for-system-X-on-site-Y = Z
 then time-to-repair-system-X-on-site-Y-with-spare-PART = Z.

 if site-designator = Y
   and component = PART
   and system-designator = X
   and spare-PART is on-site-Y
   and time-until-repairs-can-begin = W
   and repair-time-for-system-X-on-site-Y = Z
 then time-to-repair-system-X-on-site-Y-with-spare-PART = Z + W.

 if system-designator = X
   and site-designator = Y
   and component = PART
   and spare-PART is not-on-site-Y
   and spare-PART is available-at-another-site
   and maintenance-team is not-required
   and repair-time-for-system-X-on-site-Y = A
   and time-to-get-spare-PART-to-site-Y = B
 then time-to-repair-system-X-on-site-Y-with-spare-PART = A + B.

/*--------- CHECK OTHER SITES FOR ASSETS ----------------------*/

 if component = PART
   and system-designator = X
   and site-designator = Y
   and spare-PART is not-available
   and PART-in-backup-stack is not-operational
   and PART-in-jump-rig is not-available
 then check-other-sites-for-assets.

whenfound(check-other-sites-for-assets) = [other-backup is available].

question(other-backup is available-for-site-Y) = [nl,nl,nl,'
Is there a backup rig available at another site which can be moved
 to site ',Y ,' to take the place of the troubled rig?',nl,nl,nl].

legalvals(other-backup is available-for-site-Y) = [yes,no].

 if site-designator = Y
   and not(other-backup is available-for-site-Y)
 then other-backup is not-available-for-site-Y.

/*------------- SET UP TIMES FOR SYSTEM FROM OTHER SITE -------*/

 if system-designator = X
```

```
   and site-designator = Y
   and component = PART
   and a-tactical-jump-of-site-Y is planned
   and time-to-site-Y-jump = A
   and PART-in-backup-stack is not-available
   and other-backup is available-for-site-Y
   and arrival-time-of-new-backup-rig-to-site = B
   and set-up-time-of-new-system = C
   and B + C <= A
 then replacement-can-be-made-before-jump cf 85.

whenfound(other-backup is available) = [arrival-time-of-new-backup-
            rig-to-site,set-up-time-of-new-system,approval-of-syscon-
            has-been-received].

question(arrival-time-of-new-backup-rig-to-site) = [nl,nl,nl,'
How long will it take for the backup rig at the other site to travel
to the troubled site and begin setting up?  Please include a certainty
 factor in your reply. (In the form ''answer cf number'')',
 nl,nl,nl].

legalvals(arrival-time-of-new-backup-rig-to-site) = number.

question(set-up-time-of-new-system) = [nl,nl,nl,'
How long will it take for the backup team from the other site to
set up and begin logging in channels once they get here?  Please
include a certainty factor in your reply.
(In the form ''answer cf factor'')',
nl,nl].

legalvals(set-up-time-of-new-system) = number.

question(approval-of-syscon-has-been-received) = [nl,nl,nl,'
SYSCON must be consulted before any rigs are transferred between sites.
We are now considering moving the distant backup rig to the troubled
location.  Has SYSCON approved the relocation of the backup rig?',
nl,nl,nl].

legalvals(approval-of-syscon-has-been-received) = [yes,no].

/*-------------- AVAILABLITY OF PARTS IN JUMP/BACKUP RIGS -----*/

 if site-designator = Y
   and component = PART
   and backup-stack is on-site-Y
   and backup-stack is operational
 then PART-in-backup-stack is available.

 if site-designator = Y
   and component = PART
   and backup-stack is on-site-Y
   and backup-stack is not-operational
   and PART-in-backup-stack is operational
 then PART-in-backup-stack is available.

 if component = PART
   and site-designator = Y
   and backup-stack is on-site-Y
   and backup-stack is not-operational
   and not(PART-in-backup-stack is operational)
 then PART-in-backup-stack is not-operational.

 if site-designator = Y
   and component = PART
   and backup-stack is on-site-Y
   and backup-stack is not-operational
   and PART-in-backup-stack is not-operational
 then PART-in-backup-stack is not-available.
```

```
if site-designator = Y
    and component = PART
    and backup-stack is not-on-site-Y
then PART-in-backup-stack is not-available.

if site-designator = Y
   and backup-stack is on-site-Y = A
   and opposite(A) = B
then backup-stack is not-on-site-Y = B.

if backup-stack is operational = A
   and opposite(A) = B
then backup-stack is not-operational = B.

if site-designator = Y
   and backup-stack is not-on-site-Y
   and other-backup is not-available-for-site-Y
then backup-stack is not-available.

if component = PART
   and PART-in-backup-stack is available = A
   and opposite(A) = B
then PART-in-backup-stack is not-available = B.

if component = PART
   and PART-in-backup-stack is not-available = A
   and opposite(A) = B
then  PART-in-backup-stack is available = B.

if site-designator = Y
   and component = PART
   and a-jump-rig is on-site-Y
   and jump-stack is operational
then PART-in-jump-rig is available.

if site-designator = Y
   and component = PART
   and a-jump-rig is on-site-Y
   and jump-stack is not-operational
   and PART-in-jump-rig is operational
then PART-in-jump-rig is available.

if site-designator = Y
   and component = PART
   and jump-rig is not-on-site-Y
then PART-in-jump-rig is not-available.

if site-designator = Y
   and component = PART
   and a-jump-rig is on-site-Y
   and jump-stack is not-operational
   and PART-in-jump-rig is not-operational
then PART-in-jump-rig is not-available.

if component = PART
   and jump-stack is operational
then PART-in-jump-stack is operational.

if component = PART
   and backup-stack is operational
then PART-in-backup-stack is operational.


/*------------ SET UP NEGATION CLAUSES -------------------------------*/

if component = PART
   and not(PART-in-jump-rig is operational)
then PART-in-jump-rig is not-operational.
```

```
if not(other-backup is available)
then other-backup is not-available.

if site-designator = Y
   and not(a-tactical-jump-of-site-Y is planned)
then a-tactical-jump-of-site-Y is not-planned.

if site-designator = Y
   and component = PART
   and spare-PART is on-site-Y = A
   and opposite(A) = B
then spare-PART is not-on-site-Y = B.

if component = PART
   and spare-PART is available-at-another-site = A
   and opposite(A) = B
then spare-PART is not-available-at-another-site = B.

if repairs-will-be-completed-before-jump = A
   and opposite(A) = B
then repairs-will-not-be-completed-before-jump = B.

if site-designator = Y
   and a-tactical-jump-of-site-Y is planned = A
   and opposite(A) = B
then a-tactical-jump-of-site-Y is not-planned = B.

if component = PART
    and not(spare-PART is available)
then spare-PART is not-available.

if jump-stack is operational = A
    and opposite(A) = B
 then jump-stack is not-operational = B.

if not(jump-stack is operational)
then jump-stack is not-operational.

if site-designator = Y
   and a-jump-rig is on-site-Y = A
   and opposite(A) = B
then jump-rig is not-on-site-Y = B.

if site-designator = Y
   and not(a-jump-rig is on-site-Y)
then jump-rig is not-on-site-Y.

if site-designator = Y
   and a-tactical-jump-of-site-Y is planned = A
   and opposite(A) = B
then a-tactical-jump-of-site-Y is not-planned = B.

if site-designator = Y
   and a-tactical-jump-of-site-Y is not-planned = A
   and opposite(A) = B
then a-tactical-jump-of-site-Y is planned = B.

if component = PART
   and PART-in-jump-rig is available = A
   and opposite(A) = B
then PART-in-jump-rig is not-available = B.

if component = PART
   and PART-in-jump-rig is not-available = A
   and opposite(A) = B
then PART-in-jump-rig is available = B.

if not(the-approval-of-syscon-has-been-obtained)
then the-approval-of-syscon-has-not-been-obtained.
```

```
/*----------------------END NEGATION CLAUSES-------------------------*/

/*--------------------- BEGIN QUESTIONS ------------------------------*/

question(backup-stack is on-site-Y) = [nl,nl,nl,'
Is a backup stack available on site ',Y , '?',nl,nl,nl].

legalvals(backup-stack is on-site-Y) = [yes,no,glossary].

question(backup-stack is operational) = [nl,nl,nl,'
Is the backup stack operational?',nl,nl,nl].

legalvals(backup-stack is operational) = [yes,no,glossary].

question(PART-in-backup-stack is operational) = [nl,nl,nl,'
Even though the whole backup stack is not operational is the ',PART ,'
out of the backup stack operational?',nl,nl,nl].

legalvals(PART-in-backup-stack is operational) = [yes,no,glossary].

question(a-jump-rig is on-site-Y) = [nl,nl,nl,'
Is a jump rig on site ',Y ,'?',nl,nl,nl].

legalvals(a-jump-rig is on-site-Y) = [yes,no,glossary].

question(jump-stack is operational) = [nl,nl,nl,'
Is a jump stack operational?',nl,nl,nl].

legalvals(jump-stack is operational) = [yes,no,glossary].

question(PART-in-jump-rig is operational) = [nl,nl,nl,'
Even though the stacks in the jump rig are not operational, is the
  ',PART ,' operational?',nl,nl,nl].

legalvals(PART-in-jump-rig is operational) = [yes,no,glossary].

question(a-tactical-jump-of-site-Y is planned) = [nl,nl,nl,'
Is a tactical jump planned for site ',Y ,' that you know of?
',nl,nl,nl].

legalvals(a-tactical-jump-of-site-Y is planned) = [yes,no,glossary].

question(time-to-site-Y-jump) = [nl,nl,nl,'
Approx. how many hours until the site ',Y ,' makes a jump?  Please
include a certainty factor in your answer.
(In the form ''answer cf number'')
',nl,nl,nl].

legalvals(time-to-site-Y-jump) = number.

question(repair-time-for-system-X-on-site-Y) = [nl,nl,nl,'
Approx. how many hours will it take to repair the system ',X ,'  given
that the spare component is available at site ', Y ,'?
Please include a certainty factor in your reply.
(In the form ''answer cf number'')
',nl,nl,nl].

legalvals(repair-time-for-system-X-on-site-Y) = number.

question(spare-PART is on-site-Y) = [nl,nl,nl,'
Is a spare ',PART ,' on site ',Y ,'?',nl,nl,nl].

legalvals(spare-PART is on-site-Y) = [yes,no,glossary].

question(spare-PART is available-at-another-site) = [nl,nl,nl,'
Is a spare ',PART ,' available at another site?',nl,nl,nl].

legalvals(spare-PART is available-at-another-site) = [yes,no,glossary].
```

```
whenfound(spare-PART is available-at-another-site = yes) = [location-
of-spare-PART, time-until-repairs-can-begin].

question(location-of-spare-PART) = [nl,nl,nl,'
You have indicated that a spare ',PART ,' is available at another
site.  Please input the location of the ',PART ,' in single ('')
quotes.',nl,nl,nl].

question(time-to-get-spare-PART-to-site-Y) = [nl,nl,nl,'
Approx. how many hours will it take to get the ',PART ,' from the other
site?  Please include a certainty factor in your reply.
 (In the form ''answer cf factor'')',nl,nl,nl].

legalvals(time-to-get-spare-PART-to-site-Y) = number.

question(the-approval-of-syscon-has-been-obtained) = [nl,nl,nl,'
We are now considering deletion of the system.  Has the SYSCON given
approval to deletion of the system?',nl,nl,nl].

legalvals(the-approval-of-syscon-has-been-obtained) = [yes,no].

question(time-to-replace-system-X-with-another-stack) = [nl,nl,nl,'
About how long will it take to replace the current system with another
stack, fire it up, and start logging in channels?  Please include a
certainty factor. (In the form ''answer cf factor''',nl,nl,nl].

legalvals(time-to-replace-system-X-with-another-stack) = number.

question(time-to-cannibalize-a-stack) = [nl,nl,nl,'
How long would it take to remove the appropriate part from another
stack, put it in the system, and begin logging in channels?  Please
include a certainty factor. (In the form ''answer cf factor'')',nl,nl,nl].

legalvals(time-to-cannibalize-a-stack) = number.

/*----------------------END QUESTIONS---------------------------------*/
```

LIST OF REFERENCES

1. *Department of Defense Dictionary of Military and Associated Terms*, Joint Chiefs of Staff (JCS) Publication 1, Joint Chiefs of Staff, 1 June 1979.

2. Army Institute for Professional Development, *Division Tactical Communications*, Signal Subcourse 282, 1st ed., U.S. Government Printing Office, 1 May 1981.

3. Wanner, E., Steigerwale, R., and Clark, D., *"The Application of Decision Aid Technology to Command and Control,"* Field Circular 101-34, Command and Control on the Airland Battlefield: Selected Readings, pp 498-501, June 1984.

4. Army Institute for Professional Development, *Division Level Communications-Electronics Management System*, Signal Subcourse 082, 8th ed., U.S. Government Printing Office, undated.

5. U.S. Army Signal Center Field Circular 11-50 (Heavy), (FC 11-50 (Heavy), *Combat Communications Within the Division (Heavy)*, U.S. Army Signal Center, December 1984.

6. Army Institute of Professional Development, *Tactical Multichannel Radio Systems*, Signal Subcourse 083, 9th ed., U.S. Government Printing Office, 1983.

7. Ackerman, M., *Black/Gold Communications Concept*, information paper presented to the communications officers of the 1st Cavalry Division, 16 August 1984.

8. Department of the Army, Headquarters, Forces Command, *Permanent Orders 160-10, Modification Table of Organization and Equipment (MTOE) 11035JFC13 FC2086*, 15 October 1985.

9. Army Institute for Professional Development, *Division Signal Orders and Reports*, Signal Subcourse 068, 7th ed., U.S. Government Printing Office, 1983.

10. Department of the Army Regulation 750-1 (AR750-1), *Army Materiel Maintenance Concepts and Policies*, 15 August 1983.

11. Harmon, P. and King, D., *Expert Systems: Artificial Intelligence in Business*, John Wiley & Sons, Inc., 1985.

12. Barr, A. and Feigenbaum, E., *The Handbook of Artificial Intelligence, Volume II*, William Kaufmann, Inc., 1982.

13. Williamson, M., *Artificial Intelligence For Microcomputers: The Guide For Business Decisionmakers*, Brady Communications Company, Inc., 1986.

14. Teknowledge, Inc., *M.1 Instructor Notes*, Teknowledge, Inc., July 1986.

15. Teknowledge, Inc., *M.1 Reference Manual for Software Version 2.0; March 1986*, Teknowledge, Inc., 1985.

16. Kreiger, John, "Communication Planning Assistant (COMPASS)," fact sheet developed for the Directorate of Combat Development, U.S. Army Signal Center, Ft. Gordon, Georgia, undated.

17. Routh, Richard L., *Using a PC-Based Expert System Building Tool to Produce an "Intelligent-Computer-Aided-Instructor for Training Multichannel Radio Systems Troubleshooting"*, paper developed for the Communications-Electronic Leadership Department, U.S. Army Signal Center, Ft. Gordon, Georgia, undated.

18. Pressman, Roger S., *Software Engineering: A Practitioner's Approach*, McGraw-Hill Book Company, 1982.

19. Rook, F. W. and Donnell, M. L., *The Expert System Interface: An Empirical Investigation of Control Strategies and the User's Conceptual Model*, paper submitted to IEEE conference, 1986.

20. Boehm, Barry W., *Software Engineering Economics*, Prentice-Hall, Inc., 1981.

21. Korb, Lawrence J., "Designs For Men And Machines," *Journal of Electronic Defense*, March 1984.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943-5002	2
3.	Superintendent, Code 54JS ATTN: Prof. Carl R. Jones Naval Postgraduate School Monterey, California 93943-5000	1
4.	Superintendent, Code 52IS ATTN: Maj. John B. Isett Naval Postgraduate School Monterey, California 93943-5000	1
5.	Superintendent, Code 55PK ATTN: Prof. G. K. Poock Naval Postgraduate School Monterey, California 93943-5000	1
6.	Superintendent, Code 74 ATTN: Prof. M. G. Sovereign Naval Postgraduate School Monterey, California 93943-5000	1
7.	Superintendent, Code 39 Joint C3 Curricular Officer Naval Postgraduate School Monterey, California 93943-5000	1
8.	Cpt. Joseph S. Yavorsky P.O. Box 8039 Augusta, Georgia 30905	3
9.	Cpt. Richard Routh 2528 Springwood Drive Augusta, Georgia 30904	1
10.	HQ, USA Signal Center and Ft. Gordon Directorate of Combat Development ATTN: ATZH-CD Ft. Gordon, Georgia 30905	1
11.	HQ, USA Signal Center and Ft. Gordon Artificial Intelligence Center of Excellence for Training ATTN: ATZH-SLC-A Ft. Gordon, Georgia 30905-5160	2

Thesis
Y3255
c.1
Yavorsky
Development of a prototype multichannel communications networks maintenance expert system.