NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

DEVELOPMENT OF A KNOWLEDGE BASE FOR USE IN AN EXPERT SYSTEM ADVISOR FOR AIRCRAFT MAINTENANCE SCHEDULING (ESAAMS)

by

Mark H. Stone, Jr.

March 1991

Thesis Advisor: Martin J. McCaffrey

Approved for public release; distribution is unlimited

Unclassified

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. DCLASSIFICATION/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)* 37	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c. ADDRESS *(city, state, and ZIP code)* Monterey, CA 93943-5000	7b. ADDRESS *(city, state, and ZIP code)* Monterey, CA 93943-5000
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 6b. OFFICE SYMBOL *(If Applicable)*	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS *(city, state, and ZIP code)*	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. PROJECT NO. TASK NO. WORK UNIT ACCESSION NO.

11. Title (*INCLUDE SECURITY CLASSIFICATION*)
DEVELOPMENT OF A KNOWLEDGE BASE FOR USE IN AN EXPERT SYSTEM ADVISOR FOR AIRCRAFT MAINTENANCE SCHEDULING (ESAAMS)

12. PERSONAL AUTHOR(S)
Mark H. Stone, Jr.

13a. TYPE OF REPORT Master's Thesis	13b. TIME COVERED FROM TO	14. DATE OF REPORT *(year, month,day)* March 1991	15. PAGE COUNT 100

16. SUPPLEMENTARY NOTATION
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17. COSATI CODES

FIELD	GROUP	SUBGROUP

18. SUBJECT TERMS (*continue on reverse if necessary and identify by block number*)
Aircraft Maintenance, Expert Systems, Knowledge Acquisition

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

The Expert System Advisor for Aircraft Maintenance Scheduling (ESAAMS) was originally proposed to assist in the scheduling of discrepancies in a naval aviation squadron maintenance department. The thesis addresses the development of a knowledge base for ESAAMS which will support the stated goals of the system. An overview of expert systems in general and specifically the ESAAMS system is presented as background information to the reader. A specific approach to acquiring, documenting and storing the knowledge is suggested which will facilitate further development of the system prototype. Based on interviews with experienced maintenance controllers, an initial knowledge base is provided for use in the prototype system. Concluding the thesis are recommendations for further study based upon the findings discovered during this research.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified	
22a. NAME OF RESPONSIBLE INDIVIDUAL Martin J. McCaffrey	22b. TELEPHONE (Include Area Code) (408) 646-2388	22c. OFFICE SYMBOL AS/MF

DD FORM 1473, 84 MAR 83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
Unclassified

Development of a Knowledge Base for Use in an Expert System Advisor for Aircraft Maintenance Scheduling (ESAAMS)

by

Mark H. Stone, Jr.
Lieutenant Commander, United States Navy
B.S.M.E., University of Wisconsin-Madison, 1980

Submitted in partial fulfillment of the requirements
for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL
March 1991

ABSTRACT

The Expert System Advisor for Aircraft Maintenance Scheduling (ESAAMS) was originally proposed to assist in the scheduling of discrepancies in a naval aviation squadron maintenance department. This thesis addresses the development of a knowledge base for ESAAMS which will support the stated goals of the system. An overview of expert systems in general and specifically the ESAAMS system is presented as background information to the reader. A specific approach to acquiring, documenting and storing the knowledge is suggested which will facilitate further development of the system prototype. Based on interviews with experienced maintenance controllers, an initial knowledge base is provided for use in the prototype system. Concluding the thesis are recommendations for further study based upon the findings discovered during this research.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

AFB	Airframe Bulletin
AFC	Airframe Change
AIMD	Aircraft Intermediate Maintenance Department
ASPA	Aircraft Service Period Adjustment
AVC	Avionic Change
AWM	Awaiting Maintenance
AWP	Awaiting Parts
CO	Commanding Officer
CPU	Central Processing Unit
DOD	Department of Defense
ESAAMS	Expert System Advisor for Aircraft Maintenance Scheduling
FMC	Full Mission Capable
IMA	Intermediate Maintenance Activity
IW	In Work
IWSR	Integrated Weapon Systems Review
JCN	Job Control Number
LHS	Left Hand Side
MC	Mission Capable
MDS	Maintenance Data System
MIS	Management Information System
MMCO	Maintenance Material Control Officer
MRC	Maintenance Requirement Card
MTBF	Mean Time Between Failure

NALCOMIS	Naval Aviation Logistics Command MIS
NALDA	Naval Aviation Logistics Data Analysis
NMC	Not Mission Capable
OASIS	Organizational Activity Strategic Information System
OMA	Organizational Maintenance Activity
OOP	Object Oriented Programming
OPC	Optimum Performance Capable
PED	Period End Date
PMC	Partial Mission Capable
PMS	Planned Maintenance System
PPC	Powerplant Change
RHS	Right Hand Side
SPINTAC	Special Interest Aircraft
SRC	Scheduled Removal Component
TD	Technical Directive
TDSA	Technical Directive Status Accounting System
TMS	Type/Model/Series
TSN	Time Since New
VFA	Strike Fighter Squadron
VIDS	Visual Indicating Display System
VIDS/MAF	Visual Indicating Display System/Maintenance Action Form
WUC	Work Unit Code

ACKNOWLEDGEMENT

Sincere thanks are given to my thesis advisor, Professor Martin J. McCaffrey for the help and insight he provided throughout the thesis development. His enthusiasm for the subject matter of this thesis was infectious and will undoubtedly result in the successful deployment of ESAAMS in the not too distant future. Grateful appreciation is also extended to the many professional maintenance managers, particularly the maintenance staff of VFA-147 who graciously conducted the inital knowledge acquisition session in spite of the pressure of high tempo operations.

Special gratitude is conferred upon my parents who instilled a love of learning and a desire to serve in the armed forces; two factors which I have been able to combine while assigned to the Naval Postgraduate School. Their support and encouragement throughout this effort have been appreciated.

Finally, I thank my best friend and wonderful wife, Julie. Without her editing, encouragement and devoted love, I could not have accomplished this work. I owe her a measureless debt for tolerating my absence late at night and on some of those beautiful California weekends.

I. INTRODUCTION

A. BACKGROUND

The most often cited stumbling block in expert system development and utilization has been the inability of knowledge engineers to successfully capture and represent the knowledge used by experts in the decision making process. The inability to extract and translate expert knowledge into rules is most likely the primary reason that a significant portion of current expert system implementations deal primarily with small knowledge bases of less than one hundred rules (McGraw & Harbison-Briggs, 1989, p. xiii). Developing an expert system advisor for aircraft maintenance scheduling is a complex task which will require at the least several hundred rules and will encompass the knowledge of many different experts in the maintenance, operational, and logistical environments. This increased complexity will require that knowledge acquisition be conducted in a structured procedural manner in order to ensure that decision rules are soundly considered and to facilitate thorough validation and verification of the knowledge base. The concept of using expert system technology in the aircraft maintenance environment is based on previously published research which discussed the feasibility of developing an Expert System Advisor for Aircraft Maintenance Scheduling (ESAAMS). (McCaffrey, 1985)

B. OBJECTIVES

This thesis discusses the plan for the knowledge acquisition phase of the ESAAMS system development. It will discuss all phases of the knowledge acquisition plan from the administrative preparations through the validation and verification of the knowledge base. It can be thought of as a practical handbook for the knowledge engineering team and is intended as a down-to-earth guide rather than as a theoretical discussion of the knowledge acquisition paradigm.

Knowledge acquisition is the process through which knowledge engineers capture that knowledge which domain experts use to perform the task at hand. This knowledge is analyzed and then codified in a structured format as an expert system application.

The following research questions will be addressed:

- What knowledge must be included in an expert system advisor for aircraft maintenance scheduling?
- What are the possible sources of the required knowledge?
- What knowledge is too subjective to include?
- What makes an expert, an expert?
- How is growth of the knowledge base controlled?
- How is the validity/quality of the knowledge determined?
- How valid is the knowledge included in the knowledge base?
- How should the knowledge base be documented?

C. METHODOLOGY

Preliminary discussions were held in August and September of 1990 with representatives of VFA-147, in which the various factors upon which domain experts base their maintenance decision making were discussed. Due to unscheduled operational commitments follow-on interviews could not be scheduled with the squadron. Instead, several aircraft maintenance officers assigned to the Naval Postgraduate School readily volunteered to provide their expertise to the knowledge acquisition effort.

D. THESIS ORGANIZATION

It is not intended that this thesis provide a comprehensive description of the expert system development process, rather it is intended to focus solely on the knowledge acquisition phase of a development project. Never the less, it is important that the reader be familiar with the basic features of an expert system in order to understand the purpose of the knowledge acquisition phase. Accordingly, Chapter II will describe the basic components of an expert system; the knowledge base, inference engine and user interface. It will also expose the reader to the expert system development life cycle. Chapter III will provide specific recommendations for acquiring the substantial knowledge base which will be required for the successful resolution of the aircraft maintenance scheduling problem. It will provide an outline of the knowledge acquisition process and define the types of knowledge which will be required. Choosing the domain experts and working with those experts will be discussed. Finally, interviewing techniques and methodologies will be presented.

The application area, aircraft maintenance scheduling, will be discussed in Chapter IV. The factors that domain experts must consider and the underlying policies of United States Navy aircraft squadron organizational maintenance departments will be introduced.

Chapter V will provide a brief discussion of knowledge representation schemes and suggest a potential architecture for the fully developed expert system. The contents and evaluation of the knowledge base will be the topic of Chapter VI. Further, a review of the verification and validation of the knowledge base will be offered. Chapter VII will conclude the thesis and provide recommendations for further research in the topic area.

II. AN INTRODUCTION TO EXPERT SYSTEMS

In order to fully appreciate the knowledge acquisition task it is essential that the basic architecture of expert systems in general be understood. As such, this chapter provides the reader with a brief overview of expert systems. The components which make up knowledge based systems and how those components interact is discussed. This chapter will conclude with an overview of the proposed Expert System Advisor for Aircraft Maintenance Scheduling as envisioned by McCaffrey (1985).

A. INTRODUCTION

With the advent of increased capabilities and decreased costs in digital computers, there has become an increased sophistication in their use. The computers built during the last three decades were huge machines which cost millions of dollars. Today, those large computer systems are being replaced by smaller, less costly computers which have the same capabilities as their "big brothers". The ultimate design and subsequent use of these newer computers has branched into two distinct areas.

One area is the continued progression toward faster and faster processing machines. These machines can quickly and accurately calculate large numbers, plot complicated graphs, and even understand the human voice. The trend of these computer systems is toward increasing the ease of man-machine communication which will tend to decrease the special training requirements for humans to interact with the computer.

The second area is the increasing sophistication of computers used in decision-making processes. These machines use complicated algorithms to correlate and disseminate information. The judgement and decision-making capabilities of these computers were formerly attained only by "intelligent" humans. Because of their reasoning capability, these computers have fallen into the field of "artificial intelligence".

Since computers are not endowed with any knowledge on their own, they must be provided with information from a human. Computers are currently being used for diagnostic applications in fields such as medicine and mineral explorations (Feigenbaum, 1988, pp. 166-168). These computers are supplied with a large amount of the knowledge of a human "expert" in a specific field of endeavor. These computers are then used to augment the human intellect of the "less than expert" individual in the diagnosis of a specific problem of that field.

A computer used in this manner is called an "Expert System" or "Knowledge-Based System". The domain of factual knowledge possessed by an expert system is real; however, the knowledge is artificially generated. Limited to a specific problem domain, this knowledge can be accessed much faster and with greater accuracy than the same knowledge can be obtained from the human expert. For these reasons, the realm of artificial intelligence and expert systems is of significant interest to the Department of Defense (Ferguson, 1983, p. I-4).

Within the last few years research in the field of artificial intelligence has grown significantly and expert systems have been successfully deployed in the manufacturing, service sector, as well as within the military. Development of artificial intelligence type systems for equipment maintenance in the commercial and industrial environments is currently underway at American Airlines and

Grumman Aerospace. A project similar to ESAAMS but intended for different types of equipment was developed for TELECOM, Incorporated (Follett, 1987, p120). Using a consultative expert system many of the same factors such a preventative maintenance schedules, policy influences and inventory management were codified. Additionally, and significantly, this system queried a database as to the maintenance history of equipment in planning and scheduling its repair or disposistion. Although it is unlikely that the Navy would allow an expert system to specify or even suggest the non-repairability of an aircraft, the TELECOM system does posess many of the features specified for inclusion in the ESAAMS project.

B. COMPONENTS

The main difference between an expert system and a traditional application is that in an expert system, the model of problem solving in the application domain is explicitly in view as a separate entity or knowledge base rather than appearing only implicitly as part of the coding of the program.

Expert systems are composed of at least three basic entities: the knowledge base, an inference engine, and a user interface. The knowledge base contains rules expressing an experts heuristics for the domain. The inference engine is made up of rules that are used to control how the rules in the knowledge base are processed. The user interface allows communication or interaction between the expert system and an end user.

1. Knowledge Base

The knowledge base houses the information used by the expert system in pursuit of a solution to a problem. It is a step above a conventional database in that a knowledge base not only contains static data, but also contains relational

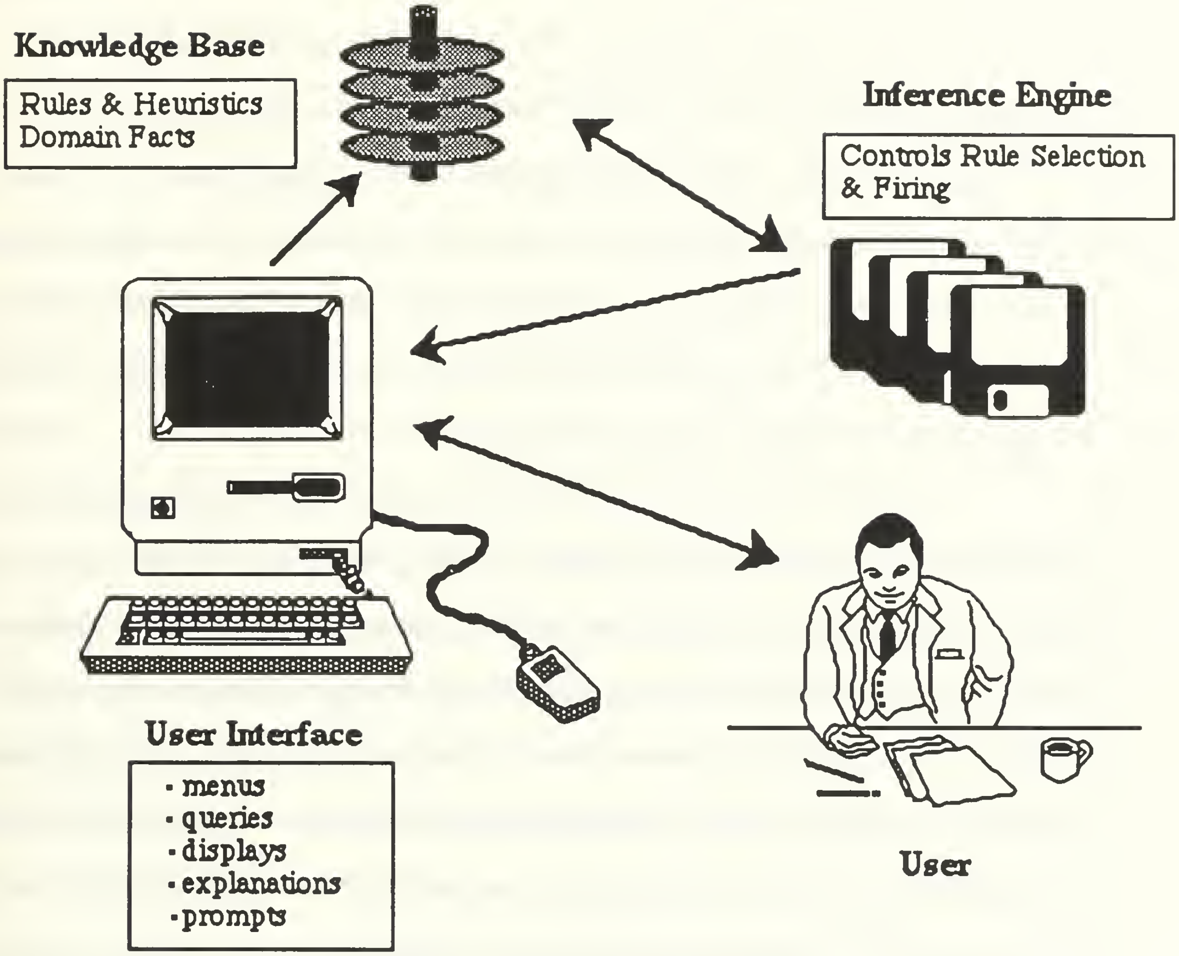

Figure 2-1. Expert System Components

information. A third area of the knowledge base is working memory. Working memory is used only during processing and is the resident space for information manipulation.

a. *The Database*

The database includes the facts of the problem, both related and unrelated. This is a passive area of the expert system--simply a storage space for data and formulas. The information included encompasses the given and unchanging knowledge about the problem and domain. This database may be updated on a real time basis through the user interface of the expert system, or it may be periodically updated from data stored in a separate database, such as the Naval Aviation Logistics Data Analysis (NALDA) database.

For example, within ESAAMS this database would contain data of a historical nature about the specific aircraft within the squadron as well as data in general about the aircraft type, model and series (TMS). Elapsed maintenance times for a given maintenance action, or information which would point out a problem of a recurring nature in a particular aircraft or series (block) of aircraft. There should also be a database which holds current information about the aircraft (status, location), support equipment (status, availiablility) and parts (status, estimated delivery date). These could either be a part of the expert system or separate databases able to be queried by the expert system on a demand basis.

b. *The Knowledge Base*

The knowledge base contains known facts about the subject, expressed as objects, attributes and conditions. It can be distinquished from the data base by its symbolic, rather than numeric content and by the fact that a relationship between the facts is not assumed. Each "chunk" of information is essentially independent. Production rules, the basis of most expert systems, are located here. This is the most difficult portion of the system to develop and implement.

c. The Working Memory

Here the knowledge base is modified by the inference engine as situations and data change--a much more interactive area of the expert system than the database. Working memory takes data from the database, knowledge from the knowledge base, and combines them with the information supplied from the user to then be massaged by the inference engine in pursuit of a solution.

2. Inference Engine

The inference engine is the mechanism which provides the central control for the expert system. Its primary effort is toward reasoning and making inferences based upon the application of rules contained in the knowledge base. This inference process can be broken down into two parts. The first involves the selection of the context structure for the problem, and the second relates to the manner in which the reasoning mechanism should process those contexts. There are two basic control strategies implemented in current expert systems. The implementation of a selected strategy is based upon the type of expert system, either diagnostic or pedagogic, and the specific domain of application.

a. Forward Chaining

One of the simplest structures is known as forward chaining or data-driven searching. This method starts with the initial given conditions and searches forward through the knowledge base towards a solution. Also known as bottom-up processing or antecedent reasoning it is best used in "what-if" scenarios. The system begins with a fact and proceeds to search for a rule whose premise is verified by that fact. The conclusion is then added to working memory in pursuit of the solution.

b. Backward Chaining

A second strategy and the opposite of forward chaining, is backward chaining. This strategy is a goal-directed search that starts at the end solution (goal state) and works backward towards the initial conditions. This is also known as top-down processing or consequent reasoning. The task is to see whether the necessary and sufficient antecedents that satisfy the goal exists in the domain by applying inverse operations. The process begins with a goal-state hypothesis. Next the system seeks to locate a rule whose premise supports the hypothesis and then attempts to verify the premise by searching the knowledge base for a relevant fact. If no fact is found, the system searches for a rule that can be used to infer the fact. This process of searching and verifying the supporting facts continues until the original hypothesis is verified or disproved (Walters, 1988, pp. 202-203).

c. Search Strategies

The effectiveness of an inference procedure is also dependent on the method in which the hierarchical structure is scrutinized. There are three methods in which this is done:

- Breadth first search
- Depth first search
- Best-first search

The breadth first search examines all nodes in order of their distance from the start node. All those nodes immediately adjacent to the start node will be considered before it goes to the next depth in the hierarchy. Although the breadth first search may be an extremely long process, by its nature, it will find the shortest possible solution sequence.

The depth first search selects one path and follows that path downward until it reaches a node that has no successors. Which path is selected first may be determined randomly or through an algorithm that selects the most promising path. After reaching the bottom node, the system must determine whether or not the node contains an acceptable solution. If it is not acceptable, then a backtrack is initiated to the next higher node that has other paths to search. An advantage of the depth first process is that it reaches potential solutions directly, and by monitoring the solutions as they are determined, the process can be terminated as soon as an acceptable solution can be derived. Without good predictive functions however, the system has the potential for spending considerable time working on paths that are not promising in the search for good answers.

The best-first approach is one that always selects the most promising node as the next node to expand. A combination of depth first and breadth first techniques, the best first search uses an evaluation function at every node to determine the promise of following a certain path. The evaluation function (f^*) is defined so that the more promising a node is, the smaller is the value of f^*. The node selected for expansion is the one at which f^* is minimum. The basic algorithm for this search was developed by Nilsson (1971) and reviewing it makes the methodology much clearer.

1. Put the start node s on a list, called OPEN, of unexpanded nodes. Calculate f*(s) and associate its value with node s.
2. If OPEN is empty, exit with failure; no solution exists.
3. Select from OPEN a node i at which f* is minimum. If several nodes qualify, choose a goal node if there is one, and otherwise choose among them arbitrarily.
4. Remove node i from OPEN and place it on a list, called CLOSED, of expanded nodes.
5. If i is a goal node, exit with success; a solution has been found.
6. Expand node i, creating nodes for all of its successors. For each and every successor node j of i:
7. Calculate f*(j)
8. If j is neither in list OPEN nor in list CLOSED, then add it to OPEN, with its f* value. Attach a pointer from j back to its predecessor i (in order to trace back a solution path once a goal node is found).
9. If j was already on either OPEN or CLOSED, compare the f* value just calculated for j with the value previously associated with the node. If the new value is lower, then
10. Substitute it for the old value.
11. Point j back to i instead of to its previously found predecessor.
12. If node j was on the CLOSED list, move it back to OPEN.
13. Go to (2)

In practice the implementation of this algorithm is not an easy task. The degree of success one will have in its use is totally dependent on the legitimacy of f*. If f* is not accurate, promising solutions are likely to be overlooked.

The inference engine is the workhorse of the expert system. It contains the processes that work the knowledge base, do analyses, form hypotheses, and audit the processes according to some strategy that emulates the expert's reasoning. The inference engine massages new information, combines it with the knowledge base, considers the relationships in the knowledge base, and proceeds to solve the problem in working memory using its established reasoning and search strategies. In other words, the inference engine is the "thinker" of a problem-solving system; it provides overall control.

3. User Interface

The user interface is often considered the preeminent measure of expert system performance, in that no matter how efficient its inference engine or extensive its knowledge base, the program is only as valuable as its ability to communicate lucidly with those who require access to its output (Sawyer, 1986, p. 49). The job of the user interface is to exchange information between the operator and the inference engine. A natural language interface simulates casual conversation, using everyday expressions in plain English.

The user has the ability to control the strategy he wishes the inference engine to pursue. He may add facts to the knowledge base or modify existing facts. If the inference strategy appears to be leading to an unacceptable solution, that path can be terminated and an alternative branch can be explored. The system may require input from the user at certain times during the session and may or may not provide default answers. Essentially the user interface exists to allow the operator to modify or tailor the direction in which the inference engine is working.

C. EXPERT SYSTEM DEVELOPMENT

Expert system development can be broken down into three major phases. Although no two projects are exactly alike, a reasonable plan will consist of three development phases as discussed below (Prerau, p. 30, 1990).

1. Initial Phases

The initial phase consists of project start up, domain selection and selection of the development environment. In the project at hand, McCaffrey(1985) essentially handled the project start up and domain selection. The development environment selected for the prototype system is NEXPERT Object®, an expert system shell developed by Neuron Data, Inc.

2. Core Development Phases

There are two core development phases. The first one is the development of a feasibility prototype system. This is a rapid prototype expert system that implements a subset of the problem being tackled by the complete system. When completed, a feasibility prototype system should, as the name implies, give evidence of the feasibility of using expert system technology for the application. The purposes of this early prototype can be any or all of the following: (Prerau, p. 30, 1990)

- It allows the project developers to get a good idea of whether it is feasible to attempt to tackle the full application using expert system technology.
- It provides a vehicle through which to study the effectiveness of the knowledge representation.
- It provides a vehicle through which to study the effectiveness of the knowledge implementation.
- It may disclose important gaps or important problems in the proposed final system.
- It yields a tangible product of the project at an early stage.
- It gives an opportunity to impress management or system sponsors with a flashy system demonstration, helping to retain or increase support of the project.
- It gives an idea of what the final system will do and will look like to outside experts and potential users.
- It allows the possibility of an early mid-course correction of the project direction based on feedback from management, consulting experts, and potential users.
- It provides a first system that can be field-tested--yielding experience in using and testing the system and, if the tests are successful, credibility that the eventual final system will perform its desired function well.

- It might provide a system with enough utility that, although it is not a final product, it may be put in the field on an extended basis. This early deployment of a limited system yields some domain benefits, gives experience to system deployers, system operators, and system maintainers, and might identify potential problems in those areas.

After testing and validation of the prototype the project team evaluates its performance to determine its suitability for further development. Should the final prototype prove desireable, the project moves into the last phase of its lifecyle.

3. Final Development and Deployment Phase

Should a project make it to the final phase, and more of them do every year now, the final production system is developed and deployed. As in a conventional software project, it then begins the maintenance phase which will last the lifetime of the system. New features are added, defects corrected and performance improvments are incorporated where possible.

D. EXPERT SYSTEM FOR AIRCRAFT MAINTENANCE SCHEDULING

Due to the sophistication and rapid technological advances of today's DOD weapon systems, there is an ever increasing need for highly qualified managers to supervise their maintenance. The incorporation of advanced technology, in both new and existing weapons systems, has made the accurate and timely assessment of damaged or malfunctioning equipment and the scheduling of its repair an extremely complex task. As the complexity of these systems increases, there will inevitably be fewer and fewer so called "technical experts" assigned to maintenance control.

The primary goal within the organizational maintenance activity (OMA) is to provide fully mission capable aircraft to support the operational flight schedule.

The maintenance department must strike a balance between the seemingly contradictory sub-goals of providing the maximum number of operationally ready aircraft and maintaining those same aircraft in top material condition. The maintenance/material control officer (MMCO) is the person within the OMA who must make optimum use of the available resources, both manpower and material, in developing the daily maintenance schedule.

Maintenance schedulers, even the experts, are normally aided with their assessment of a system through the use of technical publications, manuals and instructions. However, these manuals are bulky, difficult to understand, and usually not updated with the current information pertaining to the system. Therefore, it is evident that some method must be found that will provide current information on a weapon system, will be easy to use, and will provide a quick and accurate assessment of the particular weapon system problem.

McCaffrey (1985), studied the feasibility of implementing expert system technology at the organizational level of maintenance and concluded:

> ...it is submitted that development of an expert system for scheduling discrepancies is both feasible and appropriate. It should be emphasized that such a system would serve as a decision support tool and not as a replacement tool for the MCC/MCO's decision making for this domain. The improved management effectiveness and potential for improved aircraft operational readiness that an expert system offers are well worth the costs.

At the time it was written McCaffrey intended that his proposed system be tied into the Naval Aviation Logistics Command Management Information System and make use of the many planned features the system was incorporated with. Since that time, NALCOMIS implementation at the organizational level has been scaled back and reduced in scope to a significant degree. It is today deployed at several activities at the IMA and supply levels and its future as a comprehensive management information system (MIS) at the orgaanizational level remains in doubt. So, although the expert system concept is still viable, its incorporation will involve significantly more work than originally envisioned.

E. SUMMARY

An expert system is a special purpose computer program that solves problems by employing the technical knowledge, information, heuristics and problem solving processes that human experts use to solve such problems. The system consists of a knowledge base, inference engine and a user interface. Expert systems can best be differentiated from traditional management information systems (MIS) through their reliance on knowledge. Unlike traditional MIS's, they have the capability to develop solutions even when input data is incomplete or inconsistent. Significantly they also have the ability to explain how they arrived at a particular decision or why they are asking for certain information during a reasoning process.

Expert system development is not a fully developed, mature topic hence there are many thoughts as to how the development should occur. The clearest model encountered in research consists of three phases. The initial phase involves selecting a project and choosing a development environment. Secondly, the core development phase involves developing initial and full prototypes of the system. Lastly the final phase is development and deployment of a finished product and the maintenance of that product once it has been installed.

ESAAMS, is a system designed to assist maintenance managers within a naval aircraft squadron in planning and scheduling the daily maintenance workload. An evaluation of the feasibility of this project determined its suitability for development and the purpose of this thesis is to explore the knowledge acquisition phase of the development effort.

III. KNOWLEDGE ACQUISITION

Knowledge acquisition is the process by which expert system developers find the knowledge that domain experts use to perform the task of interest. This knowledge is then codified to form the expert system program. The essential part of an expert system is its knowledge, indeed that is what differentiates an expert system from a conventional software product. Next to actually selecting a domain, knowledge acquisition is generally regarded as the most difficult facet of an expert system development project.

Acquiring knowledge from a domain expert is not an easily accomplished task. Generally an expert does not fully realize all that goes into the decisions which they make. A quick, seemingly snap decision often encompasses a large amount of information and judgements. Furthermore, expert's actions are sometimes performed almost unconsciously, based on years of successful performance. A good example of this phenomena is the following scenario (Prerau, 1990, p.200]:

> ...I have asked experienced drivers the following question: "Approximately how often do you look into the rear view mirror when driving on a highway in normal conditions: every ten seconds? every 30 seconds? every 5 minutes?" They almost always have no idea how often they do this task, but they know they do it, and their years of good performance indicate that they do it at a reasonably expert level. This illustrates another problem for knowledge acquisition: getting expertise from experts who do not have a firm notion of exactly how they do their tasks.

This chapter discusses the knowledge acquisition phase in the development of the ESAAMS project. The first section discusses the task of familiarizing a potential knowledge engineer with the domain to be captured. A course of study involving both classroom, laboratory and real time experience would provide the knowledge engineer with sufficient background to begin the project development. Next the role of the domain expert is defined and recommendations on choosing a domain expert are identified. Following that is a discussion of common knowledge acquisition techniques which can be used in project development, however, it is probable that only a few of them will will be utilized in the development of ESAAMS.

A. THE KNOWLEDGE ACQUISITION PROGRAM

During the very early stages in an expert system development project, it is important that the knowledge engineer or engineers become familiar with the domain to be addressed. They will work with the project manager to plan the domain familiarization training, establish a properly equipped facility, develop knowledge acquisition procedures and develop a plan to orient the domain experts with expert system technology. Prior to the execution of this phase a feasibility study for the entire project should have been completed. Such a study conducted by McCaffrey (1985) confirmed the feasibility of the Expert System Advisor for Aircraft Maintenance Scheduling (ESAAMS) and forms the basis for the knowledge acquisition phase under discussion in this thesis.

1. Domain Familiarization

Until one is exposed to the vocabulary of maintenance control, the high tempo of operations and the decision making influences faced by the maintenance controller, he will have little appreciation for the expertise required. Simply placing the knowledge engineer in an operating maintenance control work center for familiarization would be fruitless. He would not comprehend the terminology, physical layout, or labyrinth of supporting ship and air station services that are available. As a sound remedy for this lack of background knowledge, the primary knowledge engineer for the project would benefit from attending the Aircraft Maintenance Officer course held several times during the year at the Naval Air Station in Pensacola, Florida. A basic familiarization with the terminology and general principles which underlie the maintenance process would result. With the same background knowledge as a novice maintenance officer he would be significantly better equipped to understand the dynamics involved in an operating maintenance control. With his classroom training complete he should be assigned to an operating squadron for a minimum of two months in order to get an appreciation for the effect that high tempo operations place on a decision maker in the maintenance control domain. As a less attractive alternative, talented domain experts could be trained in the various knowledge elicitation techniques and function as knowledge engineers in their respective areas of expertise. This is decidedly the poorer of the two alternatives.

2. The Domain Expert

In order to select appropriate domain experts, it is important to identify the experience, characteristics, and attributes that will facilitate knowledge base development goals. Identification of requirements for domain experts is only the first step. Few of the selected experts will be knowledgeable concerning expert system development in general and knowledge acquisition specifically. The importance of their role in knowledge base development requires that they become an integral part of the team. This necessitates that their interactions with the developer be characterized by professionalism and good rapport. Effective working relationships between knowledge engineers and domain experts are characterized by : (1) openness, (2) respect, and (3) interdependence. Openness describes the degree of honest or directness each party can use with the other and is important to the knowledge engineer's ability to secure valid information from the domain expert. Mutual respect refers to each participant's ability to feel valued by the other. While this does not imply that they must like each other, it does imply that each should recognize the other's professionalism and abilities. Interdependence is important in this working relationship as the knowledge engineer and domain expert must work together to meet session goals. Each must be an active participant .

The development of relationships that embody these and related characteristics requires work that begins with the initial selection of domain experts who will contribute to the knowledge base development efforts.

a. Choosing a Domain Expert

A system with the scope of ESAAMS clearly cannot be developed without the substantial input of domain experts from across the spectrum of aircraft maintenance and squadron operation’s policy. A single expert may be able to provide all the expertise required for a single phase or even several phases, but will likely fall short in at least one of the domains to be explored. Given the broad scope of this project it is important to include as many experts as feasible while at the same time excluding those who have little to add or offer to the knowledge acquisition process.

Credibility of the expert is an often overlooked concern. The expert must be credible to

- The user community who will ultimately determine the initial acceptance and subsequent success of the expert system.
- The system project team, which will need to work closely with the expert over a period of time; the initial expert will often become a "knowledge czar" since his knowledge and reasoning processes will provide the framework for the complete system.
- The "expert" community; since other experts will often be called upon to refine the initial system, or become the source of expertise for other sub-domains, the expert's credibility in the eyes of the professional "fraternity" is crucial to gaining future cooperation.
- The organization's management, who provides initial system development resources and the inevitable follow-up financing, and who will ultimately determine the level of organizational integration.

Within the aviation maintenance community it is more difficult than one may expect to find an expert suitable for the project. Many of those we may at first consider as our domain experts are senior enlisted maintenance chief petty officers. They have a significant amount of time in the Navy and have spent a large portion of their careers as maintenance control supervisors. Based on inspection results and readiness figures it is easy to select the best. Functional wing staffs will readily identify those maintenance chiefs who qualify from a technical point of view.

The difficulty will arise in gaining their cooperation in the development effort. The Navy has to date not produced a credible MIS for use by maintenance controllers, in fact the Naval Aviation Logistics Command Management Information System (NALCOMIS) is currently about ten years behind schedule in its deployment. As supervisors, maintenance control cheifs have been tasked with validating hundreds of pages of computer print outs every week with no tangible benefit gained. There is a basic distrust, not of computers in general, but in how information processing technologies have been implemented and their value at the squadron level to date. Further, based on a lack of understanding of what expert systems can do, it is likely they will be doubtful that ESAAMS will be of significant help or that it will provide any desirable benefits. A perception will exist that “it can’t work.”

b. Problems with the Expert

Regardless of a knowledge engineer's abilities, the interpersonal nature of the knowledge acquisition session, coupled with the difficulty of the task ensures that problems will arise. Even if supportive at first, the following difficulties are likely to evidence themselves at sometime during the development effort:

- Negativism and apathy.
- Lack of commitment.
- Verbal and nonverbal communication blocks.
- Hostility and defensive reactions.
- Clashes between expectations and realities.

Based on discussions with several maintenance chiefs, it appears that initial development will be critical to the success of the system. An incremental approach, starting small and with an area that is particularly difficult to manage appears to be the optimum path to take. By demonstrating successful expert system performance on a small facet of the project, a cadre of supporters may emerge. The success of many software development efforts, both conventional and knowledge based have been assured due to the the enthusiasm and dedication of these “champions”.

c. *Using Multiple Experts*

Given the broad scope of knowledge required to develop a system such as ESAAMS, the use of multiple experts is a foregone conclusion. Thus the already difficult knowledge acquisition process translates into a much more involved procedure. "If knowledge acquisition for an expert system with a single expert can be described as a bottleneck, acquisition from multiple experts, especially in a group setting, has the potential to become a 'log jam.'"(McGraw & Seale, 1987, p. 166) When utilizing multiple experts, among many other items, knowledge engineers must decide how to mediate diverse opinions to develop a coherent expertise.

Decision makers seldom rely on the expertise of a single individual, so it follows that they prefer to rely on multiple experts for the knowledge required in the expert system. The increased knowledge gained from multiple experts will result in a more flexible system, able to demonstrate the use of multiple lines of reasoning. The knowledge engineer will also enjoy more flexibility in acquiring knowledge. If one expert is busy, he can interview one of the other experts in the organization. His productivity will not depend on the availability of the single expert, who may be too busy to devote a large portion of time to the project anyway.

The benefits achieved from using multiple experts do not come without a cost. In reviews of video taped multiple expert systems, it is common to find a junior domain expert making eye contact with senior domain expert as he is interviewed in an attempt to elicit a non-verbal confirmation of his expertise. (McGraw & Briggs, 1989, p. 250) Similarly, a domain expert may be hesitant to provide expertise because of a fear of repercussions from supervisors in a phenomena known as "upward ripple paranoia." (McGraw & Seale, 1987, pp. 165-197) The diversity of opinion cited as a benefit above may also be viewed as a cost. With multiple experts providing multiple opinions, conflict may quickly arise. The knowledge engineer must assert his authority as a moderator in these cases, and move the group towards a consensus position.

3. Reference Library

Recognizing that personnel gains and transfers often occur during development of large expert systems, two additional steps should be taken prior to the project commencement. First, a reference center should be established which will function as a research and reference library for any personnel associated with the project. It will provide background information on the domain and eventually, complete records of all knowledge acquired during the project. Additionally it should be stocked with a comprehensive collection of the various instructions and policies established by the Department of the Navy, type commander, functional wing, air station and ship instructions. These documents essentially govern the operation of aircraft maintenance squadrons and having a current collection on hand will substantially ease the task of validating knowledge further on in course of the project.

Secondly a knowledge dictionary should be established and maintained from the beginning of the project. Analogous to a data dictionary in a conventional software development effort, it would provide a compilation of the domain's terminology and basic concepts. In a large development project this document undergoes frequent, even daily changes so it is advisable to maintain it electronically rather than in hard copy. Any off the shelf data base will function adequately for this task. The primary benefit in maintaining the knowledge dictionary electronically is that it would enable individual knowledge engineers to update and revise the system on a real time basis.

4. Knowledge Acquisition Facilities and Equipment

As was discovered during the initial knowledge acquisition session for this project, the environment under which the knowledge is acquired will impact the development effort. The initial knowledge acquisition session was held within the maintenance control work center of Strike Fighter Squadron 147 (VFA-147). The squadron was in the late stages of work up for a major deployment and the tempo of operations was heavy. The distractions were nearly continuous and despite the willingness of the domain expert to spend time with the knowledge acquisition team little was accomplished. Frequent interruptions were the norm and it was difficult for the expert and the knowledge engineering team to maintain a train of thought. Due to the early deployment of VFA-147, subsequent interviews were held away from the operational environment and the knowledge acquisition process was deemed much more productive. Unfortunately the domain experts were no longer part of the operational environment rather they were graduate student officers with prior experience in maintenance control whose level of expertise could neither be proven or disproven.

Although more knowledge was discovered, in this case away from the squadron work center, one should not draw the conclusion that there is nothing to be gained by observing the domain expert in his natural working environment. Indeed in later stages of development, the knowledge engineering team should expect to gather knowledge in maintenance control where the domain expert can simulate his decision making processes under real time pressures. The optimum environment for a development project is indeed a combination of the two. An office set up away from the actual squadron work center and equipped as a typical maintenance control is equipped would provide the benefit of enabling the maintenance controller to act out his daily routines while avoiding the interruptions expected in a functioning squadron. Among the items of equipment which would benefit the knowledge acquisition environment would be audio and video recording equipment. Enabling accurate transcription of knowledge into rules, the recordings would also serve as a training aid for use in improving the knowledge acquisition capabilities of the development team.

5. The Knowledge Acquisition Session

Both to maintain effective knowledge engineer-domain expert relationships and to elicit quality information from a knowledge acquisition session, it is critically important to manage the session. The knowledge engineer must strictly control the conduct of the session while at the same time function as an effective facilitator and listener. The following objectives provide guidelines for the management of knowledge acquisition sessions to increase the effectiveness of the session and enhance the domain expert-knowledge engineer relationship:

- Establish active leadership upon greeting the domain expert.
- Control the introduction of the knowledge acquisition session and establish its purpose.
- Guide the expert through the knowledge acquisition session, following the agenda as closely as possible.
- Focus the expert on the appropriate levels and points.
- Actively summarize the knowledge acquisition session and debrief the expert at the close of the session.

As the knowledge engineer manages the progress of a knowledge acquisition session, he must also act as a facilitator. The knowledge engineer uses nonverbal and verbal behaviors to act in ways that enable session goals to be attained. Auger (Bowerman, 1988, p. 353) recommends the following tips that a facilitator can use to coax a knowledge acquisition session along:

- Stimulate discussion.
- Balance the discussion if there is more than one expert so that more than one view is addressed.
- Keep discussions on track.
- Break up stumbling blocks or controversies.
- Watch the time table and end sessions on time.
- Make sure there is some conclusion and positive actions.

B. KNOWLEDGE ACQUISITION PROCEDURES

In small, simple expert system development efforts organization of the knowledge acquisition effort need not be very complex. However, in setting up a large scale expert system development project, a need exists for a more intensive project management effort and the need for knowledge traceability becomes much more acute. To set up a successful, manageable knowledge acquisition program for a large expert system development project, the following tasks should be undertaken (McGraw, 1989, p. 70):

- Participant roles and knowledge acquisition techniques should be specified.
- Knowledge acquisition forms and guidelines for use by numerous individuals must be developed.
- Procedures for tracking knowledge from source to code must be developed.

1. Recording Knowledge

The knowledge acquisition form documents the purpose and results of the knowledge acquisition session. The form shown in Figure 3-1, is initially used to set goals for the session and to inform the domain expert as to the topics to be discussed. After the session is complete and the form is completed, it becomes a permanent part of the knowledge acquisition database.

2. Translating Knowledge to Code

Although the focus of this thesis is on knowledge acquisition, it is beneficial to think about how the acquired knowledge will be codes or represented in the expert system. The knowledge engineer can substantially ease the job of encoding the rules by attempting to encode the rules during the acquisition process whenever possible. Prerau (Bowerman, 1990, p. 30) suggests several guidelines based on his experiences that include the following:

- Use English-style "pseudocode" IF-THEN rules to record domain expert knowledge during knowledge acquisition sessions whenever possible.
- Agree upon conventions (e.g., indentation, capitalization, explanations, justifications) for recording rules from knowledge acquisition sessions.
- Use terminology within rules that is consistent with that used in the knowledge dictionary.
- Name rules rather than numbering them whenever possible for the increased specificity this allows and because of the number of changes the knowledge base will go through.
- Include explanations for the rule, a summary of the rule, and a justification of the rule within its documentation.
- Note any certainty factors or factors that impact the rule's validity.
- Document the source and knowledge acquisition session from which the rule was acquired.
- If possible, run through the prototype as soon as is feasible to determine other rules that a specific rule uses and rules that use it.

Expert System Advisor for Aircraft Maintenance Scheduling
Knowledge Acquisition Form

Session #:________ Session

Date:____________

Session Topic:___________

Knowledge Engineer: Domain

Expert:__________

Session Location:_____________ Elapsed

Time:___________

Session Type:_________________

Major Session Goals:

Session Summary:

Rules Derived from Session:

Figure 3-1. Knowledge Acquisition Form

Even though this technique may assist the knowledge engineer in the acquisition process, he should be wary of restricting himself to any particular representation paradigm during the early stages of knowledge acquistion. There may be other techniques which will function more suitably as representation scheme as discussed in Chapter V.

C. KNOWLEDGE ACQUISITION TECHNIQUES

Given a system as large in scope as ESAAMS it is not difficult to establish the fact that knowledge will be acquired in a number of different ways depending on the specific domain being addressed. The field of all possible knowledge acquisition methodologies is vast and it includes techniques borrowed from the field of communications, psychology and education (McGraw, 1989, p.72). While interviewing is generally regarded as the most prevalent method, knowledge is acquired for today's expert systems using many techniques, among them are these five differing methodologies: interviews, protocols, walk throughs, questionnaires, and expert reports (Wolfgram, 1987, p.171).

1. Interviews

Interviewing is the most common technique used by knowledge engineers to elicit domain knowledge from an expert. This technique allows the knowledge engineer to quickly grasp important domain concepts and vocabulary. Interviews are most beneficial and most frequently used in the early stages of knowledge acquisition. Interviewing can be conducted on either a structured or unstructured basis. The unstructured interview is most helpful when the engineer is eliciting general information about a certain topic in the early stages of a its consideration, in order to familiarize himself with the domain. On the other hand a structured interview is appropriate when the knowledge engineer desires specific information and usually results in more useful knowledge base content.

a. Unstructured interviews

During unstructured interviews the knowledge engineer allows the domain expert to introduce concepts, vocabulary, and ideas and set the overall direction of the interview. The knowledge engineer's role is essentially to record the expert's statements and encourage expansion on points that appear important. Unstructured interviews are useful in gaining a sense of the domain and the range of issues that need to be addressed. On the other hand unstructured interviewing is sometimes allowed to dominate the entire knowledge acquisition process with usually dismal results. Hoffman (1987, p.52) discusses several reasons for this. One problem is that expert system domains are generally large and complex; thus the knowledge engineer and domain expert must actively prepare for interview situations. Unstructured interviews generally lack the organization and structure that would allow this preparation to transfer effectively to the interview itself. Second, domain experts usually find it very difficult to express some of the more important elements of their knowledge. Third, domain experts may interpret the lack of structure in this type of interview as requiring little preparation on their part prior to the interview. Fourth, data acquired from an unstructured interview is often unrelated, exists at varying levels of complexity, and is difficult for the knowledge engineer to review, interpret, and integrate. And finally, largely because of a lack of training and experience, few knowledge engineers can conduct an efficient unstructured interview. Thus, they appear unorganized and may unwittingly allow the expert to pursue tangents and diverge from desired session goals.

b. Structured interviews

Structured interviewing forces an organization of the communications between a knowledge engineer and domain expert. At the outset of each interview, the knowledge engineer specifies his goals for the session. During the interview he provides constant feedback to the domain expert in order to convey his understanding of the problem at hand. The expert will in turn, either correct, refine or reinforce the knowledge engineer's perceptions. As opposed to the informal, wandering nature of the unstructured interview, the structured interview is goal-oriented. The structure provided by goals reduces the uncertainty associated with unstructured interviews and allows the knowledge engineer to prevent the distortion caused by domain expert subjectivity.

2. Protocol Analysis

Protocol analysis involves asking experts to report on, or demonstrate, their decision making process for a specific problem. The knowledge engineer then develops a structure or framework that can be used to represent the information, actions, alternatives and decision rules the expert is using. These techniques are effective for knowledge acquisition sessions focusing on the elicitation of routine procedures, facts, or heuristics for any phase of the knowledge acquisition. Three types of protocols are in current use by knowledge engineers: verbal protocols, motor protocols, and eye-movement protocols.

a. Verbal protocols

The acquisition of knowledge through the use of verbal protocols is easy to understand and one of the most common methods of acquiring detailed knowledge from the domain expert. The domain expert is required to perform his tasks while thinking out loud about what he is doing. The knowledge engineer records every detail of what the domain expert is doing and how he appears to be processing information. The notes of the session are later transcribed and encoded as required.

b. Motor protocols

Motor protocols are used primarily as a way of supplementing verbal protocols. Obviously, in tasks that involve either essential or numerous physical activities, motor protocols are critical. To obtain protocols, observations of the expert's physical performance of the task, such as walking, reaching, and pulling, are recorded. Documentation can be done by having the knowledge engineer verbally record the activities taking place or by using a video recording.

c. Eye movement protocols

An eye movement protocol involves the use of sophisticated eye-movement cameras to record the movements of a domain expert's eyes. By evaluating an experts eye motion patterns, a trained knowledge engineer can determine the relative importance or sequence in which an expert evaluated different stimuli. As in motor protocols, it is used to supplement not replace verbal protocol analysis.

3. Walk throughs

Walk throughs resemble protocol analysis in many ways, the chief difference being that walk throughs are not conducted in real time. Because it does not take place in real time the knowledge engineer is able to probe for additional information when needed. A variation on this technique is known as the "teach through", during which the domain expert instructs the knowledge engineer on how to perform the particular task at hand. The knowledge engineer is encouraged to ask questions and to probe the domain expert on matters which he does not fully comprehend. Walk throughs offer several advantages over interviews: they take place in the normal environment of the task, thus offering cues to the expert's memory; they represent an actual problem-solving exercise and, as such, are a type of protocol; and they are relatively unobtrusive since they do not take the expert from the work place. The disadvantages when compared to protocol analysis are: the task is not in "real time," and thus the knowledge engineer may not be actually getting the details of normal problem solving; since the task performed is set up by the knowledge engineer, knowledge about how one task interacts with other tasks in other domains, may be unattainable; and, since the walk through is not under any time constraint, the expert may digress on irrelevant tangents, particularly if the knowledge engineer is asking questions during the session.

4. Questionnaires

Questionnaires may also be beneficial in certain situations. Subjective questions are appropriate for use in the early stages of knowledge acquisition in identifying domains which will require further exploration later on in the knowledge acquisition process. Clearly, open ended questions can lead to several problems. Experts may not enjoy writing responses to broad questions and will truncate their answers in order to "get it over with." At the other end of the spectrum, they may get long winded or head off on a tangent to the problem being addressed. The knowledge engineer is not available to keep him on track. Short answer questionnaires however, may be beneficial to obtain specific answers to questions the knowledge engineer has regarding previously gathered responses. They may prove less obtrusive to the domain expert and enable a lengthy project to flow more smoothly. Forced answer questionnaires are largely used in validating previously acquired knowledge. The domain expert is forced to examine the validity of previously supplied knowledge.

5. Expert Reports

Although frequently used in the past, knowledge engineer's have tended to shy away from expert reports recently. This method involves the expert simply writing a narrative of how his job is performed. The knowledge engineer then interprets and analyses the report in order to obtain the required knowledge. They have largely fallen out of favor for a number of reasons: (McGraw & Harbison-Briggs, 1989, p. 217)

- They essentially require the expert to act as a knowledge engineer, without a knowledge engineers training.
- Expert reports tend to have a high degree of bias; the reports typically reflect the expert's opinion concerning how the task "should be done" rather than "how it is really done."
- Experts will oftentimes describe new and untested ideas and strategies they have been contemplating, but still have not included in their decision-making behavior. The mixing of actual behavior and "ideal future" behavior is endemic.
- Expert reports are time-consuming efforts, and the expert loses interest rapidly. The quality of information attained will rapidly decrease as the report progresses.

However, given these caveats, under certain conditions, such as the inaccessibility of an expert or the knowledge engineer, expert reports may provide useful preliminary knowledge discovery and acquisition.

6. Automated Knowledge Acquisition

Knowledge acquisition is a time consuming and expensive component of the expert system development process. The time required to extract expertise and translate it into code consumes a significant share of any system development resources. Difficulties stem from an inability to access the expert and problems associated with expressing expertise, to the application of knowledge acquisition techniques and the inability to map a domain expert's knowledge into an appropriate representation scheme.

To alleviate some of these problems, various techniques and programs have been developed which automate the knowledge acquisition and in some cases representation. Although the early tools were little more than intelligent editors, the most current systems are known as "workbenches." They are capable of manipulating the process of conceptualization, knowledge mapping, elicitation, and even representation. Typically they promote interaction between the domain expert and the computer system itself, so that the knowledge engineer acts primarily as a facilitator. In some instances, these methods can prove more competent than humans in acquiring knowledge and they tend to operate at a significantly lower cost. Although unavailable for review, there exists a companion program to our development platform, NEXPERT OBJECT® called NEXTRA® which is an integrated tool for knowledge acquisition. Prior to a full scale knowledge acquisition effort the project may reap many benefits by acquiring and implementing this tool.

7. Techniques for Using Multiple Experts

Many of the techniques described above can easily be adapted for use in a multiple expert environment. Discussion between domain experts during walk-throughs for example can be helpful in clarifying issues that a single expert may gloss over. Further, multiple experts may contribute knowledge during the session that is not utilized by a single expert. Methods commonly in use for problem solving such as the Delphi method, brainstorming and even group decision support systems can be adapted for use as knowledge acquisition methods. All of the following methodologies have been successfully applied by knowledge engineers in working with multiple experts.

a. Brainstorming

Brainstorming encourages the free flow of ideas by relieving the tension members of a group may have in proposing solutions to problems. In brainstorming, quantity is preferred over quality. The knowledge engineer wants to get as many solutions on the table as he can in a short amount of time. When the rate of idea presentations stagnates, the session is debriefed with a discussion of the ideas that have been introduced.

b. Consensus Decision Making

A technique that can follow brainstorming is known as consensus decision making. The aim in this type of session is quality vice quantity. The team of domain experts focus on and measure the benefits and costs of each solution until they come up with the best answer.

c. Nominal-Group Technique

An extension and modification of the brainstorming process, the nominal group technique removes the vocal interaction that may inhibit some individuals. Group members work alone but in the same room, developing ideas. They then share their lists of ideas, one item at a time in round-robin fashion. This approach appears to yield more ideas than brainstorming, yet keeps some of the advantages of that technique. (Casey, Gettys, et al., 1984, pp. 112-139)

D. SUMMARY

The knowledge obtained from a domain expert lies at the heart of a knowledge based system which makes the process of obtaining that knowledge the key to developing an expert system. The knowledge engineers must fully immerse themselves in the project and place themselves as much as possible in the shoes of the domain expert. Because of the complexity of the naval aviation maintenance domain, a thorough formal and practical education is essential.

Although there are unquestionably many career maintenance controllers who could easily satisfy any standard of expertise within their field, they may not so easily qualify as domain experts. Equally important as technical expertise is the ability of the domain expert to function as part of the knowledge engineering team. He must be able to clearly analyze his own behavior and assist the knowledge engineer in formulating the production rules which will represent his expertise.

There exist many techniques to elicit knowledge from domain expert, several of which are discussed above. A combination of interviewing, protocol analysis and walk throughs have been conducted to establish the first series of production rules. It is likely that these three techniques will account for a significant portion of the entire knowledge acquisition process. Although not reviewed for this thesis, automated techniques using NEXTRA® may also play a significant part in the final development effort.

IV. AIRCRAFT MAINTENANCE ENVIRONMENT

The maintenance of Naval aircraft is the most expensive and manpower intensive facet of squadron operations. The cost to the taxpayer in maintaining these complex systems is in the billions of dollars and increasing annually. The aims of maintenance management are to increase productivity, minimize the cost of preventative and corrective maintenance, decrease the frequency of breakdowns and improve the general efficiency of the maintenance process. These aims are difficult to achieve because of the complexity of the maintenance scheduling problem. There can be no general, algorithmnic solution as the answers depend on the operational schedule, environmental factors, type aircraft and general maintenance management philosophy. Clearly, traditional MIS's are not capable of processsing the types of information required to be generated. The expertise required cannot be codified in traditional methods. An expert system does enable this type of knowledge and expertise to be captured, codified and processed and represents a likely solution to the problems cited above.

In order to fully appreciate the scope of the knowledge and expertise required for the ESAAMS project it is important to understand the environment under which aircraft maintenance scheduling is performed. Although to a lesser degree when shore based, aviation squadrons continue to operate in an extremely high tempo, "must do" environment. Squadrons are heavily tasked to provide ready aircraft to meet battle group commitments. Missing missions or even worse, having your sister squadron pick up missions that you cannot perform is something that a squadron commanding officer cannot tolerate. Accordingly, the person selected for the prestigious and powerful task of running the maintenance department is generally a very professional highly qualified "expert". Although in some squadrons this expert may be an officer, he is usually a very senior enlisted man with significant experience at both the technical and managerial levels of the aircraft maintenance organization. For the purposes of this thesis, who is in the position is not imperative, however the position itself is central to the expert system design.

A. THE MAINTENANCE/MATERIAL CONTROL OFFICER (MMCO)

The MMCO is the singular personality within a squadron who is most frequently considered the domain expert. Those most often recognized as experts in the aircraft maintenance control work centers generally have at least eight to twelve years of experience in the nuts and bolts of aircraft maintenance and an additional several years under the direct supervision of a recognized "expert" in maintenance planning and scheduling. The superior performers clearly stand out within their very talented peer group. Inspection teams and personnel who have been working within a community for a long period of time can readily identify those truly superior MMCO's whose expertise which we want to capture.

B. CONSTRAINTS

There are many factors which impact the MMCO's maintenance scheduling decisions. Some factors, which can be refered to as constraints, are those which are hard and fast. There is little room for manipulation of these items and the domain expert is forced to confront these factors head on before addressing the “influence” factors which will be discussed later.

1. Flight Schedule

A maintenance man's dream may be to have the authority to write the daily schedule. The ability to conduct both scheduled and unscheduled maintenance unhampered by operational commitments would make his task easier and less pressured and would obviate the need for this expert system. As in any typical business, however, pressure motivates workers to efficiently allocate resources in a productive and useful manner. The flight schedule is taken as gospel within a squadron and if a mission appears on the schedule, the maintenance department is obligated to provide an aircraft for that event. Additionally, many squadron commanders will require that a spare aircraft be on the flight line and ready to fill in for the primary aircraft in case of mechanical breakdown.

2. Time to Repair

The tendancy among MMCO's is to maximize the number of up, or fully mission capable aircraft at any given time. Hence, given two candidate aircraft to place in work, the maintenance controller will induct that aircraft which he calculates will be quicker to repair. To select among several aircraft to place in work, he will scan the Visual Indicating Disply System (VIDS) boards for the aircraft with the fewest downing or not mission capable (NMC) discrepancies. These are usually highlighted by a red mark overlaying Job Control Number (JCN) of the VIDS maintenance action form (VIDS/MAF). He will then evaluate each NMC discrepancy against that particular aircraft to determine an estimated time to bring it into a mission capable (MC) status. In estimating the time to repair, the MMCO must make a best guess at diagnosing the cause of a discrepancey. Based on his experience he will determine, with some degree of confidence, what the malfunction is, what he needs to repair it, and how long it will take to repair.

3. Scheduled Maintenance

Scheduled or planned maintenance is a series of inspections which ensure that aircraft are maintained throughout their life cycle by controlling the aging process and the natural wear incurred due to regular landings and takeoffs, pressurization cycles and exposure to salty air and sea spray. Many separate functions and tasks are combined to make up a particular set of inspection requirements which are known as Maintenance Requirement Cards (MRC's). In order to obtain the intended benefit of the planned maintenance system (PMS), inspections must be performed in sequence and within a specified interval of time. Preventative maintenance can be classified as phase, special, and conditional inspections.

a. Phase Inspections

The phase maintenance concept divides the total scheduled maintenance requirement into small packages or phases of approximately the same work content. These are done sequentially at a specified interval throughout the service life of an airframe. Phase inspections are tailored to a specific airframe type/model/series (TMS). Depending on the TMS,the time allowed between inspection varies anywhere from 100 to 200 flight hours. For example an F-14A Tomcat has a phase interval of 100 hours, where an S-3A Viking has an interval of 170 flight hours. Activities are allowed to perform the inspection in a window bounded by the base flight hours plus or minus ten percent of the inspection interval. In the case where an aircraft is due for a Phase B inspection at 970 flight hours, and assuming an inspection interval of

150 flight hours, the inspection may be performed anytime between 955 and 985 flight hours. The squadron also has the option of conducting the inspection prior to 955 hours provided that they reestablish the base date of the inspection cycle. For example, if the squadron decided to perform the above inspection at 930 flight hours it may do so, provided that the next phase inspection becomes due at 1080 hours. Although this adjustment can sometimes be beneficial, one must recognize that in the long run, this will waste maintenance man hours by conducting inspections more frequently then required. Returning to the example aircraft, if a squadron was unable to conduct the inspection prior to the window expiring, it must request permission to exceed the limit by another ten percent and if that extension is granted may <u>not</u> adjust the base date for the following inspection. This type of wavier is seldom granted and in fact repeat requests for such waviers will invite unwanted assistance and oversight from higher echelon commands.

b. Special Inspections

A special inspection is one which is performed at a specified interval other than a phase inspection. These intervals are different for each type of aircraft and generally are based on elapsed calendar time, flight hours or number of cycles or events. For instance many aircraft have a 7, 14, 28, 56 and 210 day, 10, 50 and 150 hour, and 10 and 100 arrested landing inspection requirements. These inspections also have windows in which they can be performed, but they vary from inspection to inspection and it would be unproductive and unneccessary to list those here.

c. Conditional Inspections

Conditional maintenance requirements are unscheduled events required as the result of a specific over-limit condition. Events such as lightening strikes, hard landings, over-speed, engine over-temp and hard landings are typical of the situations in which conditional inspections play a part. These conditions are called for in order to inspect the aircraft when it is likely that some sort of damage may have occured. Obviously, it makes no practical sense to provide for an extension of this type inspection.

4. Technical Directive Compliance

Technical directives are issued by Commander, Naval Air Systems Command and specify certain maintenance which must occur as a result of either newly discovered defects which could affect the airworthiness of naval aircraft or in an effort to improve the reliability or maintainability of those aircraft. Similar in nature to airworthiness directives issued by the Federal Aviation Administration, compliance with them is mandatory. Depending on the urgency of the maintenance required, maintenance may have to be performed prior to the next flight or any other interval specified in the directive. Based on the results of inspections so directed, permanent or temporary restrictions on the aircraft operating envelope may be imposed. For instance an aircraft may be restricted to day time flight or to a certain "g-force" limitation until a further directive can be complied with.

5. Support Equipment Availability

With the complexity of weapon systems installed in today's aircraft comes a plethora of support equipment required to maintain those systems. Often this equipment is not available in sufficient quantities to enable each squadron to have its own set. Instead, the entire package or selected items will be made available from the supporting air station or ship aircraft intermediate maintenance department (AIMD). Obviously this will lead to certain items of support equipment not being available to the maintenance department when required. In certain circumstances the use of this equipment is required prior to certifying the aircraft safe for flight. In other instances, it may be permissible to allow the aircraft to function as a test platform. An expert system should have the knowledge of what type discrepancies will require specific pieces of support equipment and determine the availability of that equipment prior to advising the maintenance controller to perform repair of that discrepancy.

6. Parts Availability

One of the most ambiguous areas for the expert system to address is the availability of repair parts. Although one may think that either the parts are available or they are not, it is not quite so simple. In recent times, more dollars have been expended to purchasing systems than to procuring repair parts. As a result, squadrons have become accustomed to cannibalizing airframes for required parts. That is to say, it is often more expedient to obtain parts from aircraft not on the flight schedule, then to wait for the supply department to deliver them. Other squadrons also, are valuable, if unofficial, sources of supply which will loan parts from their aircraft, if they can do so without impacting their operational schedule. In this situation, an expert system may recommend an aircraft within the squadron which it perceives as a potential donor of a part, if the supply system can not produce the required item.

7. Manpower

Although it may be an unwise assumption to presuppose that all maintenance departments are equally talented, in the context of this expert system project it is an assumption which will have to be made. In a pure expert system this would be unacceptable, however ESAAMS is designed to act as an advisor to the MMCO and he will have to fine tune the maintenance schedule to account for his manning strengths and weaknesses.

C. INTERNAL INFLUENCES

Internal influences of the decision maker are those factors within his organization which impact his decision making processes. Within the context of ESAAMS, the commanding officer, operations officer, and maintenance officer are the internal influences which impact on the MMCO in the course of his adjudication. Though minor variations may occur in the organization of naval aviation squadrons, they are essentially identical and for the purposes of this thesis will be treated as such.

1. Commanding Officer

Aviation squadrons generally operate with a great deal of autonomy and are given a significant amount of latitude in determining how best to perform their mission.. Commanding Officers are presented with tasking by higher authority or in many cases they may and do create tasking internally. The commanding officer's superiors hold him responsible for carrying out all tasks safely and expeditiously. As a relatively junior Commander, the squadron commanding officer is competitive by nature. In order to be selected for advancement:

- He will seek to operate his squadron at a pace which will make it stand out from similar squadrons
- At the same time,maintaining his aircraft in top material condition
- And keeping the morale of the squadron personnel high.

Unfortunately, the above three goals are conflicting in nature and the Commanding Officer must maintain a balance between the necessarily competing objectives in influencing the MMCO.

2. Operations Officer

Within the squadron organization there are two officers responsible to the Commanding Officer for the two most important functions of the squadron. The operations officer is the CO's primary assistant when it comes to aircraft tasking, training and scheduling. He is responsible for ensuring that all aircrew maintain current qualifications in a variety of areas including night flying, airways navigation, aerial refueling, carrier qualifications and formation flying. Additionally he must ensure that they are able to utilize the various weapons systems integral to the aircraft, such as the weapon control, electronic countermeasure, or photo reconnaissance camera systems. Given the multiple missions assigned to any particular aircraft and considering the varying degrees of experience of squadron aviators, matching the needs of the squadron with the capabilities of its airframes is never an easy task. In scheduling training missions, he must specify aircraft configuration, fuel loads and weapons loading instructions. Changing configuration of the aircraft may require significant lead time in order to draw the necessary equipment and parts from supporting activities.

3. Maintenance Officer

The CO's primary assistant for aircraft material is the Maintenance Officer. In addition to his normal aircrew duties, he must ensure that aircraft are available to meet the flight schedule requirements and that those aircraft are properly configured for the tasked mission. He acts as a buffer or equalizer between the flying and maintenance sides of the house and generally passes the inputs of the MMCO up the chain of command and urges that those concerns be given equal redress to the concerns of the operations officer.

D. EXTERNAL INFLUENCES

There are indeed multiple influences both within the individual organizational maintenance activity and external to the organization which exert influence upon the domain expert's decision making process. The policies established by the various commands and activities, although not by design, often provide conflicting direction and advice to maintenance organizations and hamper the effectiveness of the professional maintenance managers. A well engineered and tested expert system would clearly identify these conflicts and as one of its unintended benefits may well empower maintenance controllers with the broader authority to operate their maintenance departments.

1. Type Commander/Functional Wing

Type commanders(Commander, Naval Air Force Pacific Fleet for example) and functional wings(Commander Fighter Airborne Early Warning Wing Pacific Fleet for example) are the two immediate administrative bodies over the squadron in the chain of command. They set policy as it relates to the operation, maintenance and training of squadrons under their cognizance as well as provide logistic support to the squadrons as they prepare for scheduled deployments. Two of the many programs overseen by type and functional wings are listed below as well as a discussion of how they impact aircraft maintenance scheduling.

a. Integrated Weapon System Review (IWSR)

IWSR is a program directed by the functional wing which is a training exercise that all squadrons must participate in once during every turnaround cycle. Lasting about six weeks, each squadron is tasked to provide a total of about fourteen personnel from all ratings to the IWSR team. After a two week classroom period, the squadron must provide a fully mission capable aircraft for the team to perform a complete and detailed weapon system performance checkout. Clearly a beneficial program from a training standpoint, it removes one aircraft asset and a cadre of usually superior performers from the maintenance effort.

b. Special Interest Aircraft

The Special Interest Aircraft (SPINTAC) program was developed in order to address those particular aircraft assets within a squadron which have not flown for a particular length of time. When an aircraft has not flown for thirty days, regardless of the reason, the chain of command is required to be notified as to the status of the aircraft and its estimated fly date. At the 45 day no fly point, a SPINTAC ALERT message is required to restate the facts presented in the 30 day notification and at 60 days an aircraft is placed in SPINTAC status. Although various type wings handle the SPINTAC program slightly differently, at some point in the process, the aircraft can no longer be cannibalized, nor can parts be diverted to other squadron aircraft which are intended for the particular SPINTAC aircraft. The pressure to avoid SPINTAC status can be so intense as to cause squadrons to cannibalize squadron aircraft solely to prevent an aircraft from going thirty days without a flight as well as incur inordinately long maintenance hours.

2. Ship and Naval Air Station Policies

In addition to the influences cited above, aircraft carriers and naval air stations have a host of regulations which also significantly affect the squadron maintenance plan. Noise abatement procedures in place at many naval air stations generally impact the ability to conduct high power maintenance turns during night time hours and on Sundays. Environmental regulations play a part in when and where squadrons can apply paint or primer to aircraft. When at sea, maintenance is very dependent on where the aircraft are located on the flight deck. If the ship is steaming under bad weather personnel may not be allowed to move to the flight deck to perform maintenance and that same bad weather may impede the movement of aircraft to the hangar deck where they could be worked on safely.

Cited above are just a few samples of the effect that external factors have on the maintenance scheduler. These factors significantly limit the maintenance controllers options and it is imperative that ESAAMS be equipped to deal with these restrictions and that it be easily modified to reflect the imposition and relaxation of the various restrictions.

E. SUMMARY

In summary, the operation of maintenance control within an organizational maintenance squadron revolves around the MMCO. In order to be successful he must have a solid picture in his mind at all times of the status of the aircraft, the discrepancies that are currently being worked on and those that will need to be worked on next. The location of aircraft, availability of parts, condition of support equipment are just a few of the items of information which he must have at a moments notice.

In planning his maintenance he must take into account the myriad policies, programs desires of his superiors. Often provided with conflicting priorities developing his daily schedule is not an easy task. The many policies he must comply with however, can be codified and implemented using expert system technology. Although the MMCO will never be replaced by hardware or software, the quality of his decisions cannot help but be improved through the implementation of a soundly developed expert system.

V. KNOWLEDGE REPRESENTATION

Following the knowledge acquisition process, the knowledge engineer must determine how the chunks of knowledge are to be represented in the structure of the expert system. It is not necessary that he limit his design to one representation scheme, indeed the structure of the system may be composed of modules using any of the various techniques available. Four of the most popular approaches are discussed below followed by a proposed system architecture. It can not be emphasized enough however, that the selection of a representation scheme prior to completion of the knowledge acquisition process could jeopardize the success of the resulting system.

A. PRODUCTION SYSTEMS

1. Description

Since the earliest expert systems were released, the dominant scheme for representing knowledge in the artificial intelligence arena has been the production system. Production systems are composed of three distinct elements: (Merritt, 1986, p. 31)

- The rule set.
- A working storage area that contains the current system state.
- An inference engine that knows how to apply the rules.

Rules serve to accurately represent the heuristics which an expert uses to resolve a particular problem. They can quite readily be represented as a series of if-then statements as shown below.

If the aircraft is not mission capable,
then the aircraft can be inducted for repair

or in another example:

If the aircraft is not mission capable
and the estimated repair time exceeds 96 hours
and it is due for corrosion control repairs
and no other aircraft is in corrosion control spot,
then induct the aircraft for corrosion control

The “if” side of the equation states the condition or conditions that must be true in order for the rule to apply. The “then” side of the equation specifies the appropriate action to take. When the inference engine evaluates the “if” portion of a statement as true, the operative portion of the statement is added to the knowledge base. Using our examples above, if both were true, the following statements would be added to our knowledge base.

- The aircraft is not mission capable and can be inducted for repair.
- The aircraft is not mission capable, will be down for 96 hours, is due for corrosion repair and since no other aircraft is in the corrosion control spot, the aircraft can be inducted for corrosion control.

The inference engine then utilizes the data which is resident in the knowledge base and decides which rule will be applied next. This entire process then repeats itself until the end of the reasoning chain is reached.

2. Advantages of Production Systems

One clearly evident advantage of the production system is the ease with which the inference chain may be modified. By simply adding new rules or modifying existing rules, the performance of the system can be easily modified, although as systems become larger, this modularity becomes harder to maintain. (Rychener, 1976, pp.87-90)

The if-then structure of the production system lends a consistency to the knowledge base that is not always evident in other methodologies. Because of this uniformity, the rules can be easily explained to and understood by a human expert. The benefits of this can be easily seen in a system such as the MYCIN system. (Shortliffe, 1976, pp. 77) The MYCIN system acts as a medical consultant, aiding in the diagnosis and selection of therapy for patients with bacteremia or meningitis infections. It carries on an interactive dialogue with a physician and is capable of explaining its reasoning processes.

3. Disadvantages of Production Systems

The most significant disadvantage of a production system is the inefficiency with which the program is executed. The iterative methodology with which each rule must be evaluated for context matches results in extraordinary overhead.

Secondly, the rule structures used in a production system are not well suited for representing procedural information. The flow of control is much less apparent than it would be in a system which used algorithms. With procedural information, the knowledge engineer must be concerned with the order in which rules fire, yet the entire focus of rule-based representations is to take the ordering considerations out of developers hands.

B. SEMANTIC NETWORKS

1. Description

One of the most popular methods of representing knowledge in artificial intelligence research today is the semantic network. First developed by Quillian and others, it was invented as an explicitly psychological model of human associative memory. (Quillian, 1968, p.227) That a model of human associative memory serves equally well as a model for machine associative "thinking" should come as no surprise. A semantic network consists of a series of nodes connected by arcs which describe the relationship between two nodes. Nodes represent objects, whereas arcs represent the relationship between two nodes and can be thought of as "isa" or "has-part" connective statements. As an example consider Figure 5-1 and the statements "The airplane has an engine" and "The starter is part of the engine."

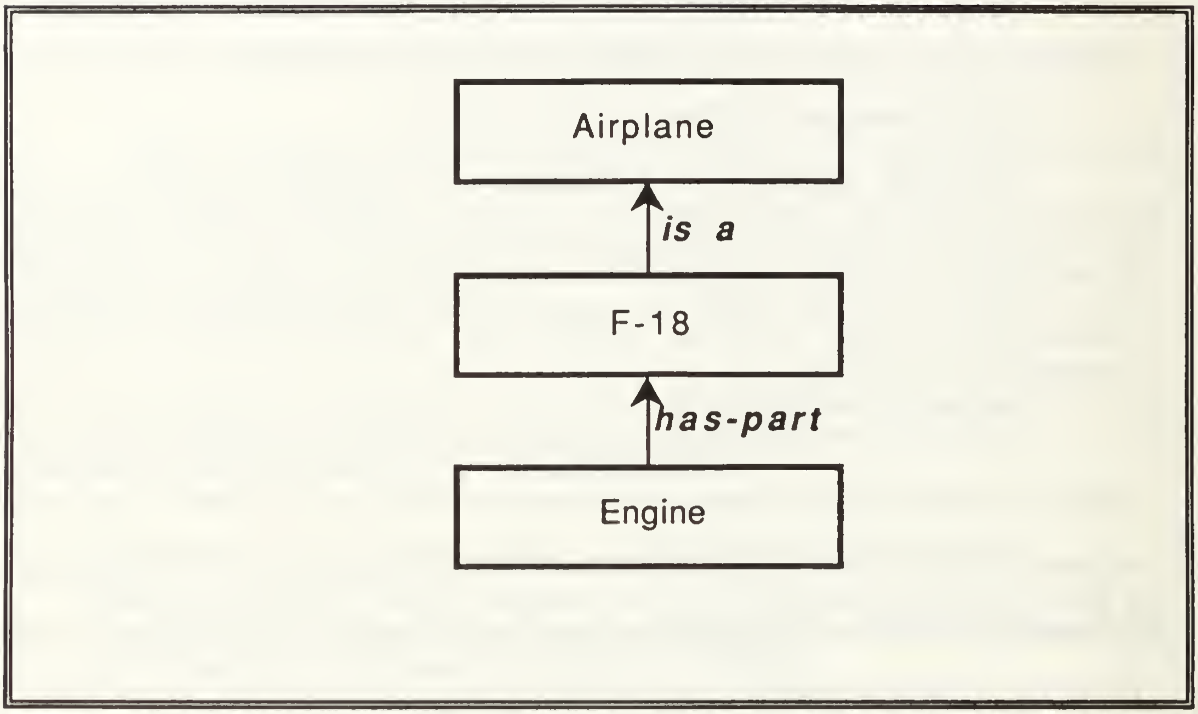

Figure 5-1. Illustration of a simple semantic network

Observing the transitive relationship between nodes one and three, we can infer a third statement from the network, that "The airplane has an engine" even though that relationship has not been explicitly stated.

2. Advantages and Disadvantages of Semantic Networks

The ease with which it is possible to make deductions about inheritance hierarchies such as this is one reason for the popularity of semantic networks as a knowledge representation scheme. The major shortcoming of early semantic networks was their inability to handle other than binary relationships. For example suppose you wanted to indicate in our example that an airplane has either General Electric engines or Pratt and Whitney engines. In order to overcome this shortcoming Frame-based knowledge representation was proposed.

C. FRAME BASED KNOWLEDGE REPRESENTATION

1. Description

Frame based and semantic network knowledge representation schemes are very closely related. Simmons and Slocum proposed a solution to the binary constraints imposed on relationships by semantic networks which allows nodes to represent situations and actions, as well as objects. (Simmons and Slocum, 1972, p. 891) A frame is a data-structure for representing a stereotyped situation, like the status of a certain supply requisition document, or the present configuration of an aircraft. Attached to each frame are several kinds of information. Some of this information is about how to use the frame. Some is about what one can expect to happen next. Some is about what to do if these expectations are not confirmed. (Misksy, 1985, p.160-176) A frame is similar in nature to a record structure in the ADA or Pascal programming languages. Frames are organized into a generalization hierarchy in which frames inherit information from their parent nodes. The attributes are stored in slots which can either take on values or describe, in general terms, constraints on what the values can be. Data can be stored in slots in numeric, symbolic, text, logical or even graphical formats. A node in a frame based system can generally be thought of as the structure shown in Figure 5-2. (Walters and Nielsen, 1988, p. 215)

Aircraft Paint	
Color	Gull Grey
	Restriction:(Value-Type:Symbol)
	Restriction:(Content-One of Red, Blue White, Gull Grey, or Black)
	Restriction:(Max-number of values: 1)
Price	36.99
	Restriction: (Value-Type: Decimal)
	Restriction: (Content- One of 36.99 or 0.00)
Surfaces	(Aluminum, Composite, Depleted Uranium)
	Restriction:(Max number of values: 1)
Instructions	Prepare surface by removing any loose paint, dirt, or grease. Apply primer...dry for 8 hrs.
Type	Polyurethane
	Restriction:(Content not one of :Water based)
Gloss	True
	Restriction: (Value type: Logical)
	Restriction: (content one of true, false unknown)
	Restriction: (Max number of values: 1)

Figure 5-2. An example ESAAMS frame

Slots may contain information passed to them from a parent node or they may be assigned default values when they are designed. In the example above "Gull Grey" is assigned the default value as a color and the type of surface is a value which would be passed from an adjacent node. Whether or not the slots are consistently ordered throughout the net is largely dependent on the implementing system.

2. Frame Based Reasoning

The above discussion deals exclusively with individual frames, without regard to how frames relate to one another in the context of an expert system. Individual frames are related to each other in the very same way that nodes are related to each other in a semantic network, with "isa" or "has-part" constructs. Frames loaded with general information are located at the top of the hierarchy and as you progress downward, the frames become increasingly more specific. Generally, there are three separate actions which may happen in relation to a slot. (Waterman, 1986, p. 74)

- If-added procedure: Executes when new information is placed in the slot.
- If-removed procedure: Executes when information is deleted from the slot.
- If-needed procedure: Executes when information is needed from the slot, but the slot is empty.

To initiate the process, a value is inserted into a slot at the top of the hierarchy. An 'if-added' procedure is initiated and the process takes off like a chain reaction, querying the user for needed information along the way to process completion at the lowest echelon.

3. Advantages and Disadvantages of Frame Based Representation

Most of the data processing aspects of this system take place within each individual frame, and the results of that processing are passed to another frame. This is conceptually similar to object oriented programming (OOP) in that each frame can be thought of as an object. In its similarity to OOP lies both the strengths and weaknesses of frame based knowledge representation. The highly structured methodology of the frame simplifies the design and construction of an expert system. The modularity of the frames enhances the portability and maintainability of the knowledge base. Like rule based systems, a major problem in the use of frame based systems is the fact that they can consume an inordinate amount of central processing unit (CPU) cycles. One should be forewarned that reasoning with frame based knowledge is a relatively straightforward process and if the designer has problems representing knowledge with frames, they should consider using a different representation.(Walters, 1988, p. 250)

D. BLACKBOARD REPRESENTATION

1. Description

The blackboard architecture is one in which independent knowledge sources communicate via a central structured data base, known as a blackboard. The name is derived from the way in which several people may gather around a blackboard to solve a problem. Every expert in the group possesses some unique knowledge that is not known by another group member. One by one the group leader requests certain facts from the members in the group and writes those facts on the blackboard. Aware of the expertise of all the group members, the leader is able to direct the inquiries in directions that appear to be most productive. Using the above analogy, we can identify the three subsets of a blackboard system as:

- Knowledge sources
- Blackboard
- Control.

2. Knowledge Source

Each knowledge source represents an area of expertise pertaining to the problem being addressed. In an aircraft maintenance scheduling system, one knowledge source may be the historical data relating to repair cycle times. Others may relate to specific aircraft systems, and still others to a specific aircraft. These sources could take on many different forms including data bases, sub-expert systems or even a procedural program. Each knowledge source is comprised of two major components. The first component is the knowledge that is to be contributed in solving the problem. The second component decides whether or not the first component can contribute to the problem at hand. The former is known as the action component and the latter as the condition component.

3. Blackboard

The blackboard can be thought of as a central clearinghouse through which all the information is exchanged. Under the blackboard system, knowledge sources must communicate through the blackboard; no direct communication between knowledge sources is permitted. Two different types of knowledge are mounted on the blackboard, static and dynamic knowledge. Static knowledge is that knowledge about the problem which does not change. Initializing conditions, constraints and associations, For instance, “the airplane is broken and must be fixed within 24 hours,” and “There is no hangar space available for twelve hours." Dynamic knowledge is that knowledge which is generated by the system. It includes requests for data, newly generated facts, hypotheses, goals and suggestions. The dynamic data will be frequently updated, modified and deleted as the system operates.

4. Control

The control subset is a very specialized knowledge source. Although it functions mechanically, much like the other knowledge sources, it assumes responsibility for the operation of the system as a whole. If progress is not evident after some set time period, the control may, by placing new information on the blackboard, steer the other knowledge sources in a different direction, in an attempt to break the deadlock. The structure of the control, now becomes critically important to the performance of the system as a whole. Controls are presently arranged in one of the four following ways.

- Event-driven
- Expectation-driven
- Request-driven
- Goal directed.

a. Event Driven Controls

Event-driven controls react to the materialization of new events on the blackboard. When new knowledge is received, the control selects the knowledge source or sources best suited to respond to the new data. It may also respond to infractions on the parameters of the system, (looping, overflows, etc.) by passing control to knowledge sources designed to handle the general housekeeping chores.

b. Expectation Driven Controls

Expectation-driven controls must be preset with a general idea of how the system is expected to operate and so is especially suited to systems involving network or process control. Based on its own knowledge of how the system should be responding, and the knowledge on the blackboard as to how the system is responding, the control can direct processing to appropriate knowledge stores.

c. Request Driven Controls

Request driven controls reflect the most passive control structure. This control simply directs specific knowledge sources to respond to requests from other knowledge sources for data.

d. Goal Directed Controls

Given a hypothetical response on the blackboard, the goal directed controls select knowledge sources which are likely to be able to prove the hypothesis. If the control senses that little progress is being made in proving the goal, it may redirect the system towards proving an alternate hypothesis.

5. Advantage of the Blackboard

What has not been mentioned thus far is the fact that generally a blackboard system will consist of many blackboards all working with different knowledge sources. They overall system is hierarchical in nature with the upper level blackboards receiving and processing information from the lower level blackboards. It is possible for blackboard systems to engage in top-down or bottom-up processing. That is they may take many specific problems and generate an overall solution, or they may take one big problem, break it down into specified sub-problems and solve them.

E. SUMMARY

Representing all the knowledge which will be required in constructing an expert system for aircraft maintenance scheduling will not be an easy task. After careful study it seems "all of the above" is the correct solution. Given the broad range of knowledge to be captured, our system will likely require the benefits of several different representation schemes. With its extraordinary flexibility, the blackboard architecture seems particularly appropriate for controlling our proposed system. The blackboard readily accommodates the use of various knowledge representation schemes which will be encompassed in our expert system. Figure 5-3 displays a candidate architecture for a ESAAMS prototype system.

Throughout the readings various authors have emphasized the need to decompose problems into many small component problems. The blackboard architecture is particularly suited to managing knowledge from different domain sources and placing all that expertise under the control of a "boss" system. It can determine strategies to follow and when to terminate those strategies that appear to be leading to non-productive solutions. It is adept at determining what knowledge applies to a particular situation and how to integrate the knowledge on the blackboard. The scheduling problem demands that multiple choices be provided to the user and the blackboard is amenable to proposing multiple solutions.

The most significant weakness of the blackboard system is its inherent high overhead cost. It requires a high performance central processing unit and significant amounts of data storage capability. It is probably safe to assume that given the trend of the last ten years, that by the time this system is ready to deploy to the fleet, the processing power and data storage problem will no longer be a significant factors.

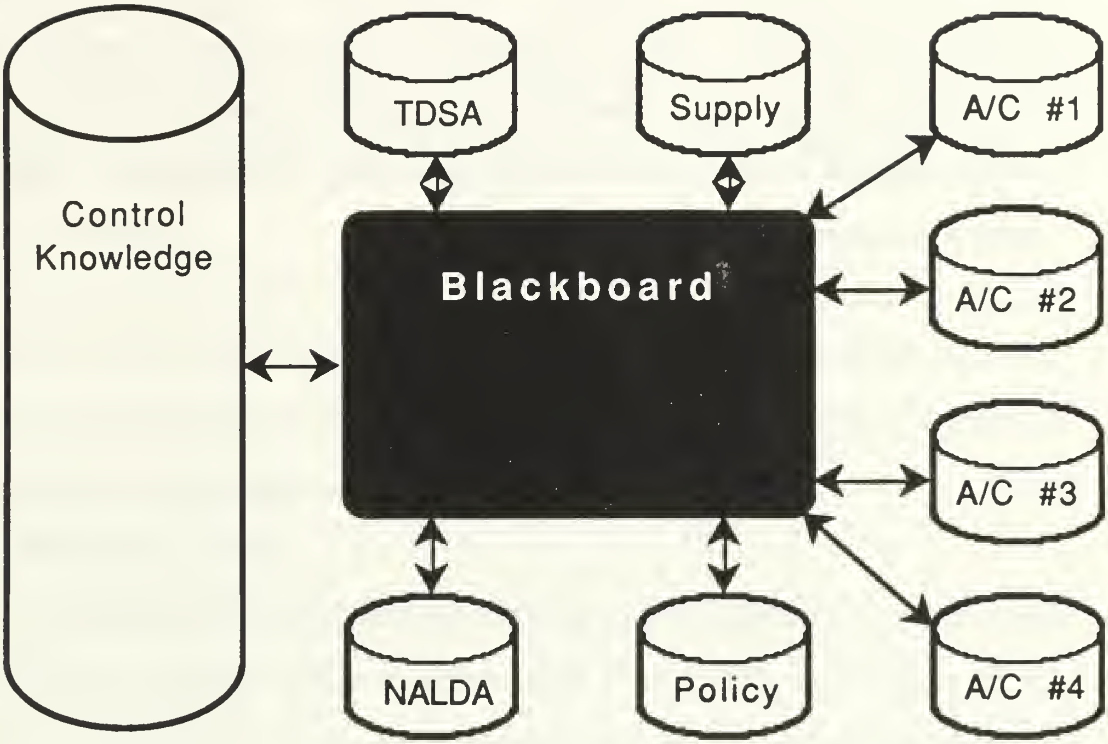

Figure 5-3. Blackboard representation of ESAAMS

Figure 5-3 does not by any means, represent a final picture of our system. It is probable the primary knowledge sources will require further decomposition as the design of our expert system progresses. The policy, NALDA and aircraft knowledge sources will likely be represented using a production scheme/semantic network. The TDSA and supply knowledge will most likely be represented in a frame based scheme. In concluding, it should again be emphasized that knowledge representation schemes are essentially dependant on the knowledge to be represented and the selection of an appropriate representation scheme should follow the actual knowledge acquisition process.

VI. KNOWLEDGE BASE

All of the domain knowledge required for ESAAMS to function is contained in its knowledge base. It contains facts, as well as rules that use those facts as the basis for decision making. This chapter will give a rather general overview of the knowledge base itself and how that knowledge base is maintained. The knowledge base is comprised of a fact base, rule base and working memory.

A. FACT BASE

The fact base contains items of interest to the maintenance expert. Information that is used in the decision making process but which is not a heuristic rule. Examples of the type of knowledge required for the fact base are historical facts, current facts and projected facts.

1. Historical Facts

All maintenance performed on naval aircraft is currently recorded in the Naval Aviation Logistics Data Analysis (NALDA) database. A study of the feasibility of extracting data of a historical nature from the NALDA database for use in ESAAMS, concluded that it is uniquely qualified to provide the information required to serve as a component in the ESAAMS system for the following reasons: (Burpo, 1990, p. 114)

- As Naval Aviation's central repository of logistical and maintenance data, NALDA is the only conceivable source for much of the data required.
- Every aircraft maintenance expert likely to be interviewed during the knowledge acquisition process will be thoroughly familiar with the data elements contained in the various NALDA databases. These data elements can thus serve as a "common language" when expert reasoning are consolidated.

- The source of much of NALDA's data, the three Maintenance Data System cycles, are in place and functioning throughout the U.S. Navy. Despite any shortcomings the system may possess, replacing it or duplicating it would be prohibitively expensive.
- The NALDA system is organized to respond to ad hoc data inquiries. Any data required during knowledge acquisition can be quickly retrieved from one or more of the various databases, and downloaded in a variety of data formats.

NALDA is capable of providing data in a format which is easily imported by all major expert system shells including the prototype development shell, NEXPERT Object®. Currently it is not capable of interacting on a real time basis with our expert system, however off line access would not severely handicap the reasoning process as the system is looking for historical data, not a current picture. The historical information of value to ESAAM would include the following:

- Elapsed Maintenance Time--Among the many data items entered on a VIDS/MAF after a maintenance action is completed are a Work Unit Code (WUC) which uniquely identifies every item of equipment installed in the aircraft and a malfunction code which identifies the mode of failure of the system. Based on these two data items and some statistical analysis routines the expert system could offer a prediction as to the repair time for any given discrepancy. It may also assign certain confidence factors to any given possible repair scenarios.
- Component Failure Trends--When a repairable component is installed on and removed from a naval aircraft, the repair VIDS/MAF is annotated with the serial number of the component, the component time since new(TSN) and the aircraft TSN. Using the installation and removal data, the NALDA system is capable of determining the approximate time of component failure and from that data is able to determine the average or mean time between failures (MTBF).
- Repeat Discrepancy Trends--NALDA data is also extracted from VIDS/MAF's generated at the intermediate or component repair level. If an item demonstrates like failure modes over a period of time, it is an indicator that the testing process at the IMA level may not be detecting the root cause of the component malfunction. On the other hand it may indicate that inadequate repairs are being accomplished.

2. Current Facts

Current facts are those which relate directly to the material position of the squadron and its support structures when the expert system is invoked. Although there is currently no system provided to squadrons to monitor this information, it is absolutely essential for ESAAMS to function. Among the many topics included are the following maintenance related factors.

- Side Number--A two or three digit number which uniquely identifies an aircraft within a squadron. In one squadron the aircraft will be numbered 100, 101, 102, 103, 104 ... and in another squadron they will be numbered 200, 201, 202, 203 and so forth. These numbers can be changed at the discretion of the commanding officer and are used as a local reference only.
- Bureau Number--As opposed to the Side Number, the bureau number is assigned at the time of manufacture and stays with an aircraft throughout its life cycle without regard to modifications or overhauls. Certain inspections, procedures and directives, when promulgated, will apply to specific aircraft only and those aircraft are cited by Bureau Number.
- Readiness Reportable Status--A three digit code which reports the actual readiness status of a particular aircraft. For example, aircraft assigned to a squadron are generally in A10 status which loosely translates to "the aircraft is an asset to the squadron." Any other code indicates that the aircraft has undergone significant damage (crash, fire, corrosion), is enroute to or at an aircraft overhaul facility, or that it is being used for a specific purpose that makes it unavailable to fly, (Training for maintenance personnel, special rework for modification etc.). There are dozens of codes, which mean many different things and what is important to realize is that certain of these codes are indicative of an aircraft in non-aging status. When an aircraft is in non-aging status, it must be preserved and that preservation must be monitored. It further permits the squadron to defer all inspections (other than preservation) until the aircraft is de-preserved.
- Mission Capability Status--Indicates whether an aircraft is Optimum Performance Capable(OPC), Full Mission Capable (FMC), Partial Mission Capable (PMC), Not Mission Capable (NMC). Either an M or an S can be annotated after PMC or NMC to indicate whether supply or maintenance is responsible for the aircraft being in that status. OPC, FMC, and PMC also fall under the general category of Mission Capable (MC).
- Discrepancy status--For each aircraft there may exist anywhere from zero to dozens of outstanding discrepancies. For each discrepancy, the system needs the Work Unit Code, Malfunction Code, When Discovered Code and the status of where in the repair cycle the discrepancy is; in work(IW), awaiting maintenance(AWM), or awaiting parts(AWP).

- For every outstanding supply requisition, the system would require the stock number, part number and supply status with estimated delivery time.
- Aircraft Time Since New (TSN)--Aircraft TSN is the number of hours an aircraft has accumulated since it was accepted from the manufacturer by the Department of the Navy. Many of the various preventative maintenance inspections are scheduled based on aircraft TSN. When manufactured, every type of aircraft is assigned an operational life and when the TSN is equal to the operational life, the aircraft is either given an extension, inducted into a service life extension program or stricken from the inventory.
- Engine Time Since New (TSN)--Engine TSN is similar to the aircraft TSN in every way. It is used to monitor the engine as a whole and also the components such as compressor and turbine disks which are installed as part of the engine.
- Total Catapult Launches in Life--Due to the extraordinary stress placed on aircraft during the catapult launch sequence, those components which play a significant role must be monitored, removed and inspected at periodic intervals. The airframe in its entirety is also limited in the number of catapult launches it may withstand in its operational life. All launch gear components are monitored in terms of Total Catapult Launches in Life.
- Total Arrested Landings in Life--As in catapult launch gear, all arresting gear must be monitored and inspected periodically. Because the arresting gear is so important extensions are generally not sought or approved.
- Date Last Flown--This date is important for two reasons. The primary reason is to ensure that an aircraft which has not flown in a significant period gets visibility when it approaches thirty days without a flight. Such aircraft, at the thirty day point must be reported first to the functional wing, at the 45 day point the type commander and at sixty days it enters special interest aircraft (SPINTAC) status. When an aircraft is reported in SPINTAC, the general consensus is that maintenance managers have failed to do a good job. Consequently, almost every effort must be expended to prevent a sixty day period without a flight.
- Phase Inspection Due--This will indicate which aircraft phase is due next (A, B....etc.). This is valuable information in that each particular phase inspection requires different levels of planning and support. For instance, a phase A may require the aircraft to be off the landing gear, which cues the MMCO to check out a set of jacks from the air station. On the other hand a phase B may require leading edge slats in the extended position.
- Phase Due Time--This figure, in flight hours is the aircraft TSN at which the next phase inspection is due. Given an average flight time and projected number of flights, the MMCO can approximate when the aircraft will become unavailable for flight operations.

- Special Inspections Due--Special inspections occur at frequent intervals on an aircraft. Some special inspections are quite simple, while others can entail a significant amount of time, labor and material. Frequently occurring special inspections such as 28, 56 and 210 day inspections are scheduled to ensure that aircraft will not all come due at the same time. Other special inspections such as hourly or cyclic ones are difficult to schedule because they depend on the operational tempo of the squadron. It is important that these inspections, all of them get visibility within the expert system.
- Flight Schedule Committments--The daily flight schedule identifies each flight by an event number and a take-off time. It further specifies the aircraft configuration required for the specific mission.
- Support Equipment/Precision Measuring Equipment--The status of all equipment needed to test and troubleshoot outstanding discrepancies.

3. Projected facts

Any item relating to or impacting future maintenance efforts. Scheduled shipboard operation, field carrier landing practice and preventative maintenance schedules. Additionally deadlines for Technical Directive Incorporation or Special Interest Aircraft Reporting may be included. Squadrons could easily maintain this data in a local database which could be queried by the expert system which could in turn update the database.

- Period End Date(PED)--The period end date is established when an aircraft commences a new service period, either when newly received or following Standard Depot Level Maintenance (SDLM). When an aircraft reaches its period end date, it must either get an extension on that life or commence another scheduled overhaul.
- Aircraft Service Period Adjustment(ASPA) Due Date--An ASPA inspection is conducted on an aircraft about six months prior to its PED to determine its suitability for a one year extension of its PED. This inspection involves a major effort by the squadron maintenance department to open up the aircraft for inspection by a depot level field team. Additionally, the aircraft is lost to the flight schedule for a number of days.
- Phase Inspection Due--Similar to the data contained in the current facts section above, however this would be a long term outlook for a complete phase cycle rather than just the next phase inspection.
- Special Inspection Due Date--Similar to the data contained in the current facts section above, however this would identify every special inspection and its due date, rather than just the inspections due in the near future.

- Technical Directives (TD's) Outstanding--This would be complete list of all TD's outstanding against the aircraft, engines and components. It would include Airframe Changes (AFC's), Airframe Bulletins (AFB's), Powerplant Changes (PPC's), Avionic Changes (AVC's) and so on.
- Scheduled Removal Components(SRC's)--SRC's are components designated by Commander, Naval Air Systems Command as planned removal/replacement items. At specified intervals, these components must be removed from the aircraft or end item and sent to an repair facility for inspection, repair or rework. A naval aircraft may have from several dozen to hundreds of such components installed.

B. RULE BASE

The other part of the knowledge base is the rule base. Here, the heuristics used by the domain expert in manipulating the fact base are placed into the expert system as rules. Ideally, each rule stands on its own with an explicitly stated meaning. A rule's inputs are its premise conditions. When input values are tested against a rule's premise conditions, the rule either produces a conclusion or it is set aside. Much like a function in conventional programming, the desired output is an inference.

In NEXPERT Object®, rules represent relations between objects, heuristics and procedural knowledge. They have three basic parts:

- Left-hand side conditions
- The hypothesis which is a boolean slot
- Right hand side actions

The conditions represent a series of tests to determine whether or not the hypothesis is true. If all of the conditions are true then the hypothesis is set to true and the right hand side actions are executed.

A rule's value depends on its left hand side (LHS) conditions:

- If no attempt has been made to evaluate the LHS conditions then the rule will be unknown

- If NEXPERT® evaluates all of the LHS conditions to True, then the rule is set to True as well
- If NEXPERT® has tried to evaluate the LHS conditions, but could not determine the value of at least one condition then the rule will be set to Not known
- If NEXPERT® evaluates the LHS conditions and one of them is False, then the rule will be set to False as well.

Where policies are clearly stated, they can readily and easily be translated into a knowledge representation schema. A policy which specifies that SPINTAC aircraft may not be cannibalized can simply be translated to, "If aircraft is SPINTAC, then cannibalization is forbidden." With the multitude of instructions, regulations and policies represented as rules, they are in a clear, comprehensible form. This could be easily modified through the user interface as changes or updates are received. A small sample of the regulations being discussed are listed below.

- SPINTAC--When an aircraft enters SPINTAC status it cannot be cannibalized without the permission of the type commander.
- Planning Factors--The planning factors for the operations and use of naval aircraft specify the readiness levels that squadrons must maintain.
- Quiet Hours--Naval Air Stations have set policy which establishes when squadrons may conduct high power engine turns which may restricts the ability to repair and troubleshoot engine related malfunctions.
- Corrosion Control--Due to concerns over the hazardous nature of certain paints and primers, many air stations have established policies on when, where and how squadrons can perform sanding, priming and painting of aircraft.
- Aircraft Wash Procedures--Aircraft washing is restricted to designated wash racks at most naval air stations to preclude hazardous chemical cleaning solvents from draining into storm drainage systems or ground water supplies.
- Heavy Weather Procedures--During certain thunderstorm conditions or when lightening is expected, fueling and ordnance transfer is restricted.
- Arm/De-arm procedures--Loading, unloading, arming and de-arming munitions is tightly controlled by air stations, type commanders as well as higher level commands. The inherent danger associated with handling live ordnance mandates strict compliance with the rules.

C. CONTROLLING GROWTH OF THE KNOWLEDGE BASE

It is not difficult to comprehend, given the complex decision making environment which we are attempting to structure, that the knowledge base for ESAAMS will eventually grow quite large. Because new rules will be added regularly, as the system is expanded and updated, it is important to take a structured and well documented approach to the maintenance of the knowledge base.

The maintainability of the knowledge base must be addressed at the very early stages of the knowledge acquisition process. One method recommended by Soloway (Bowerman, 1988, pp. 824-829) involves the use of a rule content form, similar to rule templates that guide the development of rules. In either its electronic or hard-copy form, the rule content form contains a description of a rule that includes its basic content, source, and interdependency with other rules in the knowledge base. Although maintenance of expert systems is a relatively new field of study, many developers have come to the conclusion that a completely documented system will be substantially easier to maintain than a poorly documented one.

Within the NEXPERT Object® development environment a feature known as knowledge islands is incorporated. Rules within a knowledge island share hypotheses with hypotheses or data from other rules. These islands are not implicitly developed, rather they are automatically generated by the rules themselves. This feature allows the knowledge base to be modularized, separating appropriate chunks of knowledge into different knowledge islands and processing them accordingly.

These two techniques should both be applied in the case of ESAAMS development. A well-structured and documented knowledge base would benefit largely the project as a whole. Improvements to the system over the long term would be significantly less complex. The knowledge island concept would enable end users or local commanders to implement additional policies without greatly affecting the maintainability or integrity of the system as a whole.

D. VALIDATING THE KNOWLEDGE BASE

The mass of information, data and rules which accumulate in the knowledge base over months and years of development is of little value unless the knowledge is accurate and free of contradictions. Although there will almost always be situations which occur at the limit which the system will be unable to handle, many of these can be identified through exception handling rules or through human oversight. As with a conventional software project it is advisable to test and validate the system as it is being built, rather than waiting until the system is complete.

Rule validation should begin when the very first rule set is developed. Every time new rules are added or old rules updated, the system must be checked for contradictions in processing logic and by the domain expert for flaws in reasoning. Knowledge validation should be a continual process occurring in lockstep with each step of the knowledge acquisition.

Knowledge base errors may be more difficult to find, however they are relatively easy to correct. They come in multiple forms, from typing mistakes to referring to wrong variables or using ineffective inference strategies. Bowerman (1988, p.275) concludes that a good, strong systems-analysis approach will usually turn up the sources of the problems in a reasonable time. He further states that:

> ...the most difficult expert system testing problems can arise in assigning certainty values to data and reliability ratings for rules. There may not be any "errors" in the methodology used, but the inference chains still may not produce the desired results.

He recommends a trial and error approach to correct these flaws. By manipulating the certainty values and reliability ratings the desired outcomes can be arranged. Although difficult at first, with practice it becomes easier or even intuitive.

VI. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

It was not long ago that the development of an expert system application the size of ESAAMS would not be considered feasible. Successfully deployed expert systems generally consisted of knowledge bases having less than 100 rules and were able to function well only in the most rigid domain. Improvements in technology, development techniques and experience with knowledge acquisition procedures is rapidly diminishing the difficulties of working with large knowledge bases and opening up expert system technology to a wide variety of applications.

In conclusion, one sees that by taking a structured approach to knowledge acquisition, the development of large knowledge bases becomes significantly less risky and much more productive. As in any other large problem, decomposition is the key to success. By breaking down the knowledge domain into manageable chunks, knowledge engineers will be able to address specific areas in great depth with multiple domain experts and combine them within an expert system shell. Many expert system shells have companion knowledge acquisition software which simplifies the task of converting knowledge to code.

The knowledge base provided within this thesis is barely a scratch in the surface and usable only in the most rudimentary of prototypes. No doubt about it, ESAAMS presents a challenging domain to the knowledge engineering team. The knowledge base is easily modularized however and easily tailored to situations which present themselves to organizational maintenance organizations.

In the final analysis, knowledge acquisition although time consuming, poses no obstacle to continued development of the Expert System for Aircraft Maintenance Scheduling.

B. RECOMMENDATIONS

Originally proposed in 1985, ESAAMS was to designed to utilize the data, processor and input/output devices installed as part of the installed NALCOMIS system. With the future of NALCOMIS uncertain and the extent of its impact particularly on the organizational maintenance activity in question, many of the basic assumptions that underlie the original proposal are no longer valid. Although the maintenance desk is a "target rich" environment for expert system applications, significant benefits would be gained by taking a step backward to see what is really to be expected or desired from ESAAMS. The following recommendations are offered to facilitate further development of the ESAAMS project.

1. Requirements Analysis

As in any software development project, it is important that the end-user be called upon to define the requirements for the proposed system. I would suggest that a survey of a representative sample of potential domain experts be conducted to determine what they would like to see implemented in the area of both MIS's and expert systems. The resultant "wish list" could then be translated to a valid requirements specification, from which potential expert system applications could be generated.

For each specific potential expert system application, the following issues should be addressed. (Walters and Nielson, 1985, p. 53).

- Development resources--hardware, software, knowledge engineers, domain experts, calendar time, overall cost
- Functional capabilities--logical functions that the system is to offer to the user, the breadth of the domain within which the system is to operate
- Operational environment--number of users, number of different locations, cost per delivery vehicle, operating cost, processing speed, integration with current user working environment and procedures, integration with current computing systems
- User interface--text, graphical, menu, natural language, audio
- Information sources--user input, central data base, real-time sensors
- System outputs--text output to user, graphical output to user, audio output, real-time output to other devices, updates to data bases

Given this information it would be significantly easier to design and build an expert system or set of expert systems.

2. Phased Implementation

By standards in industry, naval aviation maintenance has not yet entered the information age to any significant degree. At the organizational level all documentation, status and planning is done on hard copy VIDS/MAF. As proposed, ESAAMS counted heavily on input from the Naval Aviation Logistics Command Management Information System. (McCaffrey, 1985, p. 114) As with many DOD software projects, development of NALCOMIS has fallen behind schedule and it has not yet been deployed to any organizational maintenance squadrons. As a result, information which was to be provided to the expert system by NALCOMIS, must be obtained from other sources. The only current resemblance to a management information system at the squadron level, is what end-users themselves have developed using standard commercial software packages. Although many of the programs serve the activities well, documentation is generally poor to non-existant, making them difficult to integrate with ESAAMS.

A potential solution to this problem is to implement a program similar to the Organizational Activity Strategic Information System (OASIS). (Chase, 1990) Such a concept which advocates the implementation of a squadron information system in modules rather than in a complete package deserves consideration for many reasons. One of the most significant reasons is to overcome some resistance to automation which has developed as a backlash to the unfulfilled promises of NALCOMIS and other locally produced software applications. Starting small, an easily produced module could be produced to fulfill a need identified by squadron maintainers. With continued successful implementation of modules, "champions" of the technology will emerge. Ultimately, as suggested ESAAMS could emerge as a module or as several modules within the OASIS system.

Taking the modularization concept one step further, ESAAMS itself could easily be broken down into several modules which would enable the system to be constructed over a period of time making use of the many advantages of rapid prototyping. Modules could be developed for dealing with scheduled maintenance, airframe fatigue monitoring and component configuration control. A module could also be constructed to act as a diagnostic system to troubleshoot aircraft discrepancies. Such a system has already been demonstrated in the U. S. Air Force. (Ferguson, 1983) A diagnostic module would greatly simplify the further development of a module to schedule corrective maintenance. Modules could be constructed to enable end users to tailor the system to function differently under various operating environments. For instance there could be modules for shipboard, shorebased, cold weather and hot weather operations.

3. OMA Management Information System

The tempo of operations at the organizational level is something that must be experienced, to be believed. It should not come as a surprise that the mounds of paperwork required to monitor aircraft material condition, at times take a back seat to accomplishing the mission. The sad truth is that the maintenance controllers are being saddled with increasing requirements for data, are being tasked with monitoring the life cycle of hundreds of components per aircraft and have been given no demonstrable MIS to assist them. The requirements for such an MIS are easily defined, and an off the shelf integrated package could likely satisfy a system design specification. Every squadron has developed its own solution, however, as with many other applications built by end-users, documentation is non-existant, and shortly after the developers transfer, the program falls into disuse. It is imperative that an MIS such as OASIS (Chase,1990) be rapidly developed, standardized and deployed to organizational maintenance activities.

LIST OF REFERENCES

Bowerman, Robert G. and David E. Glover, *Putting Expert Systems into Practice*, Van Nostrand Reinhold, New York, 1988.

Burpo, John D., *The Feasability of Using the Naval Aviation Logistics Data Analysis (NALDA) Databases for the Expert System Advisor for Aircraft Maintenance Scheduling (ESAAMS)*, Master's Thesis, Naval Postgraduate School, Monterey, CA, December 1990.

Casey, Jeff T., Gettys, Charles F., Pliske, Rebecca M.,Mehle, Tom, "A Partition of Small Group Predecision Performance into Informational and Social Components," *Organizational Behavior and Human Performance,* v. 34, August 1984.

Chase, John H. Jr., *A Naval Aviation Maintenance Organizational Activity Strategic Information System (OASIS)*, Master's Thesis, Naval Postgraduate School, Monterey, CA, March 1990.

Feigenbaum, Edward A., *The Rise of the Expert Company: How Visionary Companies are Using Artificial Intelligence to Achieve Higher Productivity and Profits*, Vintage Books, New York, 1988.

Wright-Patterson AFB Human Resources Laboratory, *Use of Arificial Intelligence Methods for Avionics Maintenance, Training, and Diagnostics.* , by Gerald R. Ferguson, January 1983.

Ferguson, Gerald R., *Aircraft Maintenance Expert Systems*, Master's Thesis, Air Force Institute of Technology, Wright Patterson Air Force Base, OH, November 1983.

Follett, Ria and Tony Stevens, "Consultative Expert Systems:a Maintenance Mangagement Application," in Quinlan, J. Ross, (editor), *Applications of Expert Systems*, Addison-Wesley, New York, 1987.

Hammond, Peter, "Representation of DHSS Regulations as a Logic Program," in G. Mitra (editor), *Computer Assisted Decision Making*, Elsvier Science Publishers, Amsterdam, 1986.

Hoffman, T. L., "The Problem of Extracting the Knowledge of Experts from the Perspective of Experimental Psychology", *AI Magazine* 8:2 Summer 1987.

McCaffrey, Martin J., *The Feasibility of Implementing an Expert System for Aircraft Maintenance Discrepancy Scheduling with the Naval Aviation Logistics Command Management Information System (NALCOMIS)*, Master's Thesis, Naval Postgraduate School, Monterey, CA, September 1985.

McGraw, Karen and Seale, M., “Multiple Expert Knowledge Acquisition Methodology: MEKAM.” *Proceedings of the Third Austrailian Conference on Applications of Expert Systems*, The New South Wales Institute of Technology, Sydney, May 1987.

McGraw, Karen L. , and Harbison-Briggs,Karan , *Knowledge Acquisition, Principles and Guidelines*, Prentice-Hall, Englewood Cliffs, New Jersey, 1989.

Merritt, Dennis, “Forward Chaining in Prolog,” *AI Expert*, v.7 November 1986.

Minsky, Marvin., “A Framework for Representing Knowledge,” in Haugeland, John (editor), *Mind Design*, MIT Press, 1985.

Nilsson, N. J., *Problem Solving Methods in Artificial Intelligence*, New York, McGraw Hill, 1971.

Prerau, D., ”Knowledge Acquisition in the Development of a Large Expert System,” *AI Magazine*. v.8:2, 1987.

Prerau, David S., *Developing and Managing Expert Systems: Proven Techniques for Business and Industry*, Addison-Wesley, Reading MA, 1990.

Quillian, M. R., “Semantic Memory,” in Minsky, Marvin, (editor), *Semantic Information Processing*, MIT Press, 1968.

Rychener, M. D., *Production Systems as a Programming Language for Artificial Intelligence Applications*. Computer Science Department, Carnegie-Mellon University, 1979.

Sawyer, Brian, and Dennis L. Foster, *Programming Expert Systems in Pascal*, John Wiley & Sons, New York, 1986.

Shortliffe, E. H., *Computer-based Medical Consultations: MYCIN*, North Holland, 1976.

Simmons, R. F. and Slocum,J., “Generating English Discourse from Semantic Networks”, *Communications of the ACM*, Vol 15, 1972.

Soloway, E. J., Banchant and K. Jensen, “Assessing the maintainability of XCON-in-RIME: Coping With the Problems of a Very Large Rule-Base.” *Proceedings of AAAI-87*, v. 2, 1987.

Walters, John R., and Nielsen, Norman R., *Crafting Knowledge Based Systems, Expert Systems Made Realistic*, New York, John Wiley & Sons, 1988.

Waterman, Donald A., "A Guide to Expert Systems, Addison Wesley," 1986.

Wolfgram, Deborah D. and Teresa J. Dear and Craig S. Galbraith, *Expert Systems for the Technical Professional*, New York, John Wiley & Sons, 1987.

BIBLIOGRAPHY

Feigenbaum, Edward A. and McCorduck, Pamela, *The Fifth Generation: Artificial Intelligence and Japan's Computer Challenge to the World*, Addison-Wesley Publishing Co.,1983.

Gruber, Thomas R.,*The Acquisition of Strategic Knowledge*, Academic Press, Boston, 1989

Kearney, Michele, "Making Knowledge Engineering Productive," *AI Expert*, v.5, pp. 46-51, July 90.

Keyes, Jessica, "Where's the "Expert" in Expert Systems?,"*AI Expert*, v.5, pp. 61-64, March 90.

Keyes, Jessica, "Why Expert Systems Fail," *AI Expert*, v. 4, pp. 50-53, November 89.

Pedersen, Ken, "Well Structured Knowledge Bases: Part 1," *AI Expert*, v. 4, pp. 44-55, April 89.

Pedersen, Ken, "Well Structured Knowledge Bases: Part 2," *AI Expert,* v. 4, pp. 45-50, July 89.

Pedersen, Ken, "Well Structured Knowledge Bases: Part 3," *AI Expert*, v. 4, pp. 37-41, November 89.

Philip, George C., and Schultz, Hilbert K.,"What's Happening With Expert Systems?," *AI Expert*, v.5, pp. 57-59, November 1990.

Rasmus, Daniel W., "The Manufacturing Mind:Where AI Meets the Shop Floor," *PC AI*, v. 3, pp24-27, November/December 1989.

Surko, Pam, "Tips for Knowledge Acquisition: Things They May Not Have Taught You in School," *PC AI*, v.3, pp 14-18, May/June 1989.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145	2
2.	Library, Code 52 Naval Postgraduate School Monterey, California 93943-5100	2
3.	Commanding Officer Naval Aviation Maintenance Office Naval Air Station Patuxent River, MD 20670-5106	2
4.	LCDR Mark H. Stone CVW-2 FPO San Francisco, CA 96601-4401	2
5.	LCDR John Burpo COMNAVAIRSYSCOM (PMA-205) Washington DC 20361	1
6.	Professor M. J. McCaffrey Department of Administrative Sciences (Code ASMF) Naval Postgraduate School Monterey, CA 93943	4