NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A Proposal for the Transfer of a Large Force Management Expert System (FRESH) from the CINCPACFLT Command Center to the CINCLANTFLT Command Center

by

Craig Blaine Luigart

March 1988

Thesis Advisor: Major John B. Isett

Approved for public release; distribution unlimited.

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION
Unclassified

1b RESTRICTIVE MARKINGS

2a SECURITY CLASSIFICATION AUTHORITY

2b DECLASSIFICATION / DOWNGRADING SCHEDULE

3 DISTRIBUTION / AVAILABILITY OF REPORT
Approved for public release;
distribution is unlimited

4 PERFORMING ORGANIZATION REPORT NUMBER(S)

5 MONITORING ORGANIZATION REPORT NUMBER(S)

6a NAME OF PERFORMING ORGANIZATION
Naval Postgraduate School

6b OFFICE SYMBOL *(If applicable)*
Code 52Is

7a NAME OF MONITORING ORGANIZATION
Naval Postgraduate School

6c. ADDRESS *(City, State, and ZIP Code)*
Monterey, California 93943-5000

7b. ADDRESS *(City, State, and ZIP Code)*
Monterey, California 93943-5000

8a NAME OF FUNDING / SPONSORING ORGANIZATION

8b 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)*
A PROPOSAL FOR THE TRANSFER OF A LARGE FORCE MANAGEMENT EXPERT SYSTEM(FRESH) FROM THE CINCPACFLT COMMAND CENTER TO THE CINCLANTFLT COMMAND CENTER

12 PERSONAL AUTHOR(S)
Luigart, Craig Blaine

13a. TYPE OF REPORT
Master's Thesis

13b TIME COVERED
FROM ________ TO ________

14. DATE OF REPORT *(Year, Month, Day)*
1988 March

15 PAGE COUNT
96

16 SUPPLEMENTARY NOTATION
The views express 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	SUB-GROUP

18 SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
Force Requirements Expert System(FRESH), FCCBMP, Expert Systems, Knowlegdge Engineering, Knowledge Acquisition, Transfer of expert systems, Artifical Intelligence

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

The thesis investigates the transferability of an existing large expert system, FRESH, from its current arena of employment, the Command Center of the CINCPACFLT, to the Fleet Command Center of the CINCLANTFLT. The research is limited to the rules, heuristics and encoded knowledge used by the FRESH system and does cover interface issues. A literary review of expert system theory begins the thesis and analysis of the two fleets follows in succeeding chapters. System documentation is used to obtain a high level view of FRESH system rules, heuristics and encoded knowledge and these are then compared to Alantic fleet manual procedures gained by the use of classical knowledge engineering techniques. The environment differences developed by these comparisons between the two fleets are cited and their possible inplications on the systems transferability to the Alantic fleet explored. The Thesis concludes with a suggested method of transfer to the Alantic fleet in light of their lack of expirence with automated scheduling systems and modifications to the existing system which will be required to allow its use in the Alantic.

20 DISTRIBUTION / AVAILABILITY OF ABSTRACT
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS

21 ABSTRACT SECURITY CLASSIFICATION
Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL
Prof. John B. Isett

22b TELEPHONE *(Include Area Code)*
(408) 646-2464

22c. OFFICE SYMBOL
Code 52Is

DD FORM 1473, 84 MAR

83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

☆ U.S. Government Printing Office: 1986—606-24.

UNCLASIFIED

A Proposal for the Transfer of a Large Force Management Expert System (FRESH) from the CINCPACFLT Command Center to the CINCLANTFLT Command Center.

by

Craig Blaine Luigart
Lieutenant Commander, United States Navy
B.A., University of Louisville, 1976

Submitted in partial fulfillment of the requirements for
the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL
March 1988

ABSTRACT

The thesis investigates the transferability of an existing large expert system, FRESH, from its current arena of employment, the Fleet Command Center of the Commander-in-Chief Pacific Fleet, to the Fleet Command Center of the Commander-in-Chief Atlantic Fleet. The research is limited to the rules, heuristics and encoded knowledge used by the FRESH system and does not cover interface issues. A literary review of expert system theory begins the thesis and analysis of the two fleets follows in succeeding chapters. System documentation is used to obtain a high level view of FRESH system rules, heuristics and encoded knowledge and these are then compared to Atlantic fleet manual procedures gained by the use of classical knowledge engineering techniques. The environmental differences developed by these comparisons between the two fleets are cited and their possible implications on the systems transferability to the Atlantic fleet explored. The thesis concludes with a suggested method of transfer to the Atlantic fleet in light of their lack of experience with automated scheduling systems and modifications to the existing system which will be required to allow its use in the Atlantic.

TABLE OF CONTENTS

I. INTRODUCTION 1

- A. CONCEPT OF EXPERT SYSTEMS 1
- B. FORCE REQUIREMENTS EXPERT SYSTEM (FRESH) 3
- C. SCOPE OF THE THESIS 5
- D. LIMITATIONS 6
- E. METHOD OF RESEARCH 7
- F. ORGANIZATION OF THE THESIS 7

II. THEORY OF EXPERT SYSTEMS 9

- A. ARTIFICIAL INTELLIGENCE 9
- B. EXPERT SYSTEMS 10
    - 1. Expertise 11
    - 2. Symbol Manipulation 11
    - 3. General Problem Solving Ability in a Domain 12
    - 4. Complexity and Difficulty 12
    - 5. Reformulation 13
    - 6. Abilities Requiring Reasoning About Self 14
    - 7. Task 14
- C. EXPERT SYSTEM ARCHITECTURE 15
    - 1. The Knowledge Base 16
    - 2. The Inference Engine 17
    - 3. The Knowledge Acquisition Module 18
    - 4. The Explanatory Interface 18
- D. INTELLIGENT PROBLEM SOLVING 19
- E. KNOWLEDGE ACQUISITION 20
    - 1. Hand-crafted Mode 22
    - 2. Knowledge Engineering Mode 22
    - 3. Expert Conversant Mode 23

4. Data Induction Mode 23
5. Text Understanding Mode 23

F. KNOWLEDGE ENGINEERING 24

G. TRANSPORTABILITY 28

H. DIVERSITY OF EXPERT SYSTEMS 30

1. Surveillance Integration Automation Project (SIAP) 30
2. AIRPLAN 30
3. Knowledge Based System (KNOBS) 31
4. Rule-Based Retrieval of Information by Computer (RUBRIC) 31
5. Emergency Procedures Expert System (EPES) 32

III. CINCPACFLT FRESH DEVELOPMENT 33

A. FRESH WITHIN THE EXPERT SYSTEM FRAMEWORK 34

1. Expertise 34
2. Symbol Manipulation 35
3. General Problem Solving Ability in a Domain 35
4. Complexity and Difficulty 35
5. Reformulation 36
6. Abilities Requiring Reasoning About Self 36
7. Task 36

B. PROTOTYPING METHODOLOGY 36

C. GOALS OF THE FRESH SYSTEM 40

D. FRESH KNOWLEDGE 42

1. Rules, Constraints and Heuristics 43
2. Knowledge Bases 45

E. THE OPERATIONAL FRESH PROTOTYPE 47

1. FRESH Knowledge Engineering and Validation 47
2. Current Uses and Limitations of FRESH 49

F. SUMMARY 51

IV. CINCLANTFLT ANALYSIS 53

A. THE ATLANTIC FLEET SCHEDULING PROCESS 54

1. The Manual Scheduling Process 54
2. CINCLANT Organizational Control 55

B. IDENTIFIED DIFFERENCES IN THE ATLANTIC ENVIRONMENT 57

1. Basic Differences 58
2. The Impact of North Atlantic Treaty Organization 60
3. Additional Reasoning and Knowledge Desired 61

C. SUMMARY 64

V. CONCLUSIONS 65

REFERENCES 68

APPENDIX A 71

APPENDIX B 79

INITIAL DISTRIBUTION LIST 84

LIST OF FIGURES

Figure 2.1 Intelligent Human Problem Solving 20
Figure 2.2 The Knowledge Engineering Approach 24
Figure 2.3 Major Categories of Knowledge 26
Figure 3.1 IEEE Prototyping Paradigm 37
Figure 3.2 FRESH Functional Requirements 41
Figure 3.3 Generalized Scheduling Algorithm 42
Figure 3.4 Platforms Hierarchy 44
Figure 3.5 FRESH Schema Hierarchies 46

ACKNOWLEDGMENTS

As I write this essential page of the thesis, I am thankful that at least this one page does not require the omnipotent judgement of my thesis advisor and for this one brief moment the formal style requirements of an academic work may be dispensed with. While this all might seem to diminish the significance of this page, in my humble view, this page represents the great indebtedness I have to others for the body of this work, and therefore, in most ways this section is the most important.

At the outset, I would like to thank the CINCLANTFLT and CINCPACFLT staffs, as well as the FRESH contractor development team of Texas Instruments Inc. and btg, Inc. (for those readers kind enough to read this page, the lower case in the text for "btg" is intentional—that's the way they print it!) for all their support and help on this project. Without their unselfish dedication of time, this research would not have been possible. I would particularly like to thank Commander Bob Rankin, USN, and Lieutenant Commander Bill Christianson, USN, for their help with CINCLANT knowledge engineering. I would most especially like to thank Lieutenant Commander Mike McNeil, USN, and Lieutenant Bruce Drees, USN, without whose personal efforts this work would assuredly never been completed. I am deeply indebted to Major John Isett, USAF, Ph.D., who provided the required guidance and constant focus on the task at hand and who also assured that I learned something during the ordeal, and to Major Thomas Brown, USAF, whose sharp eyes have more than once proven the GIGO rule of computer science can also be applied to the academic works of a Naval Postgraduate School graduate student.

Last, the most important group of people I need to thank, my family. To Kristen my daughter, who, though she never really understood why, accepted my need to spend copious amounts of time on the Macintosh™ writing, and whose unquestionable love is a constant strength—may your dreams be always bright. Dad, I know you've been with me—thanks. A special thank you to my mother, Betty Luigart Snowden, for all the teachings and guidance that brought me to adulthood, as well as my parents-in-law Morris and Adelyn Trammell (really my second parents), who provided me with my strongest asset, my strength and confidence, my wife Connie.

Connie has been and remains my best friend and confidant. Her faith in my ability never faltered, even when I had lost mine. Her belief in me is a rock on which I lean. A Naval Aviator is expected to be self-assured and in control I was once taught in flight school. I only thank God I was given my wife to provide me with the needed faith and guidance to convince me the goal was worth the means and that I possessed the talent and intelligence to walk the seemingly impossible road ahead of me.

I. INTRODUCTION

This thesis will investigate and propose modifications of the instantiated knowledge base and production rule set of a large expert system, Force Requirements Expert System (FRESH), in order to facilitate its transfer to the Commander-in-Chief Atlantic Fleet (CINCLANTFLT) Norfolk, Virginia from its current application site, the Pacific Fleet Command Center for the Commander-in-Chief Pacific Fleet (CINCPACFLT) Pearl Harbor, Hawaii.

A. CONCEPT OF EXPERT SYSTEMS

Artificial Intelligence and its subset discipline of expert systems has enjoyed considerable development and research during the past two decades. Estimates of growth for the next four years exceed a four fold increase in the field. [Wang 87:6] Systems can be found in development in almost all areas where human knowledge and decision making is considered the critical element for success or failure of a task. These systems have already attained expert levels in many application areas [Chorafas 87:203-204]:

BACON - analysis and synthesis for room planning.

BLAH - analysis for income tax advice.

EMES - management of power allocation of electrical components in spacecraft.

IDT - Digital Electronics Corporation's intelligent diagnostic tool.

LUNAR - analysis of Apollo II lunar rock samples.

These few examples illustrate the range of application (several specific expert systems used by DOD will be more fully discussed in Chapter Two). Some of the benefits gained so far by expert systems technology as cited by Hayes-Roth include: PROSPECTOR: discovered molybdenum deposit...ultimate value will probably exceed $100,000,000.00, R1: configures customer requests for VAX computers, despite the fact resident experts thought it could not be done. [Hayes-Roth 83:6]

Successes in this computing discipline naturally brought about an interest in possible applications to problems addressed in the Department of Defense (DOD). With ever increasing fiscal pressure on today's military manager, it is hoped that many of the financial benefits that have been gained for the private sector through the implementation of expert systems might also be achieved in DOD. Harmon states: "This new technology will make it possible to develop quick, pragmatic answers for a wide range of problems that currently defy effective solutions" [Harmon 85:1]. There are advantages to be gained from this technology if it is applied to the increasingly complex problems of national defense.

Expert systems embody the knowledge of an "expert". This allows the average user to be supported by that embedded expertise to develop better than average solutions to complex problems. DOD expertise in a field is often a short-lived asset due to the frequent transfer, resignation and retirement of military personnel. A system that captures this expertise, an expert system, and can reason on that knowledge in efficient and effective ways may result in savings in time and money. Additionally, the solutions presented may have a higher degree of consistency than otherwise might be expected of human managers.

DOD to date has implemented a variety of expert systems. For example, intelligent flight simulators for the training of aviators reduces the cost to train through reductions of required in-aircraft training hours. Tactical decision aids have been built to support the battle front commander. The expert system and its potential benefits are rapidly becoming the focus of today's military systems development.

The effectiveness and efficiency of expert systems development depends on careful planning and proper problem selection. The need to provide systems that are cost effective will require designers to develop systems that are both "experts" in their domain of knowledge and generic enough to allow transfer to other similar problems and a variety of hardware.

With these concepts in mind, CINCPACFLT foresaw a need to capture its scheduling expertise, as well as provide capabilities for rapid evaluation of proposed scheduling in light of evolving world events. This expert system, the FRESH system, is currently in prototype development at CINCPACFLT. Additionally, its transfer after development is proposed by the Fleet Commanders for Atlantic Fleet use.

B. FORCE REQUIREMENTS EXPERT SYSTEM (FRESH)

FRESH is a DOD expert system contracted for development by Defense Advanced Research Projects Agency (DARPA) and Space and Naval Warfare Systems Command (SPAWAR) for CINCPACFLT. The system was developed using rapid prototyping methods by Texas Instruments Corporation (TI) and btg®, Incorporated (BTG). The system is designed to assist in the scheduling and monitoring of battle force units at the Commander-in-Chief (CINC) level and is installed in the CINCPACFLT Command Center (PFCC), Pearl Harbor, Hawaii.

Specifically the system prototype is currently used for three primary functions:

(1) Recognize whether a force deficiency exists and alert the user.

(2) When requested by the user, recommend actions to correct a force deficiency.

(3) Develop fuel utilization figures for proposed redirection of units indicated by function (2).

Briefly, FRESH monitors incoming automated reports of an individual units Combat Readiness Overall (CROVL or C-rating) and alerts the command center when the units C-rating has fallen below specified levels that might impact fleet performance. FRESH then proposes alternate unit tasking and replacement—this is an extremely complex operation requiring *expert* judgement.

Under FRESH, this evaluation and replacement planning has been greatly expedited. Command Center estimates indicate a reduction by factors ranging from one-fourth to one-twelth in both the needed manpower and time over previous manual methods. It is this researcher's belief that the transfer of this technology to the Atlantic Fleet Command Center (AFCC) for use in that theater of operations should provide similar benefits and is the primary motivation for this research.

Warn states, the portability and reusability of software is an important concern to large scale software purchasers like the DOD [Warn 86:409]. To date the transfer of an expert system to other application sites has been shown to be difficult even when the problems at all sites are similar! The opinion of the knowledge engineers with the TI developmental team for FRESH is that expert systems are commonly developed to handle a single and very specific problem using site unique data. A complete redesign is often required for system transfer to other sites.

The encompassing question of this thesis, then, is what is required to transfer the FRESH system to the AFCC? Unfortunately, there is not an easy answer to this question. Just as two product divisions of the single General Motors Co. are the same but different (!), the Pacific fleet and Atlantic fleet both enjoy their own ways to do essentially the same tasks—in effect two separate navys.

It is entirely possible that the environments at CINCPACFLT and CINCLANT are dissimilar enough to prohibit or make exceedingly difficult the transfer of the system. Therefore, the transfer of the expert system FRESH to another application site requires study to identify the applicability, feasibility, and effort involved.

C. SCOPE OF THE THESIS

This thesis, then, intends to explore and identify those differences as they apply specifically to the production rules and instantiated knowledge base of the existing PACFLT FRESH prototype. First, is FRESH applicable to the LANTFLT and is its transfer even feasible? If so, what changes are required, if any, to the knowledge and reasoning of FRESH to allow its benefits to be enjoyed by the Atlantic Commander? Are Atlantic and Pacific environments similar enough to allow the transfer of a developed expert system in light of the historical difficulties facing an expert system transfer?

To answer these questions, the system's current reasoning must be analyzed. Next the current Atlantic fleet manual methods will be studied and compared to those FRESH reasoning equivalents for possible alterations and modification to the FRESH system.

At the outset of this study, no formal intent to transfer the FRESH prototype to the LANTFLT existed. It was anticipated that some resistance to the introduction of FRESH to the Atlantic Fleet might be encountered due to inherent differences between the fleets. These differences might make the transfer of FRESH unsuitable politically or the system may be seen as unnecessary by the LANTFLT command. This did not materialize and as of mid-year 1987 formal initiatives exist, fiscal constraints allowing, between the CINCPAC and CINCLANT commands to transfer the system. While this diminishes the possibility of resistance to this research from CINCLANTFLT, it increased the possibility of TI and BTG withholding information over which they feel they have proprietary ownership. While this resistance did not develop, TI and BTG have been unable to provide a significant amount of requested original knowledge engineering notes used to support the original design of the schemas and rules of FRESH. As stated by TI representatives, this was due to destroyed or lost notes which occurred during turnovers within the development organization.

This research will only consider the unclassified knowledge schemas implemented in FRESH—Platform, Employment-Category, Equipment, Geographic Location, OPCON, and Battle Group as well as the published rules, constraints and heuristics found in Appendix One.

No attempt to validate the data inputs or outputs of the system will be undertaken as that is the subject of another parallel study. Further, this thesis will make no attempt to validate the efficiency or effectiveness of the current FRESH prototype.

D. LIMITATIONS

The developer has made every effort to supply all materials this researcher has requested. However, in light of the fragmented nature of the available original knowledge engineering documentation as discussed above, the thesis will be limited to the published documentation only and to an upper level view of the system. This view is presented in the FRESH Functional Design Document (FDD) and its appendices, in particular, Appendix E, the FRESH Knowledge Base Description Document (FKD).

Additional limitations occurred due to restricted travel funds for on-site visits to the AFCC, though extensive survey and telephone interviews were conducted to knowledge engineer the Atlantic Fleet environment and develop basic system requirements.

E. METHOD OF RESEARCH

This research was carried out through an investigative approach with:

(1) an initial study and analysis of the theory of expert systems.

(2) study of the problems associated with their transfer historically.

(3) an investigation of the FRESH system installed at the PFCC

(4) application of knowledge engineering methods to develop AFCC needs and system heuristics.

Interviews and study of the FRESH prototype itself were conducted at the PACFLT Command Center. CINCLANT manual methods and rules were obtained by using the following knowledge engineering methods: interview, study of existing documentation, and survey and interview of CINCLANT scheduling experts. Existing FRESH schemas and rules were compared with Atlantic Fleet's manual procedures and the results of that comparison are found in Chapter Four.

F. ORGANIZATION OF THE THESIS

Chapter Two represents the literature review for this thesis. The theory of expert systems and historical and theoretical information on the transfer of this technology to other application sites is presented. A short study in knowledge acquisition methods concludes the chapter.

Chapter Three presents the FRESH prototype and the basics of the instantiated schema, production rules, and an in-depth look at the FRESH prototype's use at CINCPAC.

Chapter Four analyzes the current schedule production method used at CINCLANT and presents differences between that approach and the current FRESH system. Further, the major data which is held by the system requiring modification is explored and the chapter concludes with a recommended transfer methodology to be used.

Chapter Five presents conclusions to the research.

II. THEORY OF EXPERT SYSTEMS

As noted in the introduction, FRESH is an expert system, a very special expert system. With FRESH, peoples lives and national security may be at risk. Existing expert systems in the private sector rarely, if ever, affect life and death decisions. Additionally, FRESH must be relied on to quickly make the right decision at the right time both in peace and war. This trait, to recommend action for the decision maker and to do so before the decision maker alone could have developed a solution, is a common goal of expert systems.

To provide the reader with a more complete understanding of the FRESH system, a better understanding of expert systems and their development is required. This chapter will discuss artificial intelligence and expert systems, expert system architecture, intelligent human problem solving, knowledge engineering, knowledge acquisition, transportability problems, and last a survey of expert system applications in DOD today.

A. ARTIFICIAL INTELLIGENCE

As defined by Chorafas, Artificial intelligence (AI) is a scientific field concerned with creating computer systems which can achieve human levels of reasoning. [Chorafas 87:42] A branch of the computer sciences discipline, artificial intelligence research exists along two major fronts. The first, to create systems that emulate natural human responses and capabilities through embedded intelligence, such as responding to environmental inputs in manners consistent with human responses. Examples are speech, computer vision, and robotics. The second area of current work is the creation of either stand alone or cooperative systems capable of reasoning in manners representative of a human expert—expert systems. FRESH is a member of this latter discipline, expert systems, and is the discipline on which this thesis concentrates.

B. EXPERT SYSTEMS

The literature takes several approaches to the formal definition of an expert system. Feigenbaum defines an expert system as:

> ...an intelligent computer program that uses knowledge and inference procedures to solve problems that are difficult enough to require significant human expertise for their solution. Knowledge necessary to perform at such a level, plus the inference procedures used, can be thought of as a model of the expertise of the best practitioners of the field.
>
> The knowledge of an expert system consists of facts and heuristics. The "facts" constitute a body of information that is widely shared, publicly available, and generally agreed upon by most experts in a field. The "heuristics" are mostly private, little discussed rules of good judgement (rules of plausible reasoning, rules of good guessing) that characterize expert-level decision making in the field. The performance level of an expert system is primarily a function of the size and quality of a knowledge base it possesses. [Harmon 85:5]

FRESH conforms well to this definition. Through the use of embedded knowledge bases and access to a complex and extensive database, the Integrated Database (IDB), FRESH gains "facts" about the fleet. Further, possessing an instantiated set of expert heuristics drawn from the CINCPAC staff through classic knowledge engineering procedures, the system is designed to interact with the user and generate solutions to the problems it is tasked to solve. Hayes-Roth further developed the definition of expert systems identifying seven features he felt fundamental to the goals of development [Hayes-Roth 83:43-50]:

(1) Expertise
(2) Symbol Manipulation
(3) General Problem Solving Ability in a Domain
(4) Complexity and Difficulty
(5) Reformulation
(6) Abilities Requiring Reasoning About Self
(7) Task

An explanation of these seven fundamentals follows.

1. Expertise

The fundamental goal of the system is to attain the high level of performance and quality of solution that could be expected from the human expert. The system must solve the problems for which it is designed and, most essentially, the manner of the solution should come from reasonable approaches to the solution. Often the logic used to arrive at the solution is as important as the solution itself. [Hayes-Roth 83:42] Ideally, the system should emulate the human expert's thought processes, using the same rules of thumb (heuristics) and inference patterns developed over time by the human expert.

2. Symbol Manipulation

Traditional procedural or algorithmic languages do not easily represent the required logic for expert systems development. Instead a more complex language construct central to the methodology of expert systems is used—*symbol manipulation.* Humans think in the terms of highly complex symbols rather than the simple variable constructs used in conventional programming such as "X" or "Y" variable. These complex symbols (often represented in our language as a single word) are in fact only representations of a much greater association of characteristics which we mentally associate, either consciously or unconsciously, as defining the symbol. An example would be: Dog, a noun, is actually representative of a great number of associated characteristics—four-legged, hairy, mammal, barks, tail, has a name, etc., etc. These characteristics, in many cases may be other symbols in their own right.

Several expert system languages have been developed to capture these symbols, LISP and PROLOG the most notable. The power of the expert system languages is their ability to capture these "symbols" and manipulate these lists or relationships as required by the expert system and user. Most often the symbol and its relation to other symbols or objects is captured in a database or, as in FRESH, hierarchial schemas. This form of information capture will be discussed more fully in Chapter Three.

3. General Problem Solving Ability in a Domain

The expert system should be developed with all relevant knowledge about the subject that can be gained. Unlike traditional computer applications which follow strict paths of solution, expert systems infer facts and solutions from other facts and previous solutions. This provides a certain robustness to the system and the ability to search other solution paths than those which would be present with strict procedural based programming. This also includes the ability to degrade gracefully in areas where domain knowledge is insufficient. [Hayes-Roth 83:46] Therefore, the quality of the solution is based on the broad availability of relevant facts and principles as well as the completeness of the system's inference methods and implementation.

4. Complexity and Difficulty

The problem needs to be sufficiently complex to allow development as an expert system, "...certain domains do not qualify as potential arenas for expertise because they are somehow not complex enough." [Hayes-Roth 83:47]. The problem must be complex enough yet structurable to a degree to provide the basis for a methodological approach to the solution. Problems with little or no structure or pattern are not easily representable in an expert system. Simple problems often do not contain the required complexity to justify the application of this emergent technology.[1]

5. Reformulation

Reformulation for the expert system is its ability to take a problem stated or presented in one form and convert it into one more suitable to the solution paths available to it. This ability to take arbitrary problems and reformulate them into a form suitable to his heuristics is commonly found in the human expert. However, for the computer based system one must assure that the system does not attempt to fit inappropriate problems to the wrong model. [Hayes-Roth 83:47-48]

The expert system should be able to take as input the human view of the problem and transfer that information into the symbolic constructs it requires for its processing. The information must be reasonable to the problem at hand and information which the system was designed to have as input. The system should understand its reformulation limits and be designed with bounds to its inference on the information provided to it. This last feature prevents the system from inferring relationships which do not exist and therefore concluding solutions which cannot be made from the inputs.

---

[1] To a certain extent this trait implies a need for the project to provide a cost effective return for the dollar. If the project is trivial then it should not be coded; it's cost outweighs the gain. Also implied is if the project is not definable (structurable and possessing sufficient knowledge domain) the system may never produce a quality result. This then is not cost beneficial.

6. Abilities Requiring Reasoning About Self

The expert system must be able to reconstruct the paths of inference it selected in the development of a solution. The system must be able to reason about its own thought process and explain its reasoning to the user. The human expert is often required to explain how he arrived at a decision, i.e., the logic used.

7. Task

Each expert system is developed to solve the problems of a specific environment. As such, comparisons and performance measures between different expert systems (i.e.,for different domains) are essentially meaningless. When two different systems are applied to the same problems, the application tasks that they are focused on may differ, therefore, the expert systems differ. [Hayes-Roth 83:49] A given expert system is designed for a specific set of tasks, which are governed by the needs of the environment to which it is to be applied. The system should not be applied in other than those environments for which it was designed. This leads to the problem being studied: Is the Atlantic environment and scheduling problem of sufficient similarity to the Pacific's to allow the transportability of FRESH?

Hayes-Roth states no present system has fully attained all seven of these attributes, though the seven remain as the baseline for measurement of all developed systems. [Hayes-Roth 83:43] Today's systems continue to expand the limits within these seven areas. Notably, reformulation, abilities requiring reasoning about self, and general problem solving ability in a domain are subject to significant research and validation.

Expert systems are bounded by diverse demands and attributes. Their goal, though, is universal: to diagnose problems, recommend alternative solutions and strategies, offer rationale for their diagnosis and recommendations, and learn from experience by adding knowledge developed in previous solutions to their current knowledge base. [Davis MW 85:201ff]

C. EXPERT SYSTEM ARCHITECTURE

Expert systems can be developed when a problem is to some degree structurable and sufficient "expert" knowledge exists which is capturable in a computer based system. The human "expertise," called heuristics, must be representable in the expert system. Additionally, the system requires an extensive body of knowledge about a specific problem area that the expert would normally possess. Given these two are properly developed, the system should be able to develop correct solutions approaching the validity of the human expert that was modeled.

Forsyth identifies four component parts of the architecture of the expert system which capture and hold the knowledge and replicate the heuristics. Additionally, the components allow for the explanation of the result, the reasoning behind the answer [Forsyth 84:10]:

(1) The Knowledge Base.
(2) The Inference Engine.
(3) The Knowledge Acquisition Module.
(4) The Explanatory Interface.

Of these four, knowledge and inference form the common heart of all systems. [Hayes-Roth 83:90]

1. The Knowledge Base

A knowledge base contains all the facts, rules, and relations. Human decision making employs large amounts of information gained from daily life and study of a subject. Often this learning occurs in very unconscious ways. We learn to speak English by interacting with our environment at an early age, learned but not taught information that is stored away without conscious effort. Much of our learning is stored away as rules of thumb, things that worked under a given set of circumstances and those that did not. Hayes states the average human expert has ten years of experience in a field. [Hayes-85:392] The information is gained through subconscious as well as conscious learning and may not be readily recallable by the expert. The goal for the knowledge engineer is to uncover the critical knowledge and instantiate it in the expert system knowledge base. This task is a formidable one because the knowledge obtained during the knowledge acquisition process may be biased, erroneous, or incomplete. [Kessel 86:541] How this is undertaken will be discussed later in this chapter.

From the above, one may infer that to have the expert system accurately emulate the human expert, one would need all the experiences of the person's life. As this is of course impossible, today's knowledge bases are much less ambitious. [Keller 87:3] The goal then is to represent all that is essential about a given application environment in the knowledge base: facts, rules, and relations concerning the problem. Facts represent the short term information that can change rapidly. Rules and relations are the longer term information and represent how to generate new facts and assertions from what is presently known. [Forsyth 84:11] This represents the difference between a database and the knowledge base; the database for a traditional system is a collection of static facts—

they are either there or they are not. Given facts, the expert system attempts to fill in its missing knowledge through the use of inferred reasoning (inference) to complete its view of the world. Similarly, we infer the presence of a lamp when we enter a lit room on a dark night even though we may not be able to see the lamp itself.

2. The Inference Engine

This last statement leads to the second essential portion of the expert system—the inference engine. This is the portion of the system which infers new knowledge from formerly held knowledge. This is typically done through one of two reasoning methods called "forward" or "backward" chaining. [Forsyth 84:12] While both methods of reasoning are employed, Bui states backward chaining has had the most success and is the most often used for large systems. An example of the technique follows [Bui 87:ip]:

Facts known to exist:	**F, J, K.**
Rules provided to the system:	If **A** and **B** then C. If **E** then **A**. If **G** and **H** then **B**. If **F** then **E**. If **J** then **G**. If **K** then **H**.
Goal for the system:	**Find C.**

> In solving, backward chaining looks at the goal first and attempts to back out away from it to the facts provided. A strategy of beginning from a goal or expectation of what is to happen and working backwards, looking for evidence that supports or contradicts the expectation. By using the instantiated rules and heuristics to develop new facts (the defined relationships and equivalents) it attempts to prove the goal.

In the example we are given, if we can prove the existence of **A** and **B** we can prove **C**. Backing away from this rule we first search for **A** or **B** without these we back further to attempt to prove the existence **E**, **G**, and **H** which if found will prove **A** and **B**. These still not provided we back further out to attempt to prove existence of any of them (**E**, **G**, and **H**). At this level we find we can prove their existence with rules governing inference on **F**, **J**, and **K**. With these known we can then infer the condition **C** is true.

Whether the inference method works forward or backward, the system often will work with "facts" with a degree of certainty or uncertainty. Because the system is using some information with only a given degree of certainty, the system must be able to develop confidence factors to be presented in the final solution. Essentially it must develop a probability of "correctness" for its solution just as a human expert might say, "I'm 90% confident of this solution." Various schemes are employed in different systems to handle this problem, such as Bayesian and Fuzzy logic. [Forsyth 84:12] These approaches work reasonably well.

3. The Knowledge Acquisition Module

More properly this is called knowledge engineering and will be left to a later portion of this chapter. Traditionally, this is not a computer module as might be inferred by its name, but rather it is the human process of gathering data and information to be used in the system. It is sufficient to say at this point, however, that the success of the system depends directly on how well we gain human expertise and represent it in the system. A variety of approaches exist and no single one is appropriate in all situations.

4. The Explanatory Interface

Probably from the user's stand point, the single most important portion of the system is its interface. How the user should (is expected to) perceive and understand the system—its uses and limitations—in the context of the user's particular decision situation is also an important aspect of the interface architecture. [Bennett 83:224] Keen states, "the user system interface is not a 'cosmetic' issue: to the user, the interface is the system" [Keen 76:1]. Nowhere is this more true than in the expert system interface. To the non-expert, the interface, the window to the system's reasoning, must be able to remove the veil covering the logic used to develop the decision. Most skeptics—Commanding Officers especially—will require the system to prove its thinking. Expert systems must be open to interrogation and inspection. In short, a reasoning method that cannot be explained is unsatisfactory, even if it performs better than a human expert. [Forsyth 84:14]

D. INTELLIGENT PROBLEM SOLVING

Hayes-Roth identifies the basic concepts underlying the approach to intelligent human problem solving in his book *Building Expert Systems* as shown in Figure 2.1 [Hayes-Roth 83:19]. The core idea to the problem solutions is that the system, human or machine, must construct its solution selectively and efficiently from a space of alternatives, often a resource limited space representing incomplete knowledge, to determine directly the solution. Experts commonly face problems that are not algorithmic in nature and are heavily dependent on heuristics. The expert needs to search this space selectively and efficiently, identifying useful data and employing heuristics to infer new data and eventual solutions. Essentially, a timely use of currently held data to identify the potentially promising paths to a solution and the ruling out of those without promise.

Referring to Figure 2.1, the first three concepts address how to develop the primacy of knowledge that must be represented in the expert system. The last two concepts are methods to be used to aid in the refinement of the knowledge base and increase its capacity to solve more complex problems.

1. Knowledge = Facts + Beliefs + Heuristics
2. Success = Finding a good enough answer with the resource available
3. Search efficiency directly affects success
4. Aids to Efficiency:
    a. applicable, correct, and discriminating knowledge
    b. rapid elimination of "blind alleys"
    c. elimination of redundant computation
    d. increased speed of computer operation
    e. multiple, cooperative sources of knowledge
    f. reasoning at varying levels of abstraction
5. Sources of increased problem difficulty
    a. erroneous data or knowledge
    b. dynamically changing data
    c. the number of possibilities to evaluate
    d. complex procedures for ruling out possibilities

Figure 2.1 Intelligent Human Problem Solving [Hayes-Roth 83:19]

This overview of the human and machine problem solving domain infers a great need to identify and represent the relationships and knowledge used by the human expert for an expert system to function. This discipline of study and documentation of the human expert's thought, educated guesses, and knowledge is that of Knowledge Engineering (KE), the single essential foundation on which all expert systems are based.

E. KNOWLEDGE ACQUISITION

The task of knowledge acquisition is to acquire and formulate knowledge for the expert system—a significant burden, that of uncovering and formalizing the extensive knowledge and background used by the human expert in his job. Knowledge acquisition is often the bottleneck in the construction of expert systems. Schafer, author of the *Intelligent Systems Analyst,* states:

> Acquisition is one of the main obstacles to the wider implementation of expert systems technology....the process is simply not at all that well understood yet. One thing we've learned is that human experts don't generally communicate their expertise well, either to another human or to a machine. In part, this is because a good bit of their knowledge is unconscious. In fact, it can be argued quite persuasively that expertise consists of sublimated knowledge, i.e., skills about which one does not need to think. The expert simply doesn't know that he has this knowledge. Another part of the obstacle, though, can be found in the fact that there is often tremendous ego involvement in knowledge. Experts don't want to reveal all they know about a subject because their expertise is a great deal of their identity. [Schafer 88:146]

Knowledge of a domain takes many forms some of which are easily transferred if firm or fixed and algorithmic in nature. This type of knowledge is associated with traditional procedural based computer programs solving complex mathematical equations. However, as previously stated, this form of knowledge is rarely the form used by the human expert decision maker who, rather, relies on subjective, ill-codified, and partly judgmental knowledge. Expert systems with their symbolic processing of information and relationships is then more appropriate. [Hayes-Roth 83:128] This type of knowledge, which is heuristic in nature, is rarely in forms easily translated to algorithmic methods and, therefore, does not permit simple translation to a computer program. Knowledge acquisition is the process of acquiring, defining and implementing the knowledge and relationships of the human expert. As defined by Hayes-Roth:

> Knowledge acquisition is the transfer and transformation of problem-solving expertise from some knowledge source to a program. Potential sources of knowledge include human experts, textbooks, databases, and one's own experience. [Hayes-Roth 83:129]

Bui identifies five modes of knowledge acquisition representative of the various methods employed in the field today—hand-crafted, knowledge engineering, expert conversant, data induction, and text understanding mode. More specifically, each of these modes represents different methods of acquisition as follows [Bui 87:ip]:

1. Hand-crafted Mode

In this mode, the developer must first make himself an "expert" in the field to be modeled, then models himself in the system. An intense approach, considering the average time (ten years) to gain expertise in a domain. The approach may yield inconsistency and potential problems exist with the maintenance of system knowledge. EMYCIN, an expert medical diagnostics program, was developed in this manner by a physician who also had expertise in computer science and the theory of expert systems.

2. Knowledge Engineering Mode

The system to be developed is subdivided into two distinct systems: the inference engine (problem-solving knowledge) and the knowledge base. Information and expertise for representation in these two subsystems is gained through conversations and surveys with the expert. The quality of the interpersonal communication is important and directly affects the validity of the system. Poor down-loading (acquisition) of the expert's knowledge, such as misunderstanding or a lack of thoroughness, will lead to ineffective and incomplete systems.

This method of knowledge acquisition is the most widely used and will be discussed more fully later in this chapter.

3. Expert Conversant Mode

This development mode is essentially the same architecture as the knowledge engineering mode except the knowledge engineer is not used. Here a complex, computer-generated, knowledge acquisition program is used to query the expert. The expert interacts directly with the system via an intelligent interface editor to develop the required system. This method requires sophisticated interfaces and dialogue capabilities and the quality of the man machine interface is essential. This method is the beginnings of an expert system for building expert systems.

4. Data Induction Mode

This mode is very much like the above (3), but even more intelligent. It uses an induction program to build its knowledge bases from analysis of the expert. This approach uses induction techniques that derive causal relationships to define new relationships. The derived system is usually lacking in transparency (the user may be unable to understand the system's logic) and therefore, the user may be unable to understand how the system derived its answer.

5. Text Understanding Mode

As yet this method exists in theory only. Under this methodology of development, it is theorized that the system will be self-taught. It will have the capability to call on "textbooks" and other hard data sources using text understanding programs and thereby teaching itself to be the expert—the definitive expert system to develop expert systems.

As stated above, the most common approach to knowledge acquisition is that of knowledge engineering. This approach was the one used for the development of FRESH and is discussed more fully below.

F. KNOWLEDGE ENGINEERING

Knowledge engineering is by far the most widely used method of knowledge acquisition. This approach uses investigative "down-loading" of the expert's knowledge (the translation of the "expert knowledge" to the computer system) by a computer scientist or information sciences engineer commonly referred to as a Knowledge Engineer (KE). This alleviates the expert from the need to gain a complete understanding of the information sciences discipline and the necessary background that would be required for him to develop his own system. Waterman graphically illustrates the knowledge engineering approach in Figure 2.2 [Waterman 85:153].

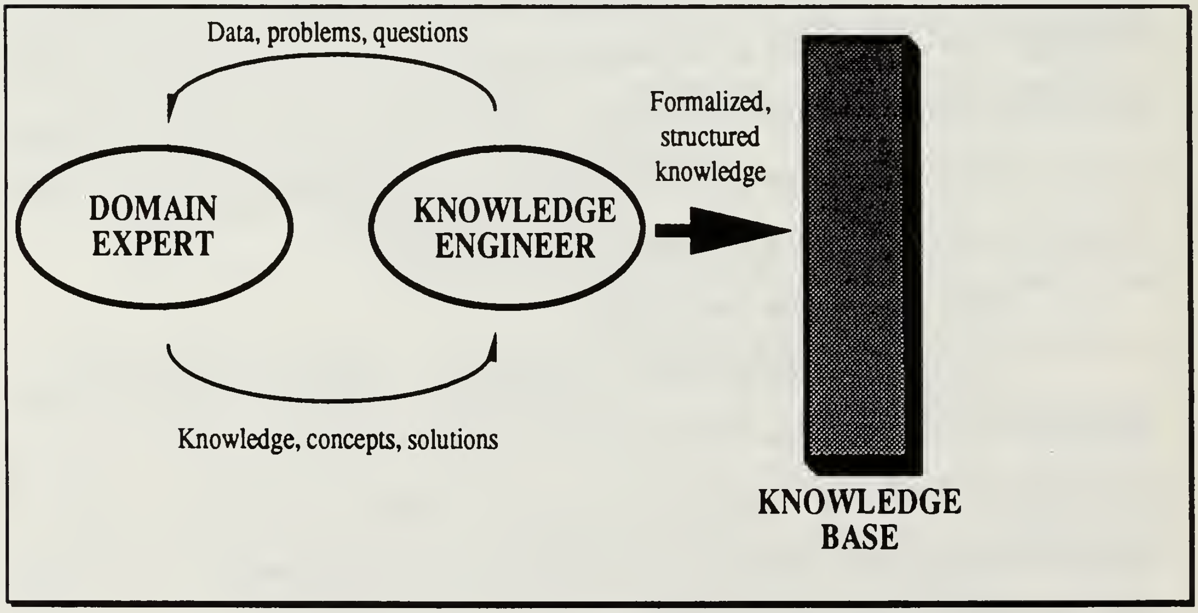

Figure 2.2 The Knowledge Engineering Approach [Waterman 85:153]

The transfer of the human expert's knowledge to another is a very fragile link and often the most critical in the design and implementation process for the expert system.

The pessimistic view to the knowledge acquisition process presented by Schafer is intuitively correct. [Schafer 88:146] Each of us can fully empathize with the problems to be encountered when one human questions another about how he or she does their job. Our own experiences suggest often what we hear and what we then employ is as different as night and day as to what was meant. We often, in explaining a task or situation, make broad assumptions or generalizations fully assuming the individual with whom we are relating, either has the background necessary or can "fill in the gaps" on their own. One only has to remember the last time he or she tried to teach even a moderately simple concept to someone else to respect the task at hand in knowledge engineering. This critical breakdown in the conveyance of information can spell disaster for the expert system. Waterman further amplifies this obstacle in the following:

> ..."experts", it appears, have a tendency to state their conclusions and the reasoning behind them in general terms that are too broad for effective machine analysis. It is advantageous to have the machine work at a more basic level, dealing with clearly defined pieces of basic information that can build into more complex judgements. In contrast, the expert seldom operates at a basic level. He makes complex judgements rapidly, without laboriously reexamining and restating each step in his reasoning process. The pieces of basic knowledge are assumed and are combined so quickly that it is difficult for him to describe the process. When he examines a problem, he cannot easily articulate each step and may even be unaware of the individual steps taken to reach a solution. He may ascribe to intuition or label a hunch that which is the result of a very complex reasoning process based upon a large amount of remembered data and experience. In subsequently explaining his

> conclusion or hunch he will repeat only the major steps, often leaving out most of the smaller ones, which may have seemed obvious to him at the time. Knowing what to consider basic and relevant and not requiring further reevaluation is what makes a person an "expert". [Waterman 85:153-154]

In light of this substantial barrier to the conveyance of expertise from the human expert to be later represented in the system, a plethora of approaches have been suggested to assure validity and completeness of the gained information. Lind identified the 11 major categories of knowledge to be obtained for the construction of the expert system. They are found in Figure 2.3.

1. Relationships among various kinds of data and activities.
2. Judgements about the relative validity and importance of data and data sources.
3. Inferences and deductions from minimal, incomplete, or errorful data.
4. Bases for assumptions and educated guesses.
5. Priority judgements about the importance and order of performing various activities.
6. Recognition of promising approaches to problems.
7. Shortcuts—ways to reduce computations and steps.
8. Possible tradeoffs, and the results of tradeoffs.
9. Approximations and rules of thumb that work.
10. Unexpected or counterintuitive outcomes.
11. Ways of knowing when you are on the right track.

Figure 2.3 Major Categories of Knowledge [Lind 86:548]

The variety of the list suggests a need for a variety of investigative tools for the use of the knowledge acquisition engineer and Lind suggests four basic elicitation techniques for gaining the information for the system, [Lind 86:548]:

(1) interviews, either structured or unstructured

(2) questionnaires, including questions ranging from open-ended to completely structured

(3) problem solving sessions, where the expert is given a situation and asked to describe how he or she would reach conclusions

(4) observations of experts at their normal work

The simple application of the various forms of information collection techniques to the variety of forms of data needed to be gained, however, is not enough. As stated previously, the perilous transfer of information between two or more humans and the recording and implementation of it is much like teaching another a task, often the best intentions yield wrong results. This deviation from the intent of the information as gained by the knowledge engineer is caused by various things. I have identified some of the reasons commonly restricting the completeness of information transfer, such as lack of full recall of events or incompleteness, assumption on the part of the expert, bias, etc. Kessel has identified methods of overcoming these barriers to collection of good information for the system. She suggests remedies to attack specific verbal and judgmental restrictions to valid information gathering, as well as social interaction approaches to aid the knowledge engineer in better understanding experts. Specifically, she suggests the knowledge engineer employ methods for improved memory and enhanced recall of events. For example, do not begin with an anticipated starting value for a problem, rather, attempt to determine if different values might lead to a different method of solution. Single value approaches are often clouded by anchoring biases. Another, ask experts to imagine situations where the most probable event will not occur. [Kessel 86:544, 545] The intent of both approaches is to make the expert consciously aware of those factors that result in alternative judgements in order to make less available choices more available to consciousness. Experts may then sort out the best courses of action from this new available and previously submerged set of alternatives.

Experts should be asked the same question each time framed in a different way to insure consistency of the response. Inconsistent responses then become points of discussion between the expert and knowledge engineer to determine those factors invoked in the expert process that yielded different results to essentially the same questions or possibly the differences in framing or setup of the problem invoked by the engineer's questions.

Kessel suggests the engineer learn to guard against social interaction problems inherent in the knowledge engineering process. An example of such a problem is boredom brought on by the repetitiveness of his questions, potentially leading to omission and lack of attention to detail. [Kessel 86:543-545]

Knowledge acquisition through knowledge engineering can become tedious and drawn out but is necessary to the creation of a valid expert system. Most if not all of the difficulties of this approach are surmountable through a diligent and structured methodology by an informed professional aware of the tools as well as his own and the expert's limitations.

G. TRANSPORTABILITY

Transportability is a major concern of DOD. Little is found in the literature on this subject and for expert systems, none at all. Concerning transportability of software to different hardware systems, Warn writes:

> Software developed on one computer make or model that is not compatible with other computers hinders portability and results in forced software obsolescence. This resulting software incompatibility can occur when a manufacturer drops one model for a newer software incompatible one. It can also occur when a computer manufacturer changes its software to a new software incompatible version and then fails to support its old versions. [Warn 86:409]

Simple interpolation of the problems identified by Warn suggests the problems are further exacerbated when an application is transferred between two entirely different computer systems. To attempt to minimize this concern for DOD applications and to support system interoperability DOD Directive 5000.31 specifies unless waived, LISP, a transportable artificial intelligence language, is to be used in all AI applications developed for DOD. The intent is to certify compilers on all applicable systems to provide interoperability and transportability. [Warn 86:409] FRESH complies with this directive and was developed in LISP to a large degree, though a commercial expert system shell, Knowledge Craft (developed in another AI language, PROLOG, by Carnegie Group Inc.) is used to represent the knowledge base.

While this adequately addresses the hardware issue, nothing to date exists in the literature concerning the much more likely problem. That is, due to the nature of expert systems as defined by Hayes-Roth, this researcher feels the greatest concern to transportability is related to the issues of (1) expertise, (2) general problem solving ability in a domain, and (3) task. These three features are particularly tied to the application environment. Minor changes in the human expertise to be represented, the intended use, or the knowledge required can cause the system to be unusable in a new environment.

These concerns are potential problems for the proposed FRESH system fleet transfer. Differences between fleet procedures may be great enough to prohibit easy transport of the application to another site. For FRESH, two fleets create two different but similar problems. No academic studies have been undertaken to date to identify the range of application of an expert system to similar yet slightly different problems and the issues invoked by this transfer. Though this thesis attempts to investigate some aspects of this issue pertaining to FRESH, further research into the limits on transportability of expert systems to different decision environments must be conducted.

H. DIVERSITY OF EXPERT SYSTEMS

In order to provide the reader with a balanced view of the diversity of application of expert systems within the military alone the following overview is provided of several current systems in use within DOD:

1. Surveillance Integration Automation Project (SIAP)

SIAP detects and identifies various types of ocean vessels using digitized acoustic data from hydrophone arrays. The data takes the form of sonogram displays, which are analog histories of the spectrum of received sound energy. The system uses knowledge about the sound signature traits of different ship classes to perform the interpretation. SIAP attempts to identify the vessels and to organize them into higher-level units, such as fleets. It provides real-time analysis and situation updating for continuously arriving data. Knowledge is represented as rules within a blackboard architecture using a hierarchically organized control scheme. HASP (also known as SU/X) was an initial investigation phase and formed the basis for SIAP. The system is implemented in INTERLISP and was developed through a joint effort by Stanford University and Systems Control Technology. It reached the stage of a research prototype. [Nii 82:23-35]

2. AIRPLAN

AIRPLAN assists air operations officers with the launch and recovery of aircraft on an aircraft carrier. The system analyzes current information (e.g., the aircraft's fuel level, the weather conditions at a possible emergency divert site) and alerts the air operations officer of possible problems. AIRPLAN assesses the seriousness of a situation and manages its use of time by attending first to the most significant aspects of a problem. If time permits, it extends the analysis based on its initial conclusions. AIRPLAN is a rule-based system implemented in OPS7. It interfaces with the ship's officers through ZOG, a rapid-response, large-network, menu-selection system for human-machine communication. The system was developed at Carnegie-Mellon University and tested aboard the USS Carl Vinson, CVN-70. It reached the stage of a field prototype. [Masui 83:233-235]

3. Knowledge Based System (KNOBS)

KNOBS helps an air-controller at a tactical air command and control center perform mission planning. The system uses knowledge about targets, resources, and planned missions to check the consistency of plan components, to rank possible plans, and to help generate new plans. Knowledge in KNOBS is in the form of frames and backward chaining rules and it uses a natural language subsystem for database queries and updates. The system is implemented in FRL and ZETALISP. It was developed by the MITRE Corporation and reached the stage of a research prototype. [Engleman 79:247-249]

4. Rule-Based Retrieval of Information by Computer (RUBRIC)

RUBRIC helps a user to access unformatted textual databases. The system performs conceptual retrieval; e.g., when the user names a single topic, RUBRIC automatically retrieves all documents containing text related to that topic. In RUBRIC, the relationships between topics, subtopics, and low-level word phrases are defined in rule form. The rules also define alternative terms, phrases, and spellings for the same topic or concept. The user can formulate a query in the form of a rule that specifies retrieval criteria, e.g., a heuristic weight that specifies how strongly the rule's pattern indicates the presence of the rule's topic. During retrieval, RUBRIC presents the user with documents that lie in a cluster containing at least one document with a weight above a user-provided threshold. This prevents an arbitrary threshold from splitting closely ranked documents. RUBRIC is implemented in FRANZ LISP. It was developed at Advanced Information & Decision Systems and reached the stage of a research prototype. [McCune 83:166-172]

5. Emergency Procedures Expert System (EPES)

EPES assists F-16 pilots in handling in-flight emergency procedures, such as loss of canopy. The system uses knowledge about aircraft features (e.g., canopy, pilot) and mission goals (e.g., maintain the current state of the aircraft) to decide how to respond to emergencies. The primary goal of EPES is to maintain the aircraft at a constant airspeed, heading, and altitude. When emergencies arise, violating this goal, the system first warns the pilot and then takes corrective action, sending requests for changes to a robot-pilot. Knowledge in EPES is represented in both rule-based and semantic net form. The rules decide when to set new goals and are linked via semantic net to all parts and goals that affect their activation. EPES is implemented in ZETALISP. The system was developed at Texas Instruments and reached the stage of a demonstration prototype. [Anderson 84:496-501]

III. CINCPACFLT FRESH DEVELOPMENT

FRESH is designed as a command and communication support tool for the Navy, tracking the employment and CROVL status of almost three hundred ships distributed over more than half the surface of the world with the capability of generating optimal employment of these units in light of emergent fleet requirements. This chapter investigates the FRESH system as installed at the PFCC. As noted in Chapter One, the system was developed for CINCPACFLT by TI and BTG under contract authorization by DARPA and SPAWAR. The first prototype was developed and delivered in August of 1986 and as of this writing further prototype refinement is on going through the use of an on-site knowledge engineer and a remotely located development group in Dallas, Texas. It is at the Dallas site where the actual application coding and module development is done. The FRESH contract mandated the expert system's development under the rapid prototyping paradigm. Initially delivered in mid 1986, the system was considered operational in the late fall of 1987 (though with limited capability) after one year of refinement. It is currently in use by the CINCPAC PFCC staff while further refinement is on going.

FRESH is an extension of the CINC's existing Command and Control (C^2) fusion system, Operations Support Group Prototype (OSGP), and is intended to be one of four expert systems in a currently developing C^2 support system identified as the Fleet Command Center Battle Management Program (FCCBMP).

OSGP is a fusion of three separate databases and is grouped under a relational database management system (ORACLE). To FRESH it appears as a single database previously referred to as the IDB (see Chapter II). Using inputs from the

IDB as well as *significant amounts of instantiated knowledge in the FRESH program code itself,* the system is designed to emulate the PFCC's expert schedulers for the employment of ships in the Pacific Fleet.

This chapter will investigate FRESH within the Hayes-Roth framework of expert systems, the prototype methodology, the goals set forth for the system, as well as the knowledge, heuristics and rules of the system. It concludes with prototype enhancements which are felt desirable by the PFCC staff and last, examples of the current system's strength and shortcomings.

A. FRESH WITHIN THE EXPERT SYSTEM FRAMEWORK

FRESH, one of the largest expert systems in the world, makes significant strides in attaining the seven features of an expert system developed by Hayes-Roth and cited in Chapter Two. A discussion of how these attributes are applied within the FRESH system follows.

1. Expertise

For FRESH, expertise means the system should perform as the CINCPAC scheduling staff, replicating the staff's manual methods and heuristics of ship employment scheduling. The system's solution should compare favorably both in quality and timeliness to justify its use. The CINCPAC staff has indicated that solutions formerly requiring hours by manual methods are generated by the system in minutes with a reliability of nearly 90%. Additionally, the system provides the ability to process ad hoc queries to potential schedule adjustments that were too time intensive under former methods. The capabilities of increased speed and greater breadth of investigation have occurred without sacrifice of the level of effectiveness of the decision quality.

2. Symbol Manipulation

FRESH employs hierarchial schemas for symbol capture and presentation to the system for manipulation. The system employs both LISP and PROLOG, the two major symbol manipulation languages. LISP is used as the driver for the expert system operating on TI Symbolics™ computers and a PROLOG based shell is used for the knowledge base or schema construction. The latter, the symbolic information defined in the schemas, is largely developed through the Knowledge Craft™ shell which is user addressable. Information captured in the schemas will be discussed in more detail later in this chapter.

3. General Problem Solving Ability in a Domain

FRESH should be developed with all relevant knowledge about fleet scheduling that can be acquired. This implies the instantiation of all relevant facts about the scheduling process and the knowledge used by the staff in scheduling decisions. With this information, FRESH should have the ability to apply its knowledge to problems not specifically anticipated, and so through inference and matching of similar situations, generate a reasonable solution for anticipated, but not formally structured, decisions.

4. Complexity and Difficulty

The application for which FRESH was developed easily manifests sufficient complexity. Between the scope of the employment process, the depth of the units involved and the unknowns surrounding world events, sufficient complexity exists as well as an identifiable need to preserve the ever transitory human expertise.

5. Reformulation

This is a design feature not validated specifically by this research. It is reasonable to assume that TI and BTG have undertaken design steps to assure this feature is built into the FRESH system. The prototype methodology, which is being used for FRESH's development, is particularly good at validating this feature over the development cycle of the system.

6. Abilities Requiring Reasoning About Self

Designed into the overall FRESH development plan, this feature is a particularly essential one for this expert system. The system, being implemented in an environment where the turnover in the personnel causes a new group of computer skeptics to be system users regularly, must be able to recreate its logic for the staff user and senior officer.

7. Task

Essentially this is the essence of the thesis. FRESH should not be applied in other than those environments for which it was designed. Is the Atlantic environment and its scheduling problems of sufficient similarity to the Pacific's to allow the transportability and application of FRESH to Atlantic Fleet?

A more detailed understanding of the FRESH prototype and its development methodology is now required.

B. PROTOTYPING METHODOLOGY

Though prototyping is not the only paradigm of expert system design and development, it is the methodology contracted by DARPA for the development of FRESH. Because of this, the thesis will limit its discussion to this paradigm only.

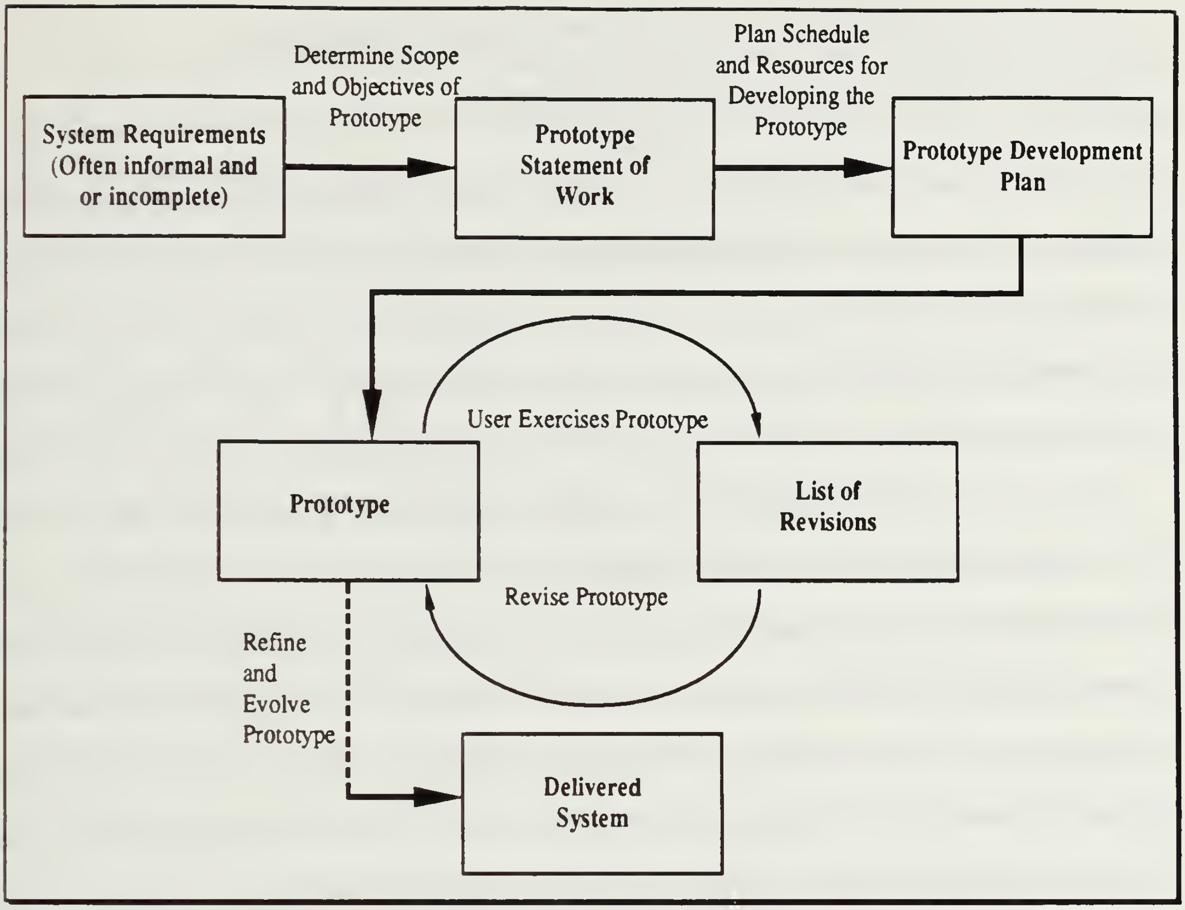

Figure 3.1 IEEE Prototyping Paradigm [IEEE 86:7]

The prototyping development approach is based on the simple proposition that people can express what they *like or do not like about an existing application system* more easily than they can express what they think they *would like in an imagined, future system* [Davis GB 85:568]. The paradigm is represented in Figure 3.1 as presented in the IEEE Tutorial of Software Paradigms [IEEE 86:7].

Distilled, the significant steps of the prototype approach are [Davis GB 85:568]:

(1) Identify the users basic information requirements

(2) Develop the initial prototype system.

(3) Use the prototype system to refine the user's requirements.

(4) Revise and enhance the prototype system.

The intent of the approach is to allow the user to see ever evolving and refining views of the system. The designer takes knowledge gained in early interviews and constructs a skeleton system of the product, typically employing screen generators and dummy programming routines to generate information and interfaces that the designer feels were indicated by the early knowledge engineering process. This "first cut" prototype system is developed quickly and presented to the user for comment and criticism typically referred to as *feedback*. This *"feedback"*, acts as new input for the designer to gain further insight into what is really expected from the system and provides more detailed information to be applied to the next generation of the prototype. For FRESH, modeling a complex environment with great diversity and significant complexity, the selection of this paradigm was intended to allow the process's multiple iterations to develop a refined system suitable to the user's needs. Its selection is felt to be entirely suitable, if not essential, to the project.

The design paradigm is not without its weaknesses. When turnover at the development site leads to inconsistency in the knowledge engineer's information, the final result is in jeopardy. Each individual may see the ultimate product differently than another. Where the initial "expert" being modeled leaves and is replaced by another, the potential exists for a new focus by the "new expert" which easily can lead the prototype in different directions. [Waterman 85:195] The prototype's development may now be in conflict with prior work. Where this occurs the final system risks being nothing more than a mix of haphazard pieces doing nothing well and typically used by no one.

Bennett cites two additional weaknesses in the paradigm. The first, when the prototype cycle is not adhered to and instead, a near final version is delivered directly from initial knowledge engineering. The second occurs when user refinements and inputs of available prototype versions does not occur, such as when the user is busy or bored with the project and fails to provide feedback. [Bennett 83:192] Either of these undermines the power of the prototype methodology—early identification of design errors which should provide easier modification of the system and reductions in costs. [Cesena 87: 7]

Waterman explains this problem as system development jumping beyond small attackable problems and basic interfaces to fully integrated modules intended to be finished versions or users losing interest in the project and no longer providing critical guidance to the developers. [Waterman 85:194] Here the prototype refinement cycle is broken and critical user input is ignored, leading to systems which do not reflect exactly the user's concept of what the finished product should look or function like.

Where the prototyping paradigm is used well, design focuses on the user's needs and goals. These are refined by user feedback to develop optimum interfaces and system responses. Iterations, therefore, leading to the most ideal system for the user. It was with this intent that DARPA contracted the development of the FRESH system by TI and BTG.

However, FRESH is being developed in a "middle-out" prototype methodology. Here the designer attempts to develop the system from close to the problems to be attacked and cycles between generalization (bottom-up) and specification (top-down) approaches throughout the problem solving process. [HURST 83: 124] An attempt to cut to the "meat" of the problems needing support and then work one's way out, first to a completed module, then a complete system. This is unlike the traditional methods of strict top-down or bottom-up approaches to system design and implementation, where the entire system would be analyzed, designed, coded, and implemented in a finished version in a single development cycle.

Stated another way, the approach attempts to divide the entire problem at hand into separate and identifiable sub-problems, therefore, sub-modules, which are developed and useful to the user as quickly as possible. This method leads to a need for continuous communication between the knowledge engineer and the user as the system expands to fruition. Supporting that need, TI has an on-site knowledge engineer for FRESH system revisions and expansions. As FRESH is still in development, knowledge engineering information is passed to the TI Dallas development site and updates to the system are forwarded back approximately every 2 to 3 months. As can be expected, the lack of on-site program development has a negative effect and an example of this problem will be discussed later in the chapter.

C. GOALS OF THE FRESH SYSTEM

From the CINCPAC perspective, the overall goals set forth for the FRESH system component of the FCCBMP are as follows [Deleot 87:CDAPPL3]:

(1) Monitor readiness and capability changes; assess significance to current/future operations.

(2) Provide alternative candidates (if needed) plus impacts associated with redirection of those units.

Figure 3.2 specifies the requirements taken from the FRESH Functional Design (FDD) document and lists the required capabilities for implementation in prototype version one. [FDD 86: para 2.1.4.2]

Detection of the significance of movements, casualties, and readiness changes based upon evaluation of user defined thresholds combined with objects (e.g., areas, battle groups, etc) for which the thresholds apply
Display of 5 year, 1 year, and Quarterly ship schedules
Identification and ranking of alternatives based upon best resolution of schedule conflicts associated with platform redirection, as well as other ranking factors
Determination of impact of schedule change
Comparison of the impact of alternative schedules
*Detection of significant changes in employment schedule
*Recommendation of rescheduling alternatives based upon user defined optimization strategies (fixed dates, fixed duration employments, etc)
*Graphic creation and manipulation of ship schedules by user
*Recommendation of schedule modifications to reduce impact
*Automatic definition of a best schedule based upon:
  -Standing commitments
  -User defined commitments
  -Fleet resource readiness and availability
*Comparison of the impact of alternative force allocation strategies

Note: items marked with a star, "*", are planned follow-on enhancements for the system in out-year development.

Figure 3.2 FRESH Functional Requirements [FDD 86: para 2.1.4.2]

The generalized algorithm used to achieve these goals is found in Figure 3.3. The figure also contains the significant sources of knowledge reasoned on to suggest schedule changes and possible substitutions. The algorithm is viewed from a high level and does not attempt to represent the exact rules provided in the system's code. The rules used to support the algorithm will be discussed next and form the basis of comparison between PACFLT and LANTFLT procedures.[2]

---

[2] The reader is invited to study Appendix A for a through understanding of the rules. They are reprinted in their entirety as published in the FDD, however, weighting factors which determine the level of impact are not represented. The reader requiring that level of knowledge is referred to the FDD Appendix E.

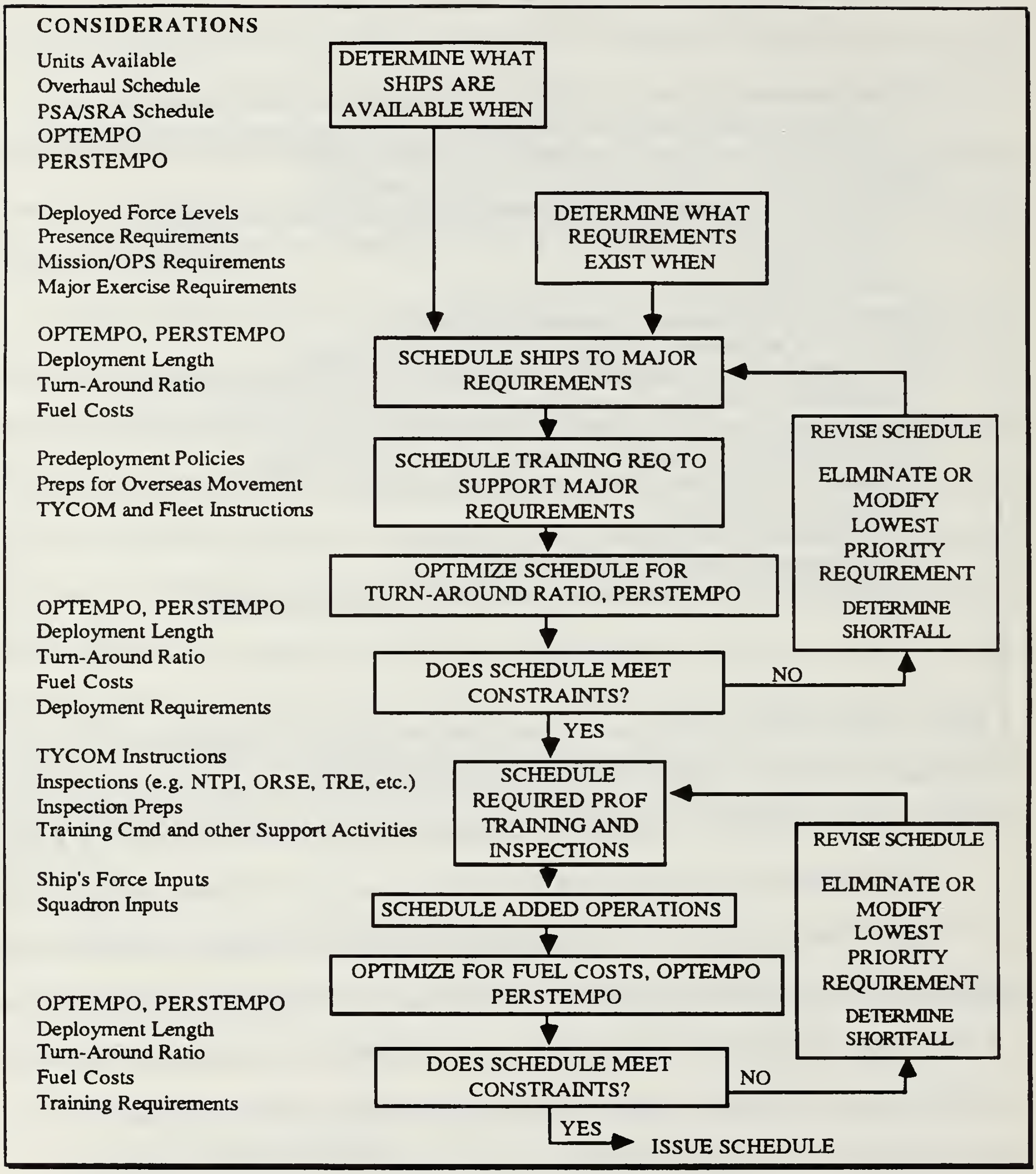

Figure 3.3 Generalized Scheduling Algorithm [Deleot 87:CDAPPL3]

D. FRESH KNOWLEDGE

In support of the algorithm, FRESH contains a significant amount of instantiated knowledge, as well as, governing rules and constraints. Appendix A contains the distillation of the significant rules and heuristics drawn from the FRESH Knowledge Base Description Document (Appendix E to the FDD). These represent a high level view of the reasoning used to support the above stated goals for the FRESH system. As an environmental indicator for this thesis, though, they are sufficiently detailed to allow comparisons of the methodologies used by the PACFLT against the desires and current practices of the LANTFLT.

1. Rules, Constraints and Heuristics

The rules and constraints used to support the FRESH system, when viewed at a high level, are used to either control the precedence of substitution for a particular unit or to assure the units do not exceed given operating limitations, such as Operating Tempo (OPTEMPO).

To clarify the latter, Navy-wide goals as of this writing are not to exceed an OPTEMPO of 50% for deployable units, or stated another way, at least one day in home port for each day at sea. As this is considered to affect personnel retention, much emphasis is placed on assuring units meet OPTEMPO limitations. And as such the FRESH system considers this in its recommendation of suitable replacements.

Additionally, significant "database like" information about the naval units being reasoned on exists in the form of hard-coded data within the FRESH application code itself. This information is coded in the program in a frame base schemata, a data hierarchy where lower elements inherit the characteristics of upper levels. Figure 3.4 is an example of this form of data representation and is taken from the Platforms Hierarchy. It illustrates the complete hierarchy for the conventional carrier's branch of the schema only [FDD 86: Fig E.1.1].

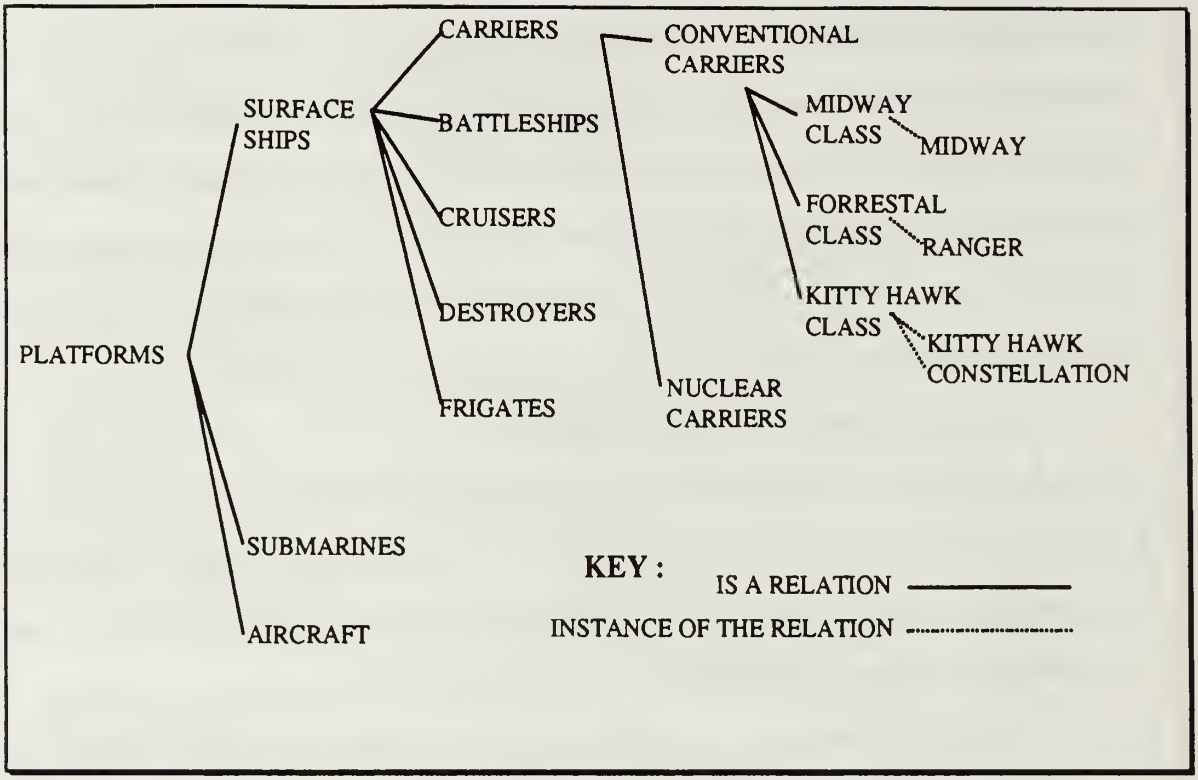

Figure 3.4 Platforms Hierarchy [FDD 86: Fig E.1.1]

Ideally, the information and characteristics captured in hierarchies should essentially be static. It is desirable that they possess a low rate of change, thus reducing system maintenance. While dynamic information, where possible, should be captured in a database system which can be addressed by the expert system.

Currently, this methodology is impossible for the FRESH system, as significant structure and normalization errors exist in its dynamic database, the IDB[3]. These data errors are corrected by hard-coded information captured within the FRESH knowledge base and updated by the on-site knowledge engineer for real world changes.

---

[3] A discussion of this problem can be found in a parallel thesis completed in March of 1988 by Lcdr Nick Sherwood, SC, USN at the United States Naval Postgraduate School.

2. Knowledge Bases

The schema hierarchies essentially contain the factual information necessary to support reasoning by the expert system. This hard-coded data input by the knowledge engineer or user often replaces information which should be available through the IDB. The eight hierarchies and examples of the information they capture are found in Figure 3.5. [FDD 86:E 1-24]

Appendix A contains the rules, constraints and heuristics and is self-explanatory. Study of these rules provides the operating and scheduling priorities used in PACFLT, essentially the scheduling environment. The broader impact of these rules lies instead in needed reasoning not reflected in the current set of rules. While the rules govern a variety of unit types and situations, it is felt they are currently too limited by the CINCPAC PFCC staff. Adjustments and additions will be necessary to fully support the required scheduling/replacement reasoning expected of FRESH.

Examples include: priority for the loss of warfare specific tailored units, such as, the loss of a TASM equipped ship should have a specific replacement hierarchy, as it is a critical battle group unit. The list can easily be extended for essentially equipped units such as, ASW helo, passive array tail ships, etc. Another is consideration of impact on other units in escort, their capabilities and needs, prior to suggesting a units cancellation from scheduled events. These suggested enhancements as well as others will be necessary in the out-year development; however, FRESH's current reasoning is sufficient for the demonstration prototype. Significant thought must be given to further reasoning refinements to assure battle group degrading recommendations do not occur.

Platforms: Holds all information concerning the PACFLT platform resources.	Examples are: - Ship-name - Fuel-consumption-coefficients - Prior Crovl rating/new Crovl rating - Reason for Crovl change - Warfare ratings by mission area - Primary-mission - Equipment-on-board
Employment: NWP-7 information about employment schedule, EMPSKD.	Examples are: - Unitrep-Activity-Code - Category-Number (NWP-7) - Individual-unit-exercise and Importance
Equipment: Structures all equipment and weapons systems on each unit. NOTE: Is not fully developed in prototype one version.	Examples are: - Platform-names - Equipment-name - Military-designation (i.e. SPS 48) - Equipment-description
Geographic Location: Contains dynamic information about types of geographic locations, such as ports, air stations, bodies of water. Contains conditions and restrictions for same as well as operational threats to units in these areas.	Examples are: - Central-coordinate - Air-threat - Surface-threat - Sub-surface-threat - Unitrep/Casrep (thresholds for entry) - Country
OPCON: Stores how each unit relates to a given task group. This may be a Battle Group or a Surface Action Group or any other organizational element.	Examples are: - Unit-reports-to - Opcon-members - Current-mission - Battle-group - Principal-ship
Battle Groups: Contains information about battle group configurations in terms of unit and equipment configurations.	Examples are: - Required-platforms - Required-equipment - English-names-of-equipment
Activities: Contains the requirements for the different fleet defined activities that may be defined for FRESH. Objectives for the given activity form the hierarchy tree for that activity.	Examples are: - Activity has-objectives - Priority - Required-platforms - Required-equipment - Start/End-date
Readiness Evaluation and Casualty Thresholds: User input levels of minimum acceptable readiness thresholds for various ships, areas missions, etc. This provides a direct mapping to the Platforms and Activities schemas.	Examples are: - Crovl-threshold - Warfare-ratings-thresholds - Ship-name - Mission - Ship-class

Figure 3.5 FRESH Schema Hierarchies [FDD 86:E 1-24]

E. THE OPERATIONAL FRESH PROTOTYPE

In its second year of an on going prototype development plan, Version One of FRESH has already proven to be a benefit to the PFCC, though its use is still limited. Its growth to an early operational version has not been without problems. The following investigates its current operational applications and the development successes and problems surrounding the system.

1. FRESH Knowledge Engineering and Validation

The knowledge covered in section D is not meant to be all inclusive, rather an overview of the breadth and depth of the information hard-coded into the system presently, as well as, its future needs. A great percentage of this knowledge should be obtainable directly by FRESH from the IDB when the IDB databases are normalized. When the IDB is corrected, much of the manual intensiveness currently required will be relieved. While this hard-coding, for the most part, adequately handles needed support for the FRESH system, limitations have occurred due to the system's inability to recognize the wide extent of differences that exist between even two units of the same class. FRESH often still accepts incorrect input from the IDB and has no recourse but to display it as accurate data. In other words, the system cannot and reasonably should not, be expected to catch all database inadequacies. A major effort to provide correct input data is required if the system is to be of any value.

An example of the system's inability to screen invalid data occurs for the nuclear carrier NIMITZ (CVN-68), which is listed by the FRESH system as being configured with *two mutually exclusive* Surface to Air Missile (SAM) systems, Nato Sea Sparrow Missile System (NSSMS) and Basic Point Defense Missile System (BPDMS). Not necessarily a critical item for the current usage of FRESH, but as the system in the future is used to replace units based on equipment loss and its effect on the units ability to continue mission and defend itself, the error could have dire effects. If the NIMITZ were to send a Casualty Report Message (CASREP) that her configured SAM system, NSSMS, was inoperable, the FRESH system currently would reason that the ship had BPDMS and therefore, still possessed an Anti-Air-Warfare (AAW) capability. This error could allow NIMITZ to continue in harm's way without any AAW capability. Obviously, incorrect answers to operational queries that are posed to the system are possible in its use by individuals who are not aware of the validity of data presented to them. Its use now must be carefully monitored should the wrong answer be provided to the right question.

Verification and Validation (V&V) is a significant problem for any system this size, particularly for FRESH as human life often hangs in the balance. Though safeguards exist in all programming to cover critical contingencies, no program of the magnitude of FRESH can be fully verified and validated.

Pressman provides an example of the complexity of V&V. Here, the thorough testing of a small 100 line Pascal program consisting of 5 logical paths, to be looped through no more than 20 times, is to be verified with a yet to be built super computer. This mythical computer can develop a test case and execute it on one of the 100 trillion logical paths defined by the above program every single millisecond. Even with this "magic power", it would still take 3170 years to test all possible paths. [Pressman 87:471] Realizing the magnitude of effort for such a small system, one rapidly begins to understand the problems of V&V for a system the size of FRESH. The developer and the Navy must take great pains to assure both quality and the correct data relations exist in the very earliest stages of the system. It is well established that postponing maintenance and modification of the system now will cause greater maintenance impact and cost later. If properly employed, the prototyping methodology minimizes this effect; however, significant efforts will be required by TI/BTG and the Navy to assure the strengths of this paradigm are reaped on the FRESH project.

2. Current Uses and Limitations of FRESH

As previously noted, FRESH is currently extremely limited in its use at CINCPACFLT. Its essential task is the monitoring of unit level mission capability degradations and alerting the PFCC staff of the drop in the units abilities to continue the mission. Stated another way, when a naval unit suffers a mission degrading personnel or equipment casualty, it reports that to higher authority through a required reporting message called a CASREP. This report is provided to allow the PFCC staff an evaluation of the possible impact on the fleet's overall mission goals, and to propose a suitable replacement in the event the unit is judged to be an ineffective platform for the assigned task. This evaluation takes place using several factors governing the units capability to continue, as well as, a proposed units ability to replace it. The major governing factors are the mission requirements and capabilities needed to complete the assigned mission, operational tempos of the units, cost in fuel to make the change, and the threat environment. This scheduling is essential and automation is highly desirable The FRESH interface configuration is extremely lacking as it pertains to its basic support of these functions. As currently designed, the system alerts the PFCC when a units rating has fallen below a preset threshold defined in the system. This reaction to a degradation in a ship's capability to perform assigned scheduling is correct and the event knowledge engineered to be acted upon. However, the essential information needed by the staff regarding the alert is not presented to the user, yet is available to the system. This being: reason for the alert, expected date of correction of the condition, and ability to continue the assigned mission.

Other uses for the FRESH system include, ad hoc queries and fuel consumption computations. The latter is a recent refinement and was unusable in the initially delivered version.

Ad hoc queries is an area in which FRESH excels. The system allows for the construction of different scheduling scenarios which can be tested for future fleet-wide impact. If this impact is unsuitable, the proposed schedule modification can be quickly changed and retested. While the system cannot currently generate a complete schedule by any means, its ability to adjust and test alternatives to the current schedule well exceeds that of the human expert in speed and depth. The variety of questions that can be posed to the system are almost limitless in this mode, though they are restricted to a single unit at a time.

Fuel use computations are a recent benefit gained by the system with the correction of original knowledge engineering information. The system originally used maximum economic speed for all transit fuel computations. While this might seem reasonable, this is far from the real world being modeled and a failing of the knowledge engineering done up front. [McNeil 88:15] This, subsequently, has been reengineered and correct fuel use tables installed. Now the user can suggest a units transfer to another geographic position and the fuel impact and cost are easily obtainable. In light of current fiscal restrictions, this is an extremely useful capability for the command center.

Though limited, the system is, nevertheless, extremely useful. It is felt by CINCPAC's staff that time and personnel savings, up to one-twelfth, are provided over previous manual methods. While this is significant, the system did not gain full benefits from the strengths of the prototype development process. The weaknesses previously noted bring to light what this researcher considers to be the largest shortcoming of the FRESH project to date—failure to closely follow the prototype development paradigm and, therefore, failure to reap the benefits to be gained by its use.

As presented by the CINCPAC FRESH project manager, the initial prototype, delivered in less than 18 months, attempted to demonstrate the full range of decision making capabilities. This disregards the essential benefits of prototyping to build a little, deliver a little and test, while gathering feedback to apply to the next generation of the prototype. [McNeil 88:3,21-24] Additional hindrance to FRESH's development occurs with the geographic separation of the development staff from the user. While this is cost beneficial to TI, it is not in the best interests of the Navy. Studies of the co-location of the development staff with the user have proven significantly superior systems are produced. Co-location fosters cohesion and a feeling of partnership among developers, user-representatives, and end users. Additionally, time between prototype versions is reduced and quality of feedback increased. These benefits yield significant cost reductions and higher quality systems. [Cesena 87:6-8]

F. SUMMARY

In this chapter, the planned development process for the FRESH system, the knowledge held by the system, and current problems have been explored. Additionally, enhancements to the system requested by the PACFLT personnel have been noted. Reasonably, these latter can be expected as a requirement for the CINCLANT version of the system.

While failings of the prototype approach have had an effect on the development of FRESH, whether by inherent problems with the paradigm or the application of it by the developer, the system remains a success. Yet, these failings are a reminder that great care must be taken where neither the expert understands expert systems nor the developer understands the application environment. This author believes careful application and strict adherence to the prototype paradigm could have greatly reduced problems encountered to date.

IV. CINCLANTFLT ANALYSIS

As stated in Chapter Three, an expert system, such as FRESH, should not be applied in other than those environments for which it was designed. Is the Atlantic environment and its scheduling problems of sufficient similarity to the Pacific's to allow the transfer and application of FRESH directly to the Atlantic Fleet? To answer this question, an understanding of the CINCLANTFLT current scheduling methods and environmental differences between the Atlantic and the Pacific must be understood. With these characterized, it is a relatively straightforward task to compare them to FRESH's current reasoning and make intelligent conclusions regarding the extent of modification required to allow its application to the Atlantic Fleet Command Center (AFCC).

In the following sections of this chapter, the Atlantic perspective will be developed—first, the methodology and scheduling procedures in use and second, an overview of emphasis areas developed by interviewing ("knowledge engineering") the current scheduling experts at the AFCC. From those knowledge engineering notes (extracted in Appendix B), the differences noted between the two fleets, in emphasis and procedure, can be deduced. Several of these identified differences then will be contrasted against current FRESH reasoning, as well as, an exploration of the required modifications to the knowledge bases. Last, those areas the AFCC and CINCLANT staff want supported in an Atlantic version of FRESH are identified and a proposal is made regarding transfer methodologies for the system.

Initial suggestions of political resistance to this research did not occur, rather, formal initiatives now exist to transfer FRESH technology to the Atlantic Fleet.

AFCC personnel who have viewed the system are impressed. When compared to their totally manual methods, FRESH, even in its limited form, provides significant options and support currently lacking at the AFCC. With this positive attitude, little doubt exists that FRESH or a system like it will eventually be developed for the Atlantic Fleet. As of this writing, a transfer initiative still formally exists, however, fiscal restrictions may postpone it beyond the planned 1989 transfer date. Realizing the application of expert system technology will most probably occur and that using an existing system is significantly cheaper than building a new one, it remains relevant to consider differences between the two Fleets.

A. THE ATLANTIC FLEET SCHEDULING PROCESS

While the Pacific Fleet already enjoys, to a limited extent, the benefits of automated support in the scheduling problem, the Atlantic Fleet, for all practical purposes, has none! Instead, a highly manual method is employed using several full time scheduling officers, as well as, additional support personnel. In this manual scheduling process, computer support is limited at best to simple database queries.

1. The Manual Scheduling Process

Requests for naval unit participation in events (exercises, training operations, support and public relation visits, etc.) originate from various sources ranging from the Secretary of Defense to the individual unit level commander (Commanding Officer of a single ship or squadron). These requests are forwarded to the CINCLANT Quarterly Scheduling Conference, where rough Type Commander level(Commander Naval Surface Forces Atlantic, Commander Naval Air Forces Atlantic, and Commander Naval Submarine Forces Atlantic) schedules are compiled with CINC level commitments and direction to develop a finished schedule for the fleet.

These conferences and the preceding staff work are entirely manual, with scheduling experts piecing the puzzle together in a time consuming and iterative way. The scheduling experts rely on their experience gained on the job, guidance from the resident CINC as to employment priority, and their own operational background in the fleet to complete this formidable task. In the overall process, computers are only used to store and retrieve schedule data; they are not used to assist decision making [Goodman 85:9].

For the Atlantic fleet, there exists no method to quickly "juggle" the schedule and assess impact as FRESH with its ad hoc query capability allows for PACFLT. Instead, experienced schedulers rely on intuitive feelings, best guesses and rules of thumb, both to construct, as well as evaluate, the effectiveness of the schedule. The construction of the Atlantic schedule is similar to the Pacific's in that it is completed in a bottom up as well as top down methodology. Requirements from above the CINC level are pushed down the chain of command to the CINC and requests for events are passed up the chain from the unit level through the Type Commander. As with most organizations, the AFCC staff is faced with more requests for scarce unit scheduling than they have available to fill the scheduling. Therefore, a juggling act occurs matching events to requirements where sufficient assets do not exist to meet requirements. Here, as it has been in the Pacific fleet, the implementation of FRESH can be of great benefit with its excellent ability to provide answers to "what if" sort of queries. [Goodman 85:13]

2. CINCLANT Organizational Control

While the AFCC will certainly benefit from FRESH technology, differences from the Pacific Fleet do exist in organizational structure and control. As has been alluded to, we in the United States Navy, in fact have two Navys. In exploring the differences for this thesis one significant organizational difference appears to exist between the two—scheduling and operational control of Naval units appears to be more decentralized in the Atlantic than in the Pacific. Whether this is due to the presence of FRESH causing centralization in the Pacific fleet or whether the two numbered fleets present in the Atlantic have more widely varied missions (2nd Fleet, Atlantic and North Atlantic theater and 6th Fleet, Mediterranean theater) acting as a decentralizing factor is unknown. This difference in and of itself will require further study beyond the scope of this paper to determine the effect of centralization of scheduling and its compatibility with over all operational requirements in the Atlantic.

One can develop a feasible solution to this decentralized scheduling, such as, the distribution of FRESH technology to the Numbered Fleet level by providing them their own stand alone systems or more reasonably, access to the CINCLANT system. While this decentralized organization might be seen as a significant impediment to the implementation of FRESH at CINCLANT, it is not. As in any organization, authority is delegated, while responsibility is withheld so final approval of all scheduling rests with the AFCC staff, and automated support is needed here. FRESH has been viewed at that level as a significant tool to support scheduling, ad hoc query, and real-time response option generation.

While subordinate units are possibly given greater control over their units' scheduling in the Atlantic, they are expected to comply with CINCLANT scheduling methodologies. This standardization in methodology allows the discounting of organizational control differences for this paper. With this difference dispensed with, the stated desire for automated support by the AFCC staff, as well as CINC level intentions for FRESH's transfer, validate the need for a thorough study of the Atlantic manual procedural differences and the knowledge held in support of these procedures.

In the following section several significant examples of the Atlantic procedures are provided to contrast the extent of the environmental differences between the fleets. For a more in-depth understanding of the differences at the rule by rule level view, the reader may contrast Appendices A and B.

B. IDENTIFIED DIFFERENCES IN THE ATLANTIC ENVIRONMENT

Based on Hayes-Roth's views of task significance of the expert system, it is sufficient to develop two or three basic differences in underlying methodology to infer a need for some other transfer method than direct implementation of the existing system. [Hayes-Roth 83:49] The result of knowledge engineering of the AFCC procedures provided several examples of difference in underlying methodology. It must be remembered, however, that the CINC retains at all times the ability to modify existing procedures. However, except where dictated by extreme circumstance (situations where the AFCC would deviate from normal procedure), the following differences in basic methodology exist between the two fleets (extracted from Appendix B):

(1) Complete use by a unit of its allotted fuel quota is considered much more important than OPTEMPO in most scheduling decisions.

(2) Disruption of training or other precedence event scheduling is apparently much more acceptable to support high priority schedule changes.

(3) Units with less than Crovl Rating of C-2 are to be used in scheduling replacement only rarely, and units of less than C-3, never. (CINC may authorize C-4 on a case by case basis.)

While these three exhibit differences in decision rules employed at the AFCC, changes in the manner the system accesses information for its knowledge base are required to support these rules—most notably a requirement for an accurate, near real-time dynamic database, to provide input.to the system vice stagnate, in-code knowledge representation. To support an Atlantic expert system, whose scheduling constraints would be based greatly on fuel use and allocation vice OPTEMPO (a major driving factor in the current FRESH reasoning), the system would need real-time data providing current fuel states and planned fuel usage for potentially assigned tasks. Though this can currently be computed to some degree of accuracy in FRESH, further refinement appears needed for this element to become the primary influence in scheduling. This need for an improved dynamic database is not unique to the Atlantic, as Chapter Three noted the need for similar capabilities for the Pacific system. For the Atlantic though, its implementation is critical to even the most basic use of FRESH, because an extremely dynamic and unpredictable variable is the major governing facet.

1. Basic Differences

Explored more closely, the difference cited in (1) above is stated as follows by the AFCC staff: OPTEMPO is based on fuel availability—you must burn the fuel or lose it in the following year's allocations. This factor, complete use of fuel allotted, dictates steaming days to a greater extent than OPTEMPO requirements and is to be weighed more heavily than exceeding the 50% at home to at sea ratio. This approach contrasts directly with the major governing limit in the FRESH system, where the 50% OPTEMPO limit is used to reject a unit from consideration as a replacement candidate. For the Atlantic, this basic rule does not strongly apply and, as a significant limiter in the current FRESH system, would need to be modified to support transferability.

Another set of limiting conditions used in the current system which appear to require modification in an Atlantic version are framed above in (2). Where the Pacific would typically restrict or prohibit the use of units in the control of type commanders or other restricted events as replacement units, the Atlantic would actively consider use of these units. Atlantic procedures appear to routinely consider both type commander controlled units, as well as, units in Preparation for Overseas Movement (POM) or Restricted Availability (RAV). While the use of these units may require CINC approval ultimately, at the generation of alternatives level, these units should be considered as replacements and not removed from consideration, as is currently done by FRESH.

The last significant difference cited above, (3), involves what is perceived as a more restricted level of readiness required in the Atlantic Fleet. Where FRESH might eventually generate a do-nothing option and allow a C-4 unit to complete an assigned task in lieu of no other acceptable replacement unit, the Atlantic, typically, would rather cancel scheduling for any unit less than C-3 and easily could require CINC approval for the use of a unit with less than a C-2 rating. While changes to this particular limiting rule in FRESH may not be extensive or difficult, this latter, paired with comments about (2), suggests a general willingness by the Atlantic to consider a wider range of options than might be considered acceptable to the Pacific Fleet. The Atlantic system would apparently need to provide a wider range of choices to be selected from by the user based on his more detailed understanding of the CINC's current intentions.

These three points highlight significant differences in basic methodology between the two Fleets, though they are not the only ones identifiable by knowledge engineering at this upper level view of of the FRESH system. Other differences which are found in Appendix B include: differences in both unit and order of unit substitution (for Carriers, DD's, FFG's, and FF's) and Personnel Tempo (PERSTEMPO) restrictions are more relaxed.

2. The Impact of North Atlantic Treaty Organization

Developed in discussions with the AFCC staff are reasoning requirements to deal with what may potentially be the most significant reasoning modifications required for FRESH—the ability to support CINCLANT's second sphere of responsibility, that of Supreme Allied Commander Atlantic, SACLANT, under the North Atlantic Treaty Organization (NATO).

In peace time CINCLANT's naval forces are essentially limited to approximately 250 naval units of United States flag only. (Actually, approximately 10-12 units of foreign flag are assigned to SACLANT as Standing Forces Atlantic, a token NATO fleet.) These US naval forces are essentially the same composition and strength as FRESH deals with in the Pacific. Therefore, relatively simple modification to the existing knowledge base is required to apply it to individual units in the Atlantic and should present no significant problem to FRESH's transfer.

For CINCLANT under his SACLANT operational hat, this US force does not represent the only force scheduling responsibility the AFCC would have to deal with. Consider the following: as international tensions begin to increase, so does the need to redirect units' scheduling in response to these tensions. With this redirection, were it to be available, a commensurately greater reliance on expert system support would occur. With this increase in tension, a phenomena occurs in the Atlantic unparalleled in the Pacific—NATO countries begin to incrementally transfer (chop) command and control of their units to SACLANT. This NATO force multiplier effect quickly swells the CINCLANT, now SACLANT, force to well over double its peace time size. This also commensurately increases the perceived threat Order of Battle, as Warsaw Pact nations chop to Soviet Fleet Commanders. These factors, both the increased friendly forces, as well as, increased "Orange Force" threat factors, must be handled by an Atlantic version of FRESH.

The above is not an insurmountable task for an expert system, but *it is a hard requirement for an Atlantic version* as viewed by the AFCC staff, which will require development by the contractor. It is easy to understand why the requirement for this absorption capability is needed. If the system were to be delivered at its current level of functionality, restricted to US Naval Forces only, the users would begin to depend on its ability to provide expert scheduling and response handling in routine use. From FRESH's daily use the human expertise will wane as users become unaccustomed to manual scheduling. Yet, exactly when the expertise or expert support is needed most, the user, facing increasing world tensions and force multiplication, would also be faced with manual scheduling and generation of operational response. Predictably, the result would be the inability to handle the required scheduling when it is most needed.

3. Additional Reasoning and Knowledge Desired

While the current system would support the CINCLANT environment following suggested modifications outlined above, several additional features have been identified by CINCLANT staff personnel which would be desired if the system were transferred.

Probably the most significant is the ability to reason on specific unit equipment vice a generic class equipment load-out for replacement considerations. Often the most valuable asset a naval unit provides to a battle group or exercise is not its presence as a particular class of ship, but rather, the specific load-out of equipment she carries. Here, the system should replace based on the needed support equipment vice a particular unit.

For example, not all cruisers are Tomahawk launch capable platforms and the loss of the Tomahawk capability is more significant than the loss of the unit. Therefore, replacement scheduling of any other Tomahawk unit is more significant than a standard cruiser type replacement process, which FRESH might currently employ.

Other capabilities desired for Atlantic support include the ability to handle Naval Air squadron scheduling, as well as, submarine scheduling. These same requirements have been cited as follow on needs for the Pacific fleet and should be applicable to the Atlantic also. For these two force categories, however, the same SACLANT factors apply as additional units' from these categories are added in NATO alert situations.

Regardless of the modification and enhancement of the Atlantic version of FRESH, added reasoning and supporting knowledge bases are required. While currently coded modules may provide a code library of sorts and in many cases be directly modified to support new functions and reasoning bases, the extent of the change is significant. The writer envisions the need for added knowledge bases to track NATO units with boolean operators to indicate the current availability or nonavailability of the unit for SACLANT's control. When indicated as chopped to SACLANT, the units current characteristics and capabilities data (which until that point would be restricted from FRESH reasoning) would now have to be automatically loaded to the knowledge bases. Similar knowledge bases to support NATO air and subsurface units, which would be chopped are required as well. These same knowledge base refinements for threat related information in the schemas are required as they pertain to "Orange Force" related factors, which also change in a globally escalating threat environment.

While this addresses at a basic level enhancements to FRESH for the additional knowledge bases, the current knowledge base obviously will require a complete reload to support transfer. Individually, the eight hierarchies in Figure 3.5 are almost completely reinitialized, though this requirement is rather obvious. Rules will require modification at much lower levels than the view presented in this thesis and their interaction with NATO force multiplication explored further. The possible new choices presented by that force multiplication will probably dictate entirely new replacement hierarchies, often requiring trade-off analysis of equipment suites and equipment capability unit by unit.

The need for modification of the current FRESH system should not lead the reader to think the task is insurmountable and the transfer of FRESH is inappropriate. Rather, it is felt the technology available through the FRESH system can be of great benefit to the AFCC staff. With no capability to generate timely, therefore useful, ad hoc "what if?" queries to scheduling possibilities, significant efficiency and effectiveness is lost in Atlantic procedures. In discussions with AFCC staff, the amount of time required to do any manner of sensitivity analysis to a generated schedule change is so time consuming that it rarely, if ever, is conducted. This leads the scheduler to the never ending circle of putting out the next brush fire, often times one he himself often set by earlier scheduling selections. This cycle of being locked into the reaction mode more than the planning mode is detrimental and one for which FRESH provides significant relief and alone may justify its transfer.

The issue becomes not whether to transfer the technology, but rather, how? As always, experience should be our guide. The history of systems development has shown, where the user is unfamiliar with exact requirements and needs and the developer is not an expert in the area to be modeled, prototyping is the most suitable development methodology. [Davis GB 85:567,568]

C. SUMMARY

This chapter has addressed the methodology currently used by AFCC staff personnel to manually schedule their units. Discussion has focused on their standard operational differences in unit handling, as well as, issues requiring modification to allow the system to be used in light of the SACLANT role performed by CINCLANT. While changes in the specific structuring of the FRESH rules will be left to the system developers, the identification of significant environmental differences suggest some transfer method, other than a direct transfer of FRESH, might be preferable.

Regardless of these changes, the system is needed by the Atlantic Fleet. Atlantic schedulers and current Pacific users alike agree—any system that supports their need to plan instead of react is a benefit. Specifically, AFCC personnel indicate they would be overjoyed to have a system that alone would provide the ability to explore ad hoc issues and its installation would gain immediate acceptance. While the issues stated previously in this chapter are many, they are not insurmountable by careful system design and implementation.

V. CONCLUSIONS

This chapter concludes the thesis and to a large extent restates and summarizes information found in Chapter Four. While issues were raised in Chapter Four beyond the original scope of the thesis and remain unanswered, their solutions will be left to the developer and will require in-depth knowledge engineering. These questions, involving the issues of SACLANT and organizational control in the Atlantic, are solvable and should be functional requirements in the CINCLANT expert system. While these are issues yet to be dealt with, their presence provides the solution to the basic question to be answered by this thesis. Do significant environmental differences exist to make the application of FRESH directly questionable? The answer is yes. The NATO issue alone, an issue not envisioned at the outset of the study to be of near the magnitude it presents itself to be, places the existing system's transfer in question without the basic scheduling control and organizational differences also cited.

The question as to whether the transfer of FRESH technology is worth the effort is a hard one for which to develop an objective answer. Certain intangibles exist which are extremely hard to quantify—a human life, for example. This author developed previously the potential costs involved in the use or misuse of FRESH in the military environment in which it is employed. In the game of C^3, given a FRESH system which is optimized to support the specific CINC's environment, the potential for success in the preservation of life and material is high. The CINCLANT staff's preference is to support the acquisition of some sort of automated support tool. FRESH or some derivative is currently preferred—desperately needed is a more meaningful statement. When being interviewed on

SACLANT effects and presented the potential difficulty to the system that the NATO issues raise, an AFCC staffer stated in near desperation, "It would seem there must be a way to make it work in the NATO environment!?" The automated support of scheduling and the benefits of ad hoc query is something the CINCLANT staff desperately wants. It is, therefore, reasonable to expect the FRESH system or a derivative of it to eventually be installed at the AFCC for their use.

The CINCLANT environment provides even greater diversity for the developer to deal with than is present in the Pacific fleet. The issue becomes how to provide the Atlantic fleet with a system refined and properly focused on their needs, while at the same time guiding their understanding of the automated support capabilities of an expert system. This development methodology should be guided by what we have learned so far so as not to repeat errors which existed in the Pacific.

On-site prototyping remains the best methodology available to bring the new vistas of automated support to the Atlantic fleet. To the Atlantic Fleet schedulers, who are totally unaccustomed to automated support tools of any kind, this methodology is even more important. With prototyping, their expertise in this new automated environment and the system's capability to support them can grow hand in hand throughout the prototyping cycle. We must learn from the mistakes in the development cycle perceived by the Pacific Fleet staff and assure that the strength of the prototyping paradigm is used. [McNeil 88:21-25] The development must be based on speed of iteration of the prototype cycle to gain critical feedback to influence further refinement vice initial functionality, as was done with the current version of FRESH.

One last recommendation for the Atlantic development methodology pertains to off-site development. As stated by Cesena [Cesena 87:7]:

> Co-location fosters cohesion and a feeling of partnership among developers, user representatives, and end users....End-user demonstration and reviews identify design errors early in the development cycle and 4GL (Fourth Generation Language) tools permit easier modification of the software.

The PFCC staff feels FRESH has suffered from the remoteness of the off-site development teams. They now state, they would have preferred a faster more modular level approach to the prototyping cycle, supported by on-site development. Small, user-digestible cycles allow sufficient feedback to create a system that supports the users' needs.

REFERENCES

[Anderson 84] Anderson, B.M., and others, "Intelligent automation of emergency procedures in advanced fighter aircraft," *Proceedings of the First Conference on Artificial Intelligence Applications, IEEE Computer Society*, December 1984.

[Bennett 83] Bennett, John L., *Building Decision Support Systems*, Addison-Wesley Publishing Co., 1983, pp 277.

[Bui 87] Bui, X. Tung, Lecture from forthcoming work presented at the Naval Postgraduate School, Monterey, CA, Sybex Publishing Co., 1988, in press.

[Cesena 87] Cesena, Mary L., "Accelerated Information Systems Development in the Army," *SPECTRUM*, vol. 4, number 3, September. 1987, pp 1-8.

[Chorafas 87] Chorafas, Dimitris N., *Applying Expert Systems in Business*, McGraw-Hill Book Company, New York, 1987, pp 232.

[Davis GB 85] Davis, Gordon B. and Olson, Margrethe H., *Management Information Systems, Conceptual Foundations, Structure, and Development*, McGraw-Hill Book Company, New York, 1985, pp 693.

[Davis MW 85] Davis, Michael W., "Anatomy of Decision Support," *Datamation*, 1985, June 15, pp 201ff.

[Deleot 87] Deleot, C., "FCC FCCBMP briefings slides," August 1987.

[Engleman 79] Engleman, C., and others, "KNOBS: an experimental knowledge based tactical air mission planning system and a rule based aircraft identification simulation facility," *Proceedings IJCAI-79*, 1979, pp 247-249.

[FDD 86] Texas Instruments Inc. and btg, Inc, *FRESH as-built Functional Design, FRESH I, Software Release 2.0, Functional Design Document,* TI INC, Dallas, TX, 5 December 1986.

[Forsyth 84] Forsyth, Richard, *Expert Systems, Principals and case studies*, Chapman and Hall Ltd., London, 1984, pp 231.

[Goodman 85] Goodman, Clarke E., *Annual Scheduling of Atlantic Fleet Naval Combatants*, Naval Postgraduate School, Monterey, CA., pp 103.

[Harmon 85] Harmon, Paul and King, David, *Expert Systems*, John Wiley & Sons, New York, 1985, pp 283.

[Hayes 85] Hayes, John R., Segal, J., editor, "Three Problems in Teaching General Skills," *Thinking and Learning*, L. Erlbaum, vol. 2,1985, pp 639.

[Hayes-Roth 83] Hayes-Roth, Frederick, *Building Expert Systems*, Addison Wesley Publishing Company, 1983, pp 444.

[Hurst 83] Hurst, E. Gerald and others, Growing DSS: "A Flexible, Evolutionary Approach," *Building Decision Support Systems*, Addison-Wesley Publishing Co., 1983, pp 277.

[IEEE 86] "What are the New Paradigms", *IEEE Tutorial: New Paradigms for Software Development*, Agresti, William W., editor, Computer Society Press, 1986, pp 6-10.

[Keen 76] Keen, Peter G.W., "Computer Systems for Top Managers: A Modest Proposal," *Sloan Management Review*, vol. 18(1), pp 1-17.

[Keller 87] Keller, Robert, *Expert System Technology*, Yourdon Press, 1987, pp 246.

[Kessel 86] Kessel, Karen Lee, "Methodological Tools for Knowledge Acquisition," *IEEE Transactions, Systems, Man, Cybernetics*, 1986, vol.1, pp 541-546.

[Lane 86] Lane, Norman E., "Global Issues in Evaluation of Expert Systems," *IEEE Transactions, Systems, Man, Cybernetics*, 1986, vol. 1, pp 121-125.

[Lind 86] Lind, Judith H., "Downloading the Expert: Efficient Knowledge Acquisition For Expert Systems," IEEE Transactions, Systems, Man, Cybernetics, 1986, vol. 1, pp 547-551.

[Masui 83] Masui, S., and others, "Decision-making in time-critical situations," *Proceedings IJCAI-83*, 1983, pp 233-235.

[McCune 83] McCune, B. P., and others, "RUBRIC: a system for rule-based information retrieval, "*Proceedings of the Seventh International Computer Software and Applications Conference, IEEE Computer Society*, November 83, pp 166-172.

[McNeil 88] McNeil, Michael, *Building a Knowledge Based System in the Operational World,* Proceedings of the Knowledge Based Information Systems Symposium, 1987, SHAPE Technical Center, The Hague, Netherlands, in press.

[Nii 82] Nii, H. P., Feigenbaum,E.A., and others, "Signal-to-symbol transformation: HASP/SIAP case study," *The AI Magazine*, Spring 1982, pp 23-35.

[Pressman 87] Pressman, Roger S., *Software Engineering A Practitioners Approach*, McGraw-Hill Book Company, New York, 1987, pp 567.

[Schafer 88] Rasmus, Daniel W., "Expert Input," *MacUser*, 1988, vol. 4, number 1, pp 136-150.

[Warn 86] Warn, Keith., "LISP vs. Ada: Implications in diagnostics -oriented expert systems," *IEEE International Automatic Testing Conference,* 1986, pp 409-415.

[Wang 87] Wang, Eugene, "Artificial Intelligence for Competitive Advantages," *AI Expert*, vol. 2, number 1, January 1987, pp.5-6.

[Waterman 85] Waterman, Donald A., *A Guide to Expert Systems*, Addison-Wesley Publishing Company, 1985, pp 419.

APPENDIX A

This appendix presents a tabular view of the rules, constraints and heuristics of the FRESH system as presented in Appendix E of the FDD. A study of this table will provide the reader with an upper level view of the Pacific Fleet schedulers' normal solutions to a scheduling problem. These rules, constraints and heuristics form the system instantiated environment for FRESH and are compared to the Atlantic Fleet's manual methods and scheduling priorities to identify the extent of difference between the two fleets.

For CONSTRAINTS, the TI/BTG identification number is presented first followed by the name of the constraint. This is then followed by the desired function, limit or choice to the given situation. Any relaxations or limitations indicated in the FDD are presented following the specific constraint.

For RULES, the name and type of rule is followed by classic if-then presentation: where given a situation "IF" occurs—"THEN" take the following action.

For HEURISTICS, the reader will find the statements themselves stand alone and represent a strict action to take in the event of a given situation.

CONSTRAINTS

Constraint Number:	ORG-035-0
Constraint Name:	OPTEMPO-NEGATIVE-CONSTRAINT
Constraint:	A SHIP'S OPTEMPO SHOULD NOT CHANGE TO EXCEED ITS QUARTERLY LIMIT.
Constraint Number:	ORG-036-0
Constraint Name:	OPTEMPO-TOTAL-NEGATIVE-CONSTRAINT
Constraint:	A SHIP'S OPTEMPO SHOULD NOT EXCEED ITS QUARTERLY LIMIT.
Constraint Number:	ORG-037-0
Constraint Name:	OPTEMPO-POSITIVE-CONSTRAINT
Constraint:	IT IS DESIRABLE FOR A SHIP'S OPTEMPO TO FALL BELOW ITS QUARTERLY LIMIT.

Constraint Number: ORG-038-0

Constraint Name: PERSTEMPO-NEGATIVE-CONSTRAINT

Constraint: A SHIP'S PERSTEMPO SHOULD NOT CHANGE TO EXCEED 50%.

Constraint Number: ORG-039-0

Constraint Name: PERSTEMPO-TOTAL-NEGATIVE-CONSTRAINT

Constraint: A SHIP'S PERSTEMPO SHOULD NOT EXCEED 50%.

Constraint Number: ORG-040-0

Constraint Name: PERSTEMPO-POSITIVE-CONSTRAINT

Constraint: IT IS DESIRABLE FOR A SHIP'S PERSTEMPO TO FALL BELOW 50%.

Constraint Number: ORG-041-0

Constraint Name: BURNED-FUEL-NEGATIVE-CONSTRAINT

Constraint: DO NOT INCREASE THE AMOUNT OF FUEL THAT A SHIP HAS TO BURN TO MEET ITS SCHEDULE.

Constraint Number: ORG-042-0

Constraint Name: BURNED-FUEL-POSITIVE-CONSTRAINT

Constraint: IT IS DESIRABLE TO HAVE A SHIP'S FUEL CONSUMPTION DECREASE.

Constraint Number: ORG-043-0

Constraint Name: DEPLOY-TIME-NEGATIVE-CONSTRAINT

Constraint: DEPLOYMENT TIMES SHOULD NOT CHANGE TO BECOME GREATER THAN 6 MONTHS.

Constraint Number: ORG-044-0

Constraint Name: DEPLOY-TIME-TOTAL-NEGATIVE-CONSTRAINT

Constraint: DEPLOYMENT TIMES SHOULD NOT EXCEED 6 MONTHS.

Constraint Number: ORG-045-0

Constraint Name: DEPLOY-TIME-POSITIVE-CONSTRAINT

Constraint: IT IS DESIRABLE FOR A SHIP'S DEPLOYMENT TIME TO BECOME SHORTER THAN 6 MONTHS.

Constraint Number: ORG-046-0

Constraint Name: TURN AROUND-RATIO-NEGATIVE-CONSTRAINT

Constraint: A SHIP'S TURN AROUND RATIO SHOULD NOT CHANGE TO BECOME LESS THAN 2:1.

Constraint Number: ORG-047-0

Constraint Name: TURN AROUND-RATIO-TOTAL-NEGATIVE-

CONSTRAINT

Constraint: A SHIP'S TURN AROUND RATIO SHOULD NOT BE LESS THAN 2:1.

Constraint Number: ORG-048-0

Constraint Name: TURN AROUND-RATIO-POSITIVE-CONSTRAINT

Constraint: IT IS DESIRABLE FOR A SHIP'S TURN AROUND RATIO TO BECOME HIGHER THAN 2:1.

Constraint Number: PREF-001-2

Constraint Name: PREF-0001

Constraint: WHEN REPLACING SHIPS IN 3RD FLEET, PREFER A SHIP NOT IN READY BATTLE GROUP.

Comments: HOWEVER, PREFER SHIPS IN RBG TO SHIPS RETURNING FROM DEPLOYMENT.

Constraint Number: PREF-002-2

Constraint Name: PREF-0002

Constraint: WHEN REPLACING A CARRIER, PREFER A NIMITZ CLASS CARRIER.

Relaxations:

1) ENTERPRISE CLASS CARRIER
2) KITTYHAWK CLASS CARRIER
3) FORRESTAL CLASS CARRIER
4) MIDWAY CLASS CARRIER
5) BATTLESHIP

Constraint Number: PREF-003-2

Constraint Name: PREF-0003

Constraint: WHEN REPLACING A CRUISER, PREFER A TICONDEROGA CLASS CRUISER.

Relaxations:

1) LONG BEACH, BAINBRIDGE, OR LEAHY CLASS CRUISER.
2) TRUXTUN OR BELKNAP CLASS CRUISER.
3) VIRGINIA OR CALIFORNIA CLASS CRUISER.
4) DDG 37 OF KIDD CLASS DESTROYER.
5) DDG 2 CLASS DESTROYER.
6) PERRY CLASS FRIGATE. (FFG 7)

Constraint Number: PREF-004-2

Constraint Name: PREF-0004

Constraint: WHEN REPLACING A SPRUANCE CLASS DESTROYER, PREFER ANOTHER SPRUANCE CLASS

DESTROYER.

Relaxations:
1) KIDD CLASS DESTROYER.
2) PERRY CLASS FRIGATE.
3) KNOX CLASS FRIGATE.
4) GARCIA CLASS FRIGATE.

Constraint Number: PREF-005-2

Constraint Name: PREF-0005

Constraint: WHEN REPLACING A GUIDED MISSILE DESTROYER, PREFER A CRUISER.

Relaxations:
1) FARRAGUT CLASS DESTROYER.
2) KIDD CLASS DESTROYER.
3) ADAMS CLASS DESTROYER.
4) PERRY CLASS FRIGATE.
5) BROOKE CLASS FRIGATE.

Constraint Number: PREF-006-2

Constraint Name: PREF-0006

Constraint: WHEN REPLACING A FRIGATE, PREFER A SPRUANCE CLASS DESTROYER.

Relaxations:
1) PERRY CLASS FRIGATE.
2) KNOX CLASS FRIGATE.
3) BROOKE CLASS FRIGATE.
4) BRONSTEIN CLASS FRIGATE.

Constraint Number: PREF-007-2

Constraint Name: PREF-0007

Constraint: IF A BATTLE GROUP HAS A NUCLEAR CARRIER, PREFER TO HAVE AT LEAST ONE NUCLEAR CRUISER TO ACCOMPANY IT.

Constraint Number: PREF-012-1

Constraint Name: SKD-0012

Constraint: DO NOT USE SHIPS IN CATEGORY 6 EMPLOYMENTS FOR REPLACEMENT.

Constraint Number: PREF-013-1

Constraint Name: PREF-0013

Constraint: REPLACING A NUCLEAR SHIP, PREFER A NUCLEAR SHIP FOR REPLACEMENT IF POSSIBLE.

Constraint Number: PREF-015-0

Constraint Name: MA-THRESHOLD-CONSTRAINT

Constraint: ALL SHIPS MUST MEET OR EXCEED THEIR MISSION AREA THRESHOLDS.

Constraint Number: PREF-016-0

Constraint Name: CASREP-THRESHOLD-CONSTRAINT

Constraint: PLATFORMS MUST MEET OR EXCEED ALL CASREP THRESHOLDS.

Constraint Number: PREF-017-0

Constraint Name: CROVL-THRESHOLD-CONSTRAINT

Constraint: PLATFORMS MUST MEET OR EXCEED ALL CROVL THRESHOLDS.

Constraint Number: PREF-018-0

Constraint Name: C-RATING-THRESHOLD-CONSTRAINT

Constraint: PLATFORMS MUST MEET OR EXCEED ALL THRESHOLDS FOR C-RATINGS.

Constraint Number: PREF-019-0

Constraint Name: M-PLATFORM-CHOPS-CONSTRAINT

Constraint: ALL PLATFORM REQUIREMENTS FOR AN ACTIVITY SHOULD BE MET.

Constraint Number: PREF-020-0

Constraint Name: CHECK-ALL-THRESHOLDS-CONSTRAINT

Constraint: PLATFORMS MUST MEET OR EXCEED ALL APPLICABLE THRESHOLD DEFINITIONS.

Constraint Number: SKD-006-2

Constraint Name: SKD-0006

Constraint: A SHIP ASSIGNED TO TYCOM SHOULD NOT BE USED FOR REPLACEMENT.

Comments: DISRUPTION OF TRAINING.

Constraint Number: SKD-007

Constraint Name: SKD-0007

Constraint: REPLACEMENT UNITS OR UNITS ASSIGNED TO NEW MISSIONS SHOULD COME FROM THE SAME FLEET OR THE FLEET IN WHOSE AOR THE MISSION IS TO TAKE PLACE.

Comments: AOR-AREA OF RESPONSIBILITY.

Constraint Number: SKD-042-0

Constraint Name: SKD-0042

Constraint: DO NOT USE SHIPS IN POM (PREPARATION FOR OVERSEAS MOVEMENT) FOR REPLACEMENT.

Constraint Number: SKD-043-1

Constraint Name: SKD-0043

Constraint: NO SURFACE COMBATANTS WITH CROVL WORSE C2 IN THE INDIAN OCEAN.

Comments: INDIAN OCEAN IS TOO FAR AWAY TO HAVE DEGRADED CAPABILITY.

Constraint Number: SKD-044-0

Constraint Name: SKD-0044

Constraint: DO NOT USE SHIPS WITH A CROVL OF 5.

Comments: BY DEFINITION, SHIPS WITH A CROVL OF 5 ARE NOT OPERATIONAL.

RULES

Rule Name: AXIOM-AG-PREPROCESS-004

Rule Type: ALTERNATIVES GENERATION

Left Hand Side: IF ALTERNATIVES ARE BEING GENERATED FOR AN EVENT TRIGGERED BY A CASREP OF CATEGORY 2 OR 3,

Right Hand Side: THEN THE DO-NOTHING OPTION WHICH APPLIES IS "GO WITH DEGRADED CAPABILITY."

Rule Name: AXIOM-AG-PREPROCESS-005

Rule Type: ALTERNATIVES GENERATION

Left Hand Side: IF ALTERNATIVES ARE BEING GENERATED FOR AN EVENT TRIGGERED BY A CASREP OF CATEGORY 4,

Right Hand Side: THEN THE DO-NOTHING OPTION WHICH APPLIES IS "GO WITHOUT EQUIPMENT."

Rule Name: AXIOM-AG-REPLACE-005

Rule Type: ALTERNATIVES GENERATION

Left Hand Side: IF A SHIP IS RAV (RESTRICTED AVAILABILITY),

Right Hand Side: THEN ELIMINATE THE SHIP FROM THE LIST OF REPLACEMENT ALTERNATIVES.

Rule Name: AXIOM-AG-REPLACE-005B

Rule Type: ALTERNATIVES GENERATION

Left Hand Side: IF A SHIP IS BETWEEN MTT-1 AND OPPE IN ITS SCHEDULE, Right Hand Side: THEN ELIMINATE THE SHIP FROM THE LIST OF REPLACEMENT ALTERNATIVES.

Rule Name: AXIOM-AG-PREPROCESS-006

Rule Type: ALTERNATIVES GENERATION

Left Hand Side: IF ALTERNATIVES ARE BEING GENERATED FOR AN EVENT TRIGGERED BY A CHANGE TO ONE OF A SHIPS C-RATINGS OR M RATINGS,

Right Hand Side: THEN THE DO-NOTHING OPTION WHICH APPLIES IS "GO WITH DEGRADED CAPABILITY."

Rule Name: AXIOM-AG-REPLACE-005AA

Rule Type: ALTERNATES GENERATION

Left Hand Side: IF A SHIP IS IN A CATEGORY 1 EMPLOYMENT (CONSTRUCTION AND OVERHAUL),

Right Hand Side: THEN ELIMINATE THE SHIP FROM THE LIST OF REPLACEMENT ALTERNATIVES.

Rule Name: RANK-AXIOM-PREF-0002-RELAX-5A

Rule Type: RANKER

Left Hand Side: IF A BATTLESHIP IS RECOMMENDED AS AN ALTERNATIVE FOR REPLACING A CARRIER AND THERE ARE CARRIERS AVAILABLE WITH GOOD RATINGS,

Right Hand Side: THEN ASSIGN THE BATTLESHIP ALTERNATIVE A NEGATIVE IMPACT OF 2000.

HEURISTICS

1. The order of preference for replacing ships in 7th fleet is a(other ships in 7th fleet, b) ships in 3rd fleet, and c) ships reporting to their type commander.
2. The order of preference for replacing ships in 3rd fleet is a) other ships in 3rd fleet, b) ships in 7th fleet, and c) ships reporting to their type commander.
3. If alternatives are being generated for an event triggered by a CASREP of Category 2 or 3, then the do-nothing option which applies is "Go with degraded equipment."
4. If alternatives are being generated for an event triggered by a CASREP of Category 4, then the do-nothing option which applies is "Go without equipment."
5. If alternatives are being generated for an event triggered by a change to one of a ship's C-ratings or M-ratings, then the do-nothing option which applies is "Go with degraded capability."
6. If a ship is in RAV (Restricted Availability), then eliminate the ship from the list of replacement alternatives.

7. If a ship is in a Category 1 employment (Construction and Overhaul), then eliminate the ship from the list of replacement alternatives.
8. If a ship is between MTT-1 and OPPE in its work-up schedule, then eliminate the ship from the list of replacement alternatives.

APPENDIX B

The following represents the distilled results from knowledge engineering efforts with CINCLANT Fleet Command Center (AFCC) schedulers. The results provided here are taken from surveys focused directly at the documented FRESH rule base. Only those responses that differ from the current CINCPACFLT Fleet Command Center (PFCC) FRESH rules are included. The rule number of that rule which is directly affected in FRESH is provided first, followed by the response given by the CINCLANT staff with their remarks where applicable. For ease of direct comparison with the rules in Appendix A, the responses below are presented in the same order as the FRESH rules provided in Appendix A. Therefore, the reader need only note the rule number found here and locate the same rule number in Appendix A. A thorough study of these two appendices provides a high level view of the differences in procedural bases used at the two FCC's.

CONSTRAINTS

ORG-035-0, ORG-036-0

SUGGESTED RULE: A SHIP'S OPTEMPO SHOULD NOT EXCEED ITS QUARTERLY LIMIT.

CINCLANT RESPONSE—DISAGREE.

REMARKS: OPTEMPO is based on fuel availability and subject to deployment work-up cycle.
Researcher's Note: CINCLANT feels that complete utilization of a unit's fuel quota is more significant than meeting OPTEMPO limits.

ORG-037-0

SUGGESTED RULE: IT IS DESIRABLE FOR A SHIP'S OPTEMPO TO FALL BELOW ITS QUARTERLY LIMIT.

CINCLANT RESPONSE—DISAGREE

REMARKS: You must burn the fuel or lose it next year.

ORG-038-0, ORG-039-0

SUGGESTED RULE: A SHIP'S PERSTEMPO SHOULD NOT EXCEED 50%.

CINCLANT RESPONSE—DISAGREE

Researcher's Note: While the AFCC staff would agree that it is desirable for PERSTEMPO to be less than 50%, they disagree on it never exceeding 50%. This is obviously tied to OPTEMPO and its governing fuel availability.

ORG-041-0, ORG-042-0

SUGGESTED RULE: DO NOT INCREASE THE AMOUNT OF FUEL THAT A SHIP HAS TO BURN TO MEET ITS SCHEDULE.

CINCLANT RESPONSE—AGREE

Researcher's Note: AFCC staff feels the system should defer reasoning on a problem which is firing this rule to human intervention.

PREF-001-2

CINCLANT METHODOLOGY—REPLACING SHIPS IN 2ND FLEET UNDEFINABLE.

Researcher's Note: No direct hierarchy of replacement ship categories were felt to be specifically identifiable. AFCC staff felt this decision should be weighted as to DEFCON Mission and PERSTEMPO.

PREF-002-2

SUBSTITUTION HIERARCHY FOR REPLACING A CARRIER.

1) ENTERPRISE CLASS CARRIER
2) NIMITZ CLASS CARRIER
3) KITTYHAWK CLASS CARRIER
4) FORRESTAL CLASS CARRIER
5) MIDWAY CLASS CARRIER
6) BATTLESHIP

REMARKS; The above replacement hierarchy is general in nature only. The AFCC staff would prefer a hierarchy tied specifically to each class of aircraft carrier being replaced.

PREF-004-2

SUBSTITUTION HIERARCHY FOR REPLACING A SPRUANCE CLASS DESTROYER.

1) SPRUANCE CLASS DD
2) KIDD CLASS DDG
3) KNOX CLASS FF
4) O.H. PERRY CLASS FFG
5) CHARLES F. ADAMS CLASS DDG

REMARKS: Prefer equipment capability matching where possible. Replacement of lost Spruance class equipped with "X" equipment suite to be replaced by similarly equipped Spruance realizing great variance may exist within a class of ships.

PREF-005-2

SUBSTITUTION HIERARCHY FOR REPLACING A GUIDED MISSILE DESTROYER.

1) KIDD CLASS DDG
2) FARRAGUT CLASS DDG
3) CHARLES F. ADAMS CLASS DDG
4) O.H. PERRY CLASS FFG
5) BROOKE CLASS FFG

PREF-006-2

SUBSTITUTION HIERARCHY FOR REPLACING A FRIGATE.

1) KNOX CLASS FF
2) O.H. PERRY CLASS FFG
3) SPRUANCE CLASS DD WITH ACOUSTIC TAIL
4) CHARLES F. ADAMS CLASS DDG

PREF-012-1

CINCLANT METHODOLOGY—THE DECISION TO USE SHIPS UNDER GOING MAJOR INSPECTION AS REPLACEMENT UNITS SHOULD BE FLAGGED FOR HUMAN INTERVENTION AS POSSIBLE SELECTION CANDIDATES.

PREF-013-1

SUBSTITUTION HIERARCHY FOR REPLACING A NUCLEAR SHIP IN ESCORT.

1) TICONDEROGA CLASS CG
2) LEAHY CLASS CG
3) ANY CGN CLASS
4) KIDD CLASS DDG
5) FARRAGUT CLASS DDG

PREF-016-0

CINCLANT METHODOLOGY—PLATFORMS SHOULD NOT BE SYSTEM CASREP BELOW C 2 FOR ANTICIPATED AREA OF SUPPORT TASKED.

PREF-017-0

CINCLANT METHODOLOGY—PLATFORMS MUST MEET OR EXCEED AN OVERALL CROVL RATING OF C 2

SKD-006-2

CINCLANT METHODOLOGY—A SHIP ASSIGNED TO TYCOM SHOULD MAY BE CONSIDERED AS A REPLACEMENT UNITS IF REQUIRED.

SKD-042-0

CINCLANT METHODOLOGY—SHIPS IN POM (PREPARATION FOR OVERSEAS MOVEMENT) MAY BE CONSIDERED FOR REPLACEMENT USE IF REQUIRED.

SKD-043-1

CINCLANT METHODOLOGY—ANY UNIT ASSIGNED TO FORWARD DEPLOYMENT WILL BE C-2.

SKD-044-0

CINCLANT METHODOLOGY—DO NOT USE SHIPS IN ANY CASE WITH A CROVL OF LESS THAN C-3 AS A POSSIBLE REPLACEMENT UNITS.

Researchers Note: CINC's approval may be required for use of C-3 units.

RULES

AXIOM-AG-PREPROCESS-005

SUGGESTED RULE: GENERALLY FOR A SHIP DESIGNATED AS C-4 WOULD PREFER TO SEND SHIP AS IS RATHER THAN CANCEL IF NO SUITABLE REPLACEMENT IS EVIDENT.

CINCLANT RESPONSE—DISAGREE

AXIOM-AG-REPLACE-005

SUGGESTED RULE: IF A SHIP IS DESIGNATED AS RAV (RESTRICTED AVAILABILITY), ELIMINATE THE SHIP.

CINCLANT RESPONSE—DISAGREE, SENSITIVITY OF THE REQUIREMENT MUST BE WEIGHED BY THE STAFF.

AXIOM-AG-REPLACE-005B

CINCLANT METHODOLOGY—SENSITIVITY OF THE REQUIREMENT MUST BE WEIGHED BY THE STAFF.

Other substitution hierarchies not represented in the FRESH system which are suggested.

SUBSTITUTION HIERARCHY FOR REPLACING A KIDD CLASS DESTROYER WOULD BE DEPENDENT OF THE OTHER UNITS IN THE BATTLE GROUP AND THE EQUIPMENT SUITE PRESENT ON THE DD's ie. TAIL, TOMAHAWK, ETC.

SUBSTITUTION HIERARCHY FOR REPLACING A FFG.

1) O.H. PERRY CLASS FFG
2) CHARLES F. ADAMS CLASS DDG
3) BROOKE CLASS FFG

SUBSTITUTION HIERARCHY FOR REPLACING A BATTLESHIP.

1) ANOTHER BB
2) A CRUISER

Thesis
L89274 Luigart
c.1 A proposal for the transfer of a large force management expert system (FRESH) from the CINCPACFLT Command Center to the CINCLANTFLT Command Center.