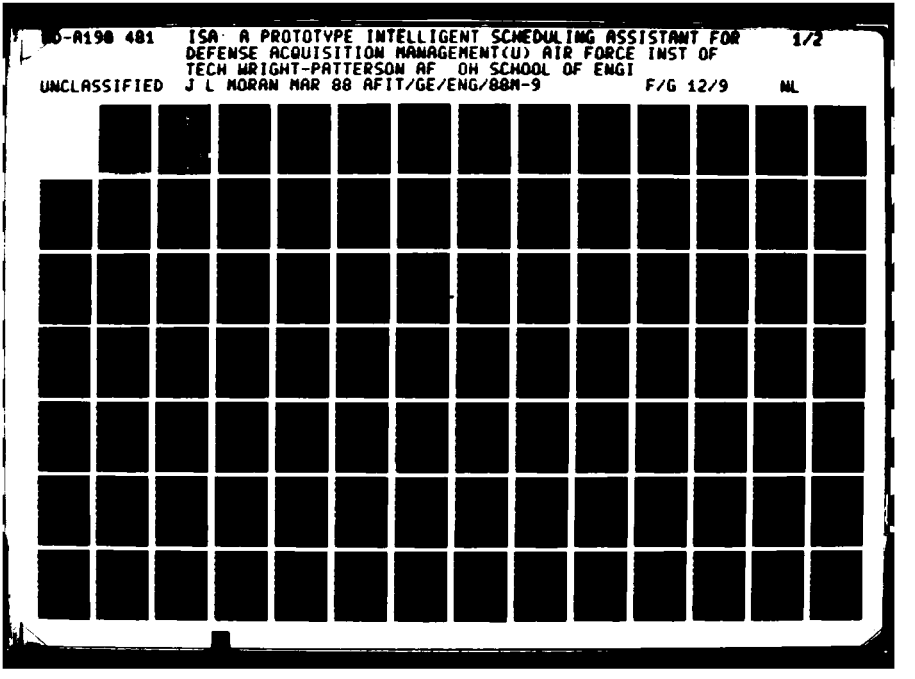
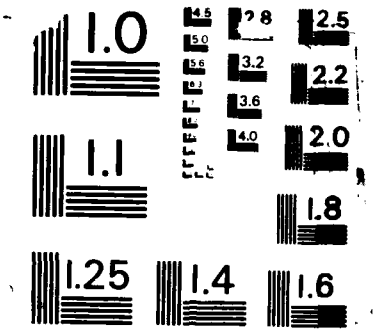

D-A190 481 ISA: A PROTOTYPE INTELLIGENT SCHEDULING ASSISTANT FOR 1/2
DEFENSE ACQUISITION MANAGEMENT(U) AIR FORCE INST OF
TECH WRIGHT-PATTERSON AF OH SCHOOL OF ENGI
UNCLASSIFIED J L MORAN MAR 88 AFIT/GE/ENG/88M-9 F/G 12/9 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

DTIC FILE COPY

AD-A190 481

DTIC
ELECTE
MAR 28 1988
S D
E

DEPARTMENT OF THE AIR FORCE

AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

This document has been approved for public release and sale; its distribution is unlimited.

88 3 24 087

AFIT/GE/ENG/88M-9

ISA

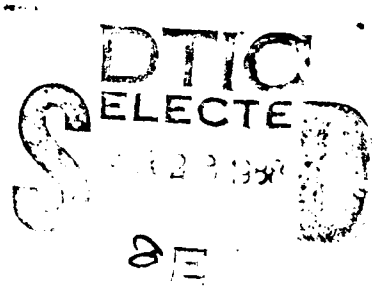
A PROTOTYPE INTELLIGENT SCHEDULING ASSISTANT

FOR DEFENSE ACQUISITION MANAGEMENT

THESIS

Jerry L. Moran
Captain, USA

AFIT/GE/ENG/88M-9



Approved for public release; distribution unlimited.

AFIT/GE/ENG/88M-9

ISA

A PROTOTYPE INTELLIGENT SCHEDULING ASSISTANT FOR DEFENSE ACQUISITION MANAGEMENT

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Electrical Engineering

DTIC COPY INSPECTED

Accession For
NTIS GRA&I
DTIC TAB
Unannounced
Justification
By
Distribution/
Availability Codes
Dist | Avail and/or Special
A-1

Jerry L. Moran

Captain, USA

March 1988

Approved for public release; distribution unlimited.

Preface

This thesis investigated the application of artificial intelligence to defense acquisition management. The goal was to demonstrate how knowledge-based methods could improve scheduling of defense acquistion programs. The objectives were 1) to determine the kind of knowledge needed to tailor schedules and 2) to develop a framework for using that knowledge to generate tailored schedules. ISA, a prototype Intelligent Scheduling Assistant, successfully shows how knowledge used to tailor program schedules can be captured in a rule-based system.

I am indebted to many people for the assistance offered in support of my thesis. I am grateful to my thesis advisor, Captain Wade H. Shaw Jr., for the autonomy granted, encouragement provided, and assistance given in support of my efforts. Sincere thanks go to James W. Roe and Leroy Verbillion for helping me identify a meaningful problem to solve and locate a good organization to work with. I would like to thank ASD/RW for the excellent support provided for my research. Many thanks to all who contributed to my thesis.

My deepest gratitude goes to my wife, Audrey, and children, Justin and Andrea, for their understanding and emotional support when my thesis demanded all of my attention.

Jerry L. Moran

Table of Contents

		Page
Preface		ii
List of Figures		vi
List of Tables		vii
Abstract		viii
I.	Introduction	1
	Background	1
	Problem	2
	Scope	2
	Approach	3
	Sequence of Presentation	4
II.	Review of Scheduling Techniques	5
	Overview	5
	Gantt Charts	5
	PERT/CPM Networks	6
	Computer-based Scheduling Tools	6
	Model Networks	7
	Knowledge-based Scheduling Techniques	9
	Summary	10
III.	Problem Assessment	12
	Overview	12
	Scheduling in ASD/RW	12
	Scheduling Process	13
	Scheduling Problems	14
	Scheduling Impact	15
	Expert System Potential	15
	Feasibility	16
	Suitability	17
	Desirability	19
	Summary	20
IV.	Knowledge Engineering	21
	Overview	21
	Knowledge Engineering Interviews	22
	Knowledge Engineering Results	23
	Summary	24

Page

V. Prototype Design and Tool Selection 25

Overview 25
System Requirements 25
Prototype Design 28
Tool Selection 30
Summary 31

VI. Prototype Development 33

Overview 33
Data Structures 33
User Interface 39
CSNAS Interface 40
Input of Program Characteristic Values 41
Schedule Generation 42
Schedule Modification 51
Alternative Approaches 52

VII. Prototype Evaluation 54

Overview 54
Prototype Testing 54
User Evaluation 56
Summary 57

VIII. Conclusions and Recommendations 59

Overview 59
Conclusions 59
Recommendations for Future Research 61
Modifying Tasks 62
Analyzing Schedules 63
Providing Reasoning 64
Automating Rule Generation 65
Summary 65

Appendix A: Problem Assessment Interviews 66

Appendix B: Tasks 73

Appendix C: Program Characteristics 85

Appendix D: Rules for Schedule Generation 88

Appendix E: ISA Evaluation Questionnaire 120

Appendix F: AFALC Letter 122

Page

Bibliography 123

Vita 125

List of Figures

Figure		Page
1.	Conceptual View of ISA	25
2.	Database Structure	34
3.	Example Prompt for a Program Characteristic Value	42
4.	Data Flow Between Working Tables and Rule Sets	43
5.	Rule for "DRAFT PMD"	45
6.	Rule for "THREAT INPUT"	46
7.	Rule for "DRAFT SPEC"	47
8.	Rule for "FINAL SOW"	49
9.	Summary of Responses to Primary Criteria	56

List of Tables

Table		Page
I.	System Requirements Matrix	29
II.	Program Table	35
III.	Characteristics Table	36
IV.	Task Table	37
V.	Link Table	38
VI.	Cross-tabulation of Concept and Improvement	57
VII.	Tasks and Relationships	73
VIII.	Tasks Which may not Apply	75
IX.	Program Characteristics and Acceptable Values	85
X.	Cross-reference of Characteristics and Tasks	86

Abstract

This thesis investigated the application of artificial intelligence to defense acquisition management. The goal was to demonstrate how knowledge-based methods could improve scheduling of defense acquisition programs. The objectives were 1) to determine the kind of knowledge needed to tailor schedules and 2) to develop a framework for using that knowledge to generate tailored schedules.

Scheduling of defense acquisition programs is a difficult problem for which expert systems are an appropriate solution methodology. This thesis identified 35 characteristics of a defense acquisition program which affect the applicability, duration, and relationships of tasks required to go from receipt of a Program Management Directive to contract award. It extends the model network concept used in the Aeronautical Systems Division and the Air Force Acquisition Logistic Command of the United States Air Force.

ISA, a prototype Intelligent Scheduling Assistant, successfully shows how knowledge used to tailor program schedules can be captured in a rule-based system. ISA uses the values of acquisition program characteristics to generate tailored schedules. The concept is applicable to any project schedule.

ISA

A PROTOTYPE INTELLIGENT SCHEDULING ASSISTANT

FOR DEFENSE ACQUISITION MANAGEMENT

I. Introduction

Background

The average defense acquisition program experiences cost growth of 50 percent to 100 percent while program deliveries slip 30 percent. Cost growth and schedule slippage are due primarily to instability in establishing requirements, planning, and budgeting for weapon systems. (Gansler, 1986:1)

Scheduling can reduce instability in a program. Scheduling is a process of deciding what work needs to be done, who will be responsible for the work, when the work should be done, and what order the work should be done in. Network schedules provide graphical representations of plans which can be used to indicate progress, identify problems, and aid communications (Woffinden, 1987).

Despite its advantages, scheduling is difficult because no two defense acquisition programs are the same. This variability makes it difficult to decide what work is needed, who should be responsible, how long it should take, and what order it should be done in. Experience is often the best guide to developing a schedule. However, due to the high rotation rate among both military and civilian personnel, many people assigned to defense acquisition programs are new and inexperienced (Gansler, 1986:9).

The Aeronautical Systems Division (ASD) of the Air Force Systems Command (AFSC) uses model networks (schedules) to alleviate problems caused by inexperience. Model networks (described further in Chapter II) attempt to capture the expertise of experienced program personnel. Model networks must be tailored to specific programs. Tailoring is the process of deciding which tasks apply, specifying who is responsible for each task, estimating how long each task takes, and making appropriate changes in the order of the tasks.

Problem

Model networks are difficult to use because they are too general and lack sufficient tailoring guidance. Model networks include every conceivable task that may occur in the program schedule. Some of these tasks are mutually exclusive and may not occur in the same network. Suggested task durations may vary widely. Unfortunately, model networks lack sufficient guidance to help inexperienced personnel make tailoring decisions. Thus, users often find tailoring model networks to be a difficult and time-consuming process.

The goal of this thesis was to demonstrate how knowledge-based methods could improve scheduling of defense acquisition programs. The objectives were 1) to determine the kind of knowledge needed to tailor schedules and 2) to develop a framework for using that knowledge to generate tailored schedules.

Scope

This thesis developed a demonstration prototype for generating tailored schedules for awarding defense contracts.

Approach

An existing model network consisting of 48 tasks and covering 28 months was used as a starting point for this thesis. The thesis effort was divided into the six distinct phases typical of many knowledge-based system developments. These phases were problem assessment, knowledge acquisition, prototype design, tool selection, prototype development, and prototype evaluation.

The problem assessment phase was used to acquire a thorough understanding of the scheduling and tailoring processes. A literature search, focusing on project scheduling, was used to gain a better understanding of scheduling techniques. Program management personnel, defense contractors, and individuals with scheduling knowledge were interviewed to gain their insights on the use of network schedules and model networks. The data collected was used to assess the potential for developing an expert system for generating tailored schedules.

The knowledge acquisition phase was used to determine the specific knowledge needed to make tailoring decisions. Program management personnel were interviewed to identify the kinds of data, knowledge, and procedures used to tailor tasks in the 48-task model network.

The prototype design phase was used to establish system requirements and design the demonstration prototype. The design was approached from a functional point of view and considered twelve principles for the design of interactive computer systems.

The tool selection phase was used to select tools for the demonstration prototype. Tool selection was based on availability, power,

sophistication, support facilities, reliability, maintainability, and tool features.

The prototype development phase was used to develop the demonstration prototype. The prototype implements that subset of the system requirements deemed necessary to demonstrate the feasibility of an operational system. The prototype was developed using an incremental approach.

The prototype evaluation phase was used to evaluate the demonstration prototype. The prototype was evaluated by testing its ability to generate tailored schedules. The prototype was also evaluated by potential users.

Sequence of Presentation

Chapter II reviews conventional scheduling methods, computer-based scheduling tools, model networks, and knowledge-based scheduling techniques. Chapter III assesses the potential for using expert systems to solve scheduling problems in ASD/RW (Deputy for Reconnaissance/Strike and Electronic Warfare). Chapter IV describes the knowledge acquisition process. Chapter V discusses system requirements, prototype design, and tool selections. Chapter VI describes the implementation of the prototype. Chapter VII presents the results of the prototype evaluation. Chapter VIII summarizes the results of the thesis and makes recommendations for further research.

II. Review of Scheduling Techniques

Overview

This chapter reviews conventional scheduling methods, computer-based scheduling tools, model networks, and knowledge-based scheduling techniques. Schedules are typically represented using bar charts or activity networks. The most popular form of bar chart is the Gantt chart. Activity networks are used in the Critical Path Method (CPM) and the Program Evaluation and Review Technique (PERT). A wide variety of computer-based scheduling tools which use bar charts and/or activity networks have been developed. Model networks are used by ASD and others to augment PERT-based schedule development. Researchers in artificial intelligence have investigated techniques for developing knowledge-based plans and schedules.

Gantt Charts

During World War I, Henry L. Gantt developed the Gantt chart as a visual aid for monitoring production performance. Gantt charts show activities in the form of a bar chart and became an accepted business tool by World War II (Fersko-Weiss, 1987:166). They are used to show the progress of individual activities in relation to a fixed time scale and exist in many forms.

Gantt charts offer three main advantages. They graphically show the important activities without clutter and detail, graphically show the progress of activities with respect to time, and are easy to read and understand. (Woffinden, 1987)

Gantt charts have two main disadvantages. They do not explicitly show inter-relationships between activities or the dynamics of activities. (Woffinden, 1987)

PERT/CPM Networks

Dr. John Mauchly developed CPM in the late 1950s to identify critical tasks in a project. Critical tasks are those activities that must be completed on time in order to meet project deadlines. PERT, developed by Willard Frazer, calculates activity durations using three estimates - best case, worst case and most likely case. Both CPM and PERT use flow diagrams (activity networks commonly referred to as PERT networks) to represent the schedule. (Fersko-Weiss, 1987:166)

PERT networks offer several advantages. They aid communication by graphically showing the work to be done, who is responsible, activity durations, and inter-relationships between activities. They improve the understanding of complex projects when activities are clearly understood and valid durations can be estimated. Most important, they help identify potential problems by showing the impact of late tasks on the schedule. (Woffinden, 1987)

PERT networks have some disadvantages. They require accurate duration estimates, are difficult to use manually for complex projects, and do not address the dynamics of project activities. (Woffinden, 1987)

Computer-based Scheduling Tools

Computer programs for project management were first developed in the 1950s during the U.S. Navy's Polaris project. Commercial marketing of mainframe software for CPM applications, used primarily to manage

large projects in construction, started in the 1960s. The first commercial personal computer (PC) version of project management software, the Harvard Project Manager, was introduced in 1983. More than 100 PC-based project management programs, ranging from $35 to almost $8000, are commercially available today. (Fersko-Weiss, 1987:153,166)

The Air Force Acquisition Logistics Command (AFALC) developed the Computer Supported Network Analysis System (CSNAS) as a PERT/CPM-based interactive scheduling system for program management. CSNAS currently has more than 100 users, is written in FORTRAN, and is available in mainframe (HP3000 and VAX) and PC (IBM-compatible and Zenith) versions. (Clark, 1987)

The Defense Systems Management College (DSMC) is developing the Program Manager's Support System (PMSS) to support the decision-making process of the PM. Designed for Zenith and IBM-compatible PCs, PMSS uses decision support systems technology. PMSS contains modules that develop Gantt charts, develop PERT networks, perform risk analysis, and do milestone management. (Scanlon and Schutt, 1987)

All computer-based scheduling tools require the user to determine and input activities, durations and relationships for a project. The software determines timing and produces a graphical representation of the project schedule based on these inputs. Thus, the user must make all scheduling and tailoring decisions independent of the project management software used.

Model Networks

CSNAS contains a database of model networks for weapons system acquisition programs (AFALC, 1987:234). Various organizations in ASD

have developed their own model networks for internal use. A model network typically consists of a CSNAS data file, a PERT network, and a set of reference sheets. The CSNAS data file, containing the schedule information, can be copied and tailored to a specific program. The PERT network graphically portrays the generic schedule. The reference sheets provide brief descriptions (duration, participants, references and other information) of each task.

The Deputy for Reconnaissance/Strike and Electronic Warfare (ASD/RW) is developing a model network for long term programs and training new program managers in ASD/RW. The model network, split into two phases, attempts to capture the corporate experiences of ASD/RW. The Phase I network, completed in August 1985, covers 48 activities and 28 months from receipt of a Program Management Directive (PMD) to contract award. The Phase II network, undergoing validation, covers 76 activities and 8 years from contract award to Program Management Responsibility Transfer (PMRT). (Zornes, 1987)

The Deputy for Aeronautical Equipment (ASD/AE) used model networks to develop the Program Office Internal Networking and Tracking System (POINTS). Derived from a historical review of 32 programs in ASD/AE, POINTS contains model networks for use by program and functional managers in ASD/AE. (ASD/AE, 1987)

Model networks include every conceivable task that may occur for a given defense acquisition program. Some of these tasks are mutually exclusive and may not occur in the same network. Some task durations vary up to a year between pessimistic and optimistic estimates. Unfortunately, model networks offer little tailoring guidance to the user.

Thus, model networks are difficult and time-consuming for the inexperienced person to use, and they have failed to gain acceptance by their intended users.

Knowledge-based Scheduling Techniques

Scheduling is a difficult, knowledge-intensive problem for two main reasons. First, many potential schedules may be developed using different combinations of tasks, durations, and relationships. Second, the evaluation of the quality of a schedule is hard to assess. (Fox, B.R. and Kempf, 1985:487-488)

Early AI research focused on developing general techniques for solving problems. The General Problem Solver (GPS) uses states, differences between states, and operators for changing states to represent problems. The key to solving a problem using GPS is to select operators which reduce the differences between the current state and the goal state. The Stanford Research Institute implemented GPS in STRIPS - a system for generating plans as a sequence of operators. (Winston, 1977:131,137-143)

Sacerdoti used procedural networks to show that plans could be created that allow operations to execute in parallel. His approach, implemented in Nets of Action Hierarchies (NOAH), recursively expands a goal step into a network of sub-steps which achieve the goal. Procedural networks are similar in structure to PERT networks. (Sacerdoti, 1975:206-208)

Procedural networks have been enhanced for a number of knowledge-based planners/schedulers. NONLIN (Non-linear planner) was developed by

Tate as an aid for constructing project networks (Tate, 1977). Wilkins and Robinson developed SIPE (System for Interactive Planning and Execution) as a domain-independent interactive planner (Wilkins and Robinson, 1981). PLANNER'S WORKBENCH, developed by a group headed by Hayes-Roth, is an aid for re-planning (Hayes-Roth and others, 1981). DEVISER, a general-purpose planner/scheduler built by Vere, provides a capability for adding time constraints to a procedural network (Vere, 1983).

Fox used constraint-directed search in ISIS (Intelligent Scheduling and Information System) to construct job-shop schedules. He discovered that human schedulers spend 80-90% of their time trying to determine what constraints apply to schedule generation. ISIS was designed to develop schedules that satisfy as many constraints as possible in near real-time. (Fox, Mark S., 1983)

Many other knowledge-based planning/scheduling techniques have been investigated. Hayes-Roth simulated errand-planning using a blackboard approach (Hayes-Roth and others, 1979). Fukumori generated train schedules using range-constriction search (Fukumori, 1980). Gazdik combined fuzzy sets and graph theory in FNET to handle uncertainty associated with scheduling (Gazdik, 1983). Sage used multiple criteria decision making to solve scheduling problems (Sage, 1984). Newman and Kempf used opportunistic scheduling to develop schedules for a robot that tends machines in a manufacturing plant (Newman and Kempf, 1985).

Summary

The most popular conventional scheduling techniques are Gantt charts and CPM/PERT networks. Gantt charts are easy to understand while CPM/PERT networks provide a better graphical representation of complex

projects. A wide variety of commercial and public domain project management software is available to handle the actual mechanics of computing schedule timing and producing network drawings. However, the user must make all schedule and tailoring decisions prior to using the software. ASD uses model networks to give the user a starting point for making schedule decisions. However, difficulties in tailoring the model networks have led to a general lack of acceptance.

AI research focuses on the use of knowledge to develop plans and schedules. Procedural networks have been used extensively to develop general-purpose planners/schedulers. Constraint-directed search has been used to develop job-shop schedules which satisfy as many constraints as possible in near real-time. Many other knowledge-based scheduling techniques have been investigated.

This thesis uses knowledge-based techniques to extend the model network concept. The prototype Intelligent Scheduling Assistant (ISA) generates tailored schedules using program knowledge and task knowledge. The results are exported to CSNAS for actual schedule computation and network drawings.

III. Problem Assessment

Overview

This chapter assesses the potential for using expert systems to solve scheduling problems in ASD/RW. An experience survey was conducted by interviewing seventeen people with various degrees of experience and responsibility for scheduling. Four upper-level and middle-level managers in ASD, two program managers in ASD/RW, three schedulers in ASD, two people who worked on ASD/RW's Phase I model network, three contractors who provide scheduling support to ASD/RW, and three people outside of ASD were interviewed. The results of the problem assessment interviews are contained in Appendix A.

The remainder of this chapter summarizes the results of the interviews. First, the current state of scheduling in ASD/RW is presented. Then the potential for using expert systems to improve model networks is examined in terms of feasibility, suitability, and desirability. Finally, the results of the problem assessment are summarized at the end of the chapter.

Scheduling in ASD/RW

ASD/RW is a matrix organization made up of systems directorates and functional area directorates . Functional area personnel support programs in the systems directorates. One person may support multiple programs. The systems directorates are Strike SPO (RWN), Reconnaissance Programs (RWQ), Directorate of Electronic Combat (RWW), Inter-Command Electronic Warfare Management (RWA), and Special Projects SPO (RWZ). The

functional areas are Manufacturing & Quality Assurance (RWD), Engineering (RWE), Contracts (RWK), Logistics (RWL), Program Control (RWP), Safety (RWS) and Acquisition Support (RWB).

Scheduling Process.

Program management teams, consisting of a PM and functional area representatives, are formed early in a program. Some programs simply use suspense calendars to keep track of events. Others use milestone charts to schedule the program. The teams that use network schedules usually tailor schedules from similar programs or build a schedule from scratch. Some programs hire contractors to do scheduling. Only two instances were found where a team had tried to use the Phase I network. Both of these efforts failed.

When modifying the Phase I network, the program management team analyzed network tasks, durations, and relationships. The analysis was done individually or in team meetings. Tasks were reviewed to determine which, if any, did not apply to the program. New tasks were added to the network during the review process. Task durations were reviewed to determine if changes were needed. Task relationships were examined for changes which would maximize parallel execution of tasks. The schedule was modified to reflect all changes, and the review process iterated until a general consensus was reached.

Both efforts at using the Phase I network failed due to insufficient tailoring guidance. Although a reasonable approach was used, most members of the team lacked sufficient experience to select appropriate tasks, durations, and relationships. Thus, tailoring the network proved to be a difficult and time-consuming task that ultimately failed.

Scheduling Problems.

Four factors contribute to scheduling problems in ASD/RW. First, most programs lack people who are experienced in developing schedules. Second, schedules are used improperly. Third, programs operate with limited resources and manpower. Fourth, many defense acquisition programs are in a constant state of flux.

Most programs lack people who are experienced in developing schedules. The Phase I model network was conceived to leverage the corporate experience in ASD/RW and give inexperienced people a starting point for developing schedules. However, many program personnel are unaware of the Phase I network. Others do not use the Phase I network because they find it too difficult and time-consuming to tailor the network and/or use CSNAS. Clearly, the Phase I network has failed to gain acceptance from its intended users.

Schedules are used improperly due to misunderstandings of the purpose of network schedules. A common error is to artificially change durations and/or relationships to reduce schedule length. Another error is to artificially increase activity durations to provide a buffer against unforeseen problems. In either case, the purpose of the schedule is defeated since it does not accurately reflect program plans. Network schedules should be used to determine program status and identify potential problems. Early identification allows the manager to act to avoid known problems rather than react to unforeseen problems using "crisis management".

Programs operate with limited resources and manpower. Meanwhile, the creation of schedules is a time-consuming and manpower-intensive

process. Thus, even program managers who would like to use network schedules are often unable to due to a lack of resources. In cases where funding is available to hire contractors, program personnel must devote time to coordinating with the contractor if the schedule is to accurately reflect the status of the program.

Many defense acquisition programs are in a constant state of flux. Rapidly changing requirements complicate the maintenance of accurate schedules. Good schedules are quickly made obsolete unless efforts are made to maintain and update schedules on a periodic basis. Some changes in requirements may require wholesale changes in the schedule. Due to the lack of resources, schedules tend to be ignored until they are invalid.

Scheduling Impact.

Schedule development and maintenance is a time-consuming and manpower-intensive process. It takes weeks of coordination effort to develop a realistic program schedule. Considerably more effort is needed to manage and maintain a schedule. Regardless of the cause, unrealistic program schedules lead to increased program costs and unforeseen schedule delays.

Expert System Potential

A conventional computer program manipulates data using algorithms. A knowledge-based system is a computer program that manipulates knowledge using heuristics. An expert system is a knowledge-based system which performs at the level of an expert in a specific problem domain. (Waterman, 1986:24-30)

Expert systems could reduce problems of inexperience by capturing the corporate knowledge and experiences of ASD/RW. Expert systems could reduce unintentional misuse of schedules by generating realistic schedules based on program knowledge and task knowledge. Expert systems could be used to suggest alternatives for resolving schedule problems. Expert systems could reduce resource requirements for schedule development by reducing the demand on experienced personnel within ASD/RW. Expert systems could facilitate schedule changes caused by changes in requirements. Expert systems could improve the quality and consistency of schedule development.

Each of the scheduling problems addressed are inter-related. However, the remainder of this assessment will focus on the lack of experience in developing schedules. In particular, the potential for using expert systems to improve model networks will be assessed.

Waterman suggests that expert systems should be considered only when expert systems are possible, appropriate, and justified (Waterman, 1986:127). Similarly, Prerau offers more than fifty factors to consider when evaluating potential expert system applications (Prerau, 1985:26-30). The factors suggested by both authors boil down to evaluating the feasibility, suitability and desirability of building an expert system. Thus, the potential for using expert systems to improve model networks is evaluated in terms of feasibility, suitability and desirability.

Feasibility.

Building an expert system is feasible when expertise is available to the developer and current expert system technology enables the developer to capture the expertise in a computer program. Although expertise

might exist, the developer may not have access to it when the expert is inarticulate or unavailable. Expertise can be captured using current technology for problems which require primarily cognitive skills, can be taught to inexperienced personnel, and take from a few minutes to a few hours to solve. (Waterman, 1986:128-129 and Prerau, 1985:26-30)

Expertise is not specifically available for tailoring the Phase I model network. However, a wealth of knowledge and experience for selecting tasks, durations , and relationships is available from experienced program managers and functional area personnel. Regulations, policy letters, and operating instructions provide another source of knowledge. The Phase I network itself contains a great deal of knowledge about required tasks, durations and relationships.

The expertise needed to make schedule decisions can be captured using current AI technology. Scheduling requires only cognitive skills and can be taught to inexperienced personnel. Scheduling can be decomposed into manageable sub-tasks which take a few minutes to a few hours to solve.

An expert system is feasible. Although experts for tailoring the Phase I network do not exist, ASD/RW has many individuals who are experienced in various aspects of the acquisition process. Thus, an expert system can be built to generate schedules by combining the knowledge of these individuals.

Suitability.

Building an expert system is suitable when the nature of the problem makes AI solutions appropriate and the system is appropriate for implementation. AI solutions are appropriate when the domain is rela-

tively stable, symbolic manipulation is required, heuristics are used, the problem is not trivial, and an explanation capability is desired. Expert systems are appropriate for implementation when the system can be phased in over time, non-optimal solutions are acceptable, and the system is testable. (Waterman, 1986:128-129 and Prerau, 1985:26-30)

AI solutions are appropriate for scheduling defense acquisitions. Although schedules may change quite often, scheduling knowledge such as regulatory requirements changes much less often. Thus, defense acquisition provides a relatively stable domain. Scheduling requires symbolic manipulation and uses heuristics to achieve solutions. It takes years of experience to develop proficiency in scheduling. The ability to document decisions and provide explanations is desirable.

An expert system is appropriate for implementation in ASD/RW. The expert system could be phased in over time as knowledge bases are developed for each functional area. Non-optimal solutions are acceptable. Indeed, since requirements and funding are likely to change, the generation of an optimal schedule is not needed. The greatest difficulty for implementation lies in the testing of the expert system. Because a unique correct solution for any program does not exist, the reasonableness of any solution is a subjective judgment. The subjectiveness of the solution makes explanation capabilities very important when the solution is reviewed for acceptability.

An expert system is suitable. Scheduling is a non-trivial problem which demands the use of heuristics. Due to the subjective nature of solutions, explanation facilities are important. An expert system could be phased in and non-optimal solutions are acceptable.

Desirability.

Building an expert system is desirable when a need for the expert system exists, benefits can be shown, and resources are available both to develop and maintain the expert system. Expert systems are not needed when expertise is readily available and relatively inexpensive. The benefits of an expert system should outweigh the costs of development and maintenance. (Waterman, 1986:128-129 and Prerau, 1985:26-30)

A need for the expert system exists. ASD/RW is responsible for more than 30 defense acquisition programs and lacks sufficient numbers of experienced schedulers to cover all of these programs. They also lack sufficient funding to contract out scheduling for all of the programs. Thus, an expert system would help leverage the corporate experience in ASD/RW.

An expert system could provide a variety of benefits. It could save the time of experienced personnel and reduce coordination efforts. It could provide a degree of consistency in schedule generation. It could provide a record of the decision process that could be used when reviewing the schedule for endorsement.

ASD/RW has limited resources for expert system development and maintenance. However, a prototype expert system can be developed with current resources. ASD currently has sufficient computer resources to support the development and maintenance of an operational expert system. The price of expert-system building tools range from a few thousand to many thousands of dollars. Development and maintenance would require one or two people to work full-time.

An expert system is desirable. A need for the expert system exists, the benefits appear to outweigh the costs, and the organization has sufficient resources.

Summary

Expert systems would be useful for alleviating scheduling problems in ASD/RW. They could capture corporate knowledge and experiences, reduce unintentional misuse of schedules, reduce resource requirements of schedule development, facilitate schedule changes caused by changes in requirements, and improve the quality and consistency of schedules. An expert system is feasible, suitable, and desirable.

IV. Knowledge Engineering

Overview

This chapter reviews the approach used to acquire the knowledge needed for an Intelligent Scheduling Assistant (ISA). The planned approach was to investigate the use of the Phase I network using a combination of on-site observation, problem discussion, and problem analysis. On-site observation involves watching the expert solve real problems on the job; problem discussion explores the kind of data, knowledge, and procedures needed to solve specific problems; and problem analysis requires the expert to solve realistic problems while explaining his rationale (Waterman, 1987:156-160)

The problem assessment phase revealed that the Phase I network had not been accepted by its intended users. Only two cases were discovered where the Phase I network had been used. Both of these efforts failed. Clearly, experts in tailoring the Phase I network do not exist. Thus, on-site observation and problem analysis could not be used to obtain knowledge about tailoring the Phase I network.

Despite the problems, the approach described for using the Phase I network seemed reasonable. Members of the project team would analyze each task individually to determine its applicability, duration, and relationship to other tasks in the networks. Modifications were made where deemed appropriate. A team approach was necessary because no single individual knew enough about the entire acquisition process to tailor the complete network. Thus, ISA proved to be a multiple expert problem.

Problem discussion became the primary method of knowledge acquisition. Two program managers, six people from RWK, six from RWE, five from other divisions, and two external to RW were interviewed to obtain tailoring knowledge for each task in the Phase I network. Appendices B and C contain the results of the knowledge engineering interviews. The remainder of this chapter describes the conduct of the interviews and summarizes the results.

Knowledge Engineering Interviews

People were selected for knowledge engineering interviews in three ways. First, people interviewed during the problem assessment phase were asked to recommend individuals who were knowledgeable in different functional areas. Second, some knowledge engineering interviews led to referrals to other people who were deemed to be better qualified in some aspect of the problem. Third, functional area supervisors were contacted for assistance with tasks which were not adequately covered by referrals.

Knowledge engineering interviews focused on each task in isolation. Individuals were asked for criteria which could be used to determine task applicability, duration and relationships. First, the individual being interviewed was asked under what circumstances a task would not be applicable. Once applicability criteria were established, the individual was asked how reasonable durations would be estimated. Finally, individuals were asked to comment on the task relationships defined in the Phase I network.

Knowledge Engineering Results

Knowledge engineering interviews re-emphasized the lack of acceptance of the Phase I network in ASD/RW. Most of the individuals interviewed were unaware of its existence. Others knew something about the effort but were unfamiliar with the result.

Knowledge engineering interviews focused on the 48 tasks contained in the Phase I network. One of these tasks was dismissed as inappropriate, three were combined with other tasks, and eight new tasks were identified. Most of the additions were the result of expanding single tasks into two or more tasks. A complete listing of the tasks is contained in Appendix B.

Although some individuals were able to clearly define criteria for making tailoring decisions, most found it difficult. This is a major obstacle to the development of an expert system because the quality of the schedule is dependent on the quality of the knowledge in the system. However, the purpose of this thesis was to demonstrate how knowledge-based techniques could be used to construct ISA. Since the technique for generating the schedule is not dependent on specific program and task knowledge, the quality of the knowledge collected was not critical.

People who were reluctant to offer criteria were told that the accuracy of the knowledge was not critical during initial stages of development. The criteria offered would serve as parameters which could be tuned based on historical data or other knowledge once the system was built. Criteria which were deemed unnecessary could be deleted with minimal impact on system performance. Every effort was made to define criteria in precise terms.

Thirty-five program characteristics were defined for use in tailoring various tasks. Thirteen of these criteria affect the applicability of twenty-three tasks. For example, follow-on programs do not require a New Start Review. The remainder of the criteria (and some of the previous thirteen) affect task durations. A complete listing of program characteristics is contained in Appendix C.

Summary

Problem discussion was the primary method used for acquiring knowledge. Knowledge engineering interviews focused on the applicability, duration, and relationships of individual tasks. Thirty-five parameters were identified for use in generating a tailored schedule.

V. Prototype Design and Tool Selection

Overview

This chapter discusses system requirements, prototype design, and tool selections. System requirements are defined for an operational Intelligent Scheduling Assistant (ISA). The prototype design covers that subset of the system requirements deemed necessary to demonstrate the feasibility of the concept. Tools were selected based on system requirements and the prototype design.

System Requirements

The design goal for the Intelligent Scheduling Assistant is to improve the PM's ability to develop, maintain, and analyze program schedules. A conceptual view of an ISA is provided in Figure 1.

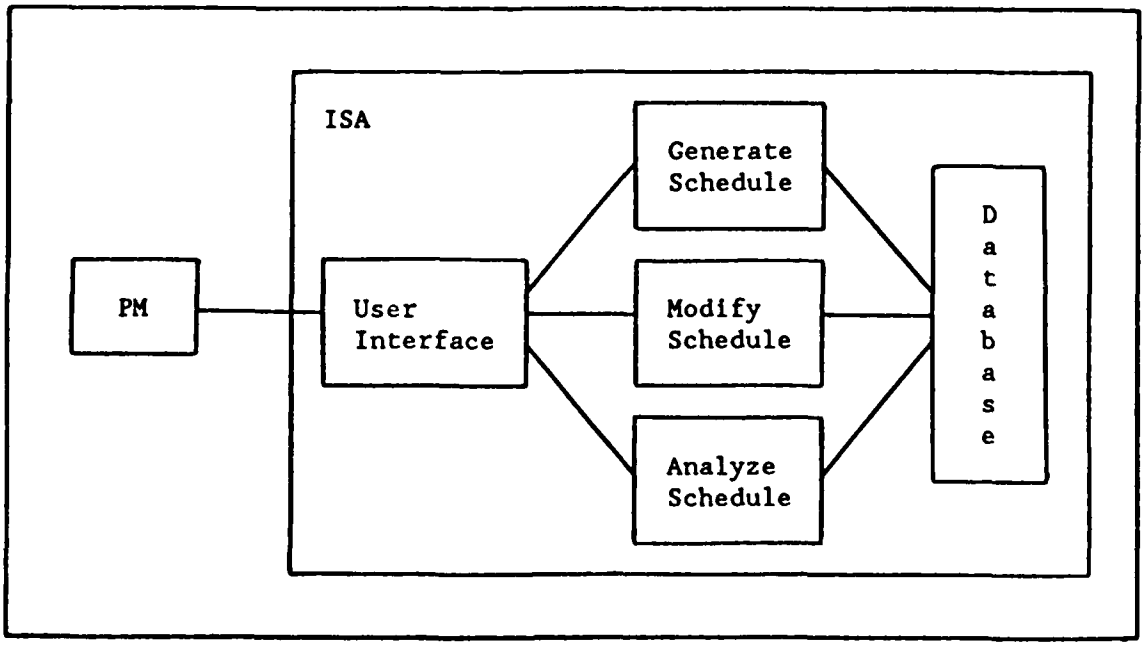


Figure 1. Conceptual View of ISA

The PM is the intended user of the system. The user interface allows the PM to direct schedule development, modification and analysis. Schedule generation builds a schedule by selecting tasks, durations, and relationships. Schedule modification changes tasks, durations, and relationships in an existing schedule. Schedule analysis computes schedule timing and suggests alternatives for resolving schedule conflicts. Separate knowledge bases could be used for schedule generation, schedule modification, and schedule analysis. The database maintains information for defense acquisition programs.

Twelve design principles for interactive computer systems proposed by Woffinden (Woffinden, 1984:20-26) were considered while refining system requirements.

The first design principle, determine the purpose of the system, requires a clear understanding of the problem before designing the user interface. The complexity of defense program management makes it impossible for a single person to understand all program requirements. This complexity is complicated by frequent duty changes and changes in regulatory requirements. The purpose of ISA is to help the PM develop, maintain, and analyze program schedules.

The second design principle, know the user, requires an understanding of the intended users. The typical PM works long hours to meet program demands, depends on functional area personnel to provide task inputs, and has little time to learn new computer systems or wait for computer responses. ISA should be easy to use, provide feedback to the user, and generate results in less than an hour.

The third design principle, identify resources available, involves the choice of hardware and software. A VAX-780 mainframe computer, Z-248 personal computers, expert system building tools, conventional programming languages, scheduling tools, and business software tools are available for prototype development. The choice of hardware and software is detailed in the section on tool selection.

The fourth design principle, consider human factors, looks at the physical and psychological impacts of the system. ISA should minimize typing requirements, provide a menu-driven interface, and use color to enhance its appeal. Menus should be limited to six options to avoid overwhelming the user with choices.

The fifth design principle, determine the interface language, suggests that the system should avoid making the user learn a new and unfamiliar language. The selection of an interface language is addressed in the section on tool selection.

The sixth design principle, consider the environment of operation, concerns comfort and efficiency. Since ISA is designed for use in an office environment, no special requirements are generated by considering this principle.

The seventh design principle, design for evolution, emphasizes providing for planned and unplanned changes to the system. A modular design should be used facilitate both planned and unplanned changes.

The eighth design principle, optimize training, allows novices to use the system without depending on experienced users. ISA should contain sufficient prompts and help facilities to guide a novice through the scheduling process.

The ninth design principle, accommodate levels of experience, suggests different methods of operation should be available to the user. ISA should provide separate modes of operation for novices and experienced users. A novice mode should provide sufficient prompts to allow inexperienced users to develop, modify, and analyze schedules without the assistance of experienced users. An expert mode should allow experienced users to develop, modify, and analyze schedules with minimal prompting and guidance from the system.

The tenth design principle, use selection versus entry, reduces the demand on the user to enter information. ISA should allow the user to select from a list of acceptable responses where possible.

The eleventh design principle, be consistent, refers to consistency in operations throughout the system. Identical operations should be invoked using the same prompts and commands throughout the system.

The twelfth principle, anticipate errors, exploits the computer's ability to remember and inter-relate details. ISA should check for errors and notify the user of invalid inputs.

Prototype Design

The purpose of the prototype is to demonstrate the feasibility of an Intelligent Scheduling Assistant (ISA) by providing a user interface; I/O facilities; and methods for generating and modifying schedules. Actual schedule computation and drawing is provided by interfacing with existing project management software. Table I summarizes the requirements for the demonstration prototype and the complete system.

ISA uses a database to maintain program information. A table of schedule data and a table of program characteristics are associated with

each program. Schedule data consists of tasks (number, duration, description, start date, complete date, etc.) and relationships (links) between tasks. Program characteristics are parameters (dollar values, type of program, etc.) that are used to select tasks and compute task durations. All schedule development and modifications are done using working copies of the program information associated with the schedule.

Table I. System Requirements Matrix

REQUIREMENT	DEMONSTRATION PROTOTYPE	COMPLETE SYSTEM
User interface	Structure	Completed
Build schedules based on program characteristics	Yes	Yes
Build schedules using direct input of task data	No	Yes
Modify schedules based on changes to program characteristics	Yes	Yes
Modify schedules by changing task data	No	Yes
Compute schedule timing	CSNAS	Yes
Recommend alternatives for resolving schedule conflicts	No	Yes

ISA uses menus and explanatory prompts to guide the user through the scheduling process. Menu choices are limited to six options and color is used to enhance system appeal. Default responses (where applicable) are offered and acceptable responses are listed. ISA checks for errors and notifies the user of invalid responses.

ISA provides I/O utilities to load and save program information. The save utility allows the user to save program information and provides an option to export the data to CSNAS. The load utility makes working copies of program information and provides an option to import data from CSNAS.

ISA generates schedules using a rule set and the values of relevant program characteristics. ISA prompts the user for the values of program characteristics needed to generate a schedule. ISA uses rules to select tasks to be included in the program schedule based on the values of the program characteristics. The rules add the task to the schedule table, compute the task duration, and establish task links.

ISA modifies schedules based on changes to the values of program characteristics. The user changes program characteristic values as desired. Then ISA builds a new schedule using the new values and the procedure described above. ISA allows the user to modify existing schedules or create alternate schedules using this approach.

Tool Selection

A Z-248 personal computer was selected for the development environment of the prototype. The operational system could be delivered on either Z-248s or the Automated Management System (AMS) used with the VAX-780 in ASD/RW. Either choice would make the operational system widely available. The AMS system would provide centralized control for the maintenance of ISA. However, the availability of expert system building tools, conventional programming languages, scheduling tools, and business software tools is better for the Z-248s.

Guru was selected as the tool for building the demonstration prototype. Waterman recommends evaluating expert-system building tools in terms of power and sophistication; support facilities; reliability; maintainability; and tool features when choosing a tool to use. (Waterman, 1986:143). Guru's integrated package of knowledge processing capabilities (expert systems, database management, screen management, programming language, and text processing) clearly provide sufficient power and sophistication for development of ISA. Guru's support facilities (menu interface, interactive environment, help facilities, and trace facilities) speed the development and maintenance of ISA. Guru's reliability and maintainability are satisfactory. Guru's features are sufficient for ISA. It provides a forward-chaining inference engine to reason from program characteristics to schedule decisions, a database to maintain information for defense acquisition programs, and a programming language to implement the user interface.

CSNAS was selected to compute schedule timing and create schedule drawings. CSNAS allows data exchange using data files, is the primary scheduling tool used in ASD, and was used for the Phase I network.

Guru and CSNAS are available in PC and VAX versions. Thus, use of these software tools makes the migration of the system to the AMS system possible if desired.

Summary

The design goal for the Intelligent Scheduling Assistant (ISA) is to improve the PM's ability to develop, maintain, and analyze program schedules. The purpose of the prototype is to demonstrate the feasibility of the ISA concept. Table I summarizes the requirements for the

prototype and a complete system. The prototype uses a Z-248 personal computer, Guru, and CSNAS.

VI. Prototype Development

Overview

The prototype was implemented in a top-down fashion using Guru's structured programming, screen management, database management, and expert system facilities. Structured programming tools were used to build the framework of the prototype. Screen management facilities were used to enhance the appearance of prompts for user commands and data input. Database facilities were used to maintain defense acquisition program information. The CSNAS import/export facility was built using a combination of programming tools and database management. Expert system facilities were used to build two knowledge bases - one to prompt the user for the values of program characteristics and one to generate the schedule.

The remainder of this chapter discusses the implementation of the prototype and alternative approaches. The data structures used by the prototype are presented first. Sections are devoted to the user interface, CSNAS interface, input of program characteristic values, schedule development, and schedule modification. The chapter concludes with a brief discussion of alternative approaches.

Data Structures

Guru's database facilities are used to manage information for defense acquisition programs. Figure 2 illustrates the structure of the database. The prototype uses a program table to keep track of acquisition programs. A set of three tables is associated with each program. One table contains the program characteristic values. The other two

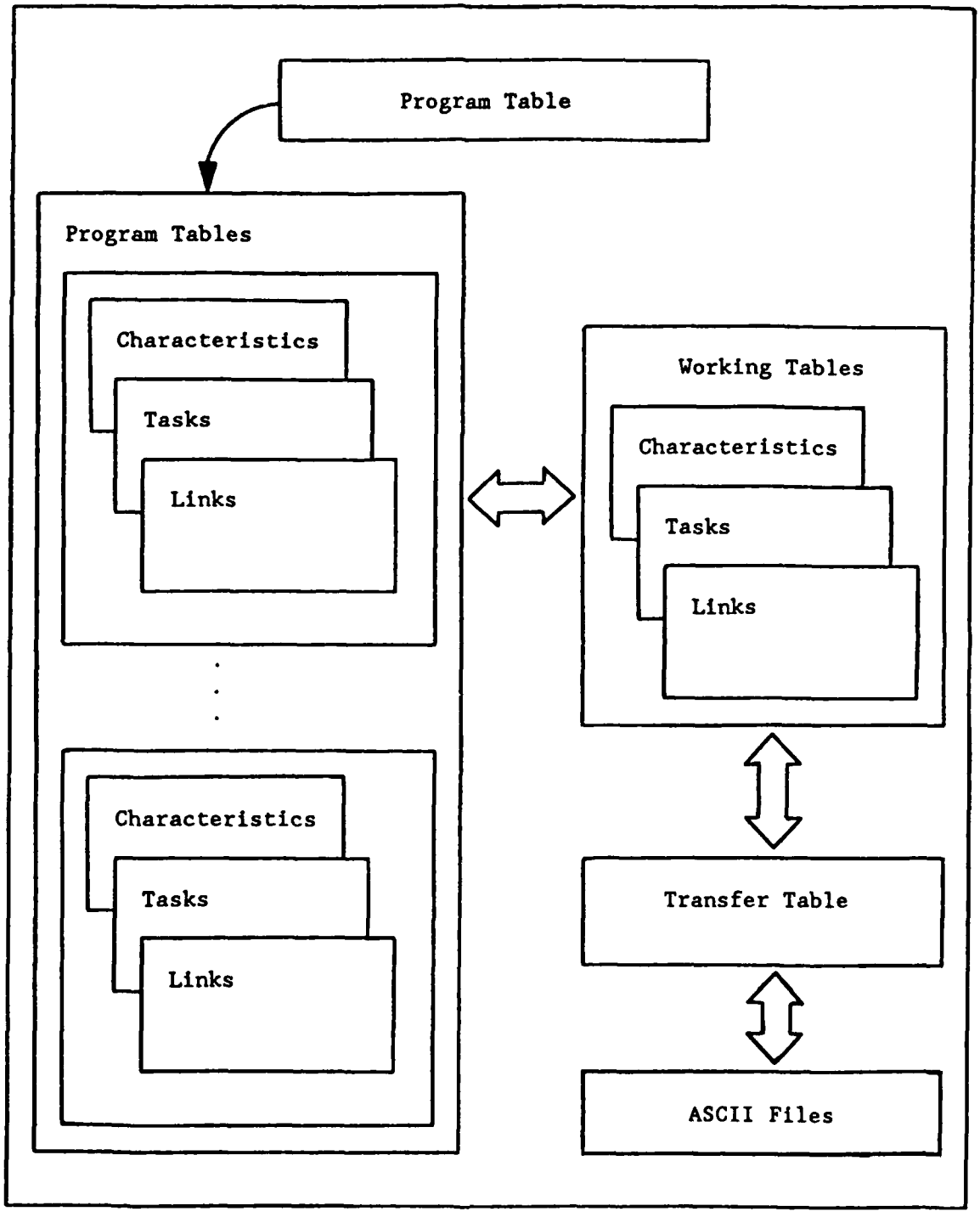


Figure 2. Database Structure

tables (one for tasks and one for links) contain schedule data. A separate set of working tables is used to load copies of program information. All data manipulation is performed using the copies in the working tables. Changes are saved from the working tables back to the appropriate program tables. The transfer table supports transfer of program information to and from ASCII data files. All data is stored in tables as strings.

The program table consists of two fields. The first field (three numeric characters) contains the program ID number. The second field (20 characters) contains the program name. The program ID number is used to identify the set of program information tables associated with the program name. Table II shows a sample program table.

Table II. Program Table

TABLES	NAME
000	Default Schedule
001	Program 001
002	Demo 2
003	Fiber Optics Central Europe

The characteristics table consists of three fields. The first field (30 characters) contains program characteristic descriptions. The second field (20 characters) contains program characteristic values. The third field (15 characters) identifies the source of the value. "DEFAULT" indicates that the user has not confirmed the value associated with the program characteristic; "USER" indicates the user input the value; and "RULE ..." indicates that the value was created by the rule

referenced. The program characteristics table is stored in a disk file. The filename consists of the letter "P" followed by the program ID number followed by the string "CHAR.ITB". For example, default program values are stored in file "P000CHAR.ITB". Table III shows a sample characteristics table.

Table III. Characteristics Table

PROGRAM CHARACTERISTIC	VALUE	SOURCE
RDTE DOLLARS	1	USER
PRODUCTION DOLLARS	1	USER
OTHER SERVICES	NONE	DEFAULT
MCCR INVOLVED	YES	DEFAULT
PMRT PLANNED	NO	USER
TYPE OF PROCESSING	NA	RULE NOPMRT
MAINT/LOG SUPPORT	NA	RULE NOPMRT
ATC TRAINING	NA	RULE NOPMRT
.	.	.
.	.	.
.	.	.
PROGRAM START DATE	900101	USER

The task table contains task information used by CSNAS. The first field (five characters) contains the task number. The second field (five characters) contains the task duration in workdays. The third field (four characters) contains the group code. The fourth field (three characters) contains the percentage completion. The fifth field (24 characters) contains a task description. The sixth field (12 characters) contains the office of primary responsibility (OPR). The seventh and eighth fields (one character each) contain the user input start and user input complete codes respectively. The ninth through sixteenth

fields (six characters each) contain the early start, early complete, late start, late complete, user start, user complete, baseline start and baseline complete dates respectively. Table IV shows a sample task table.

Table IV. Task Table

TASK NUMBER	TASK DURATION	GROUP CODE	PERCENT COMPLETE	TASK DESCRIPTION	OPR
10	0	1	0	DRAFT PMD	PM & PEM
20	20	1	0	ASSESS COST	PM & RWPE
30	23	1	0	FINAL PMD	PM & PEM
40	45	1	0	IPR PREP	PM
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
290	64	1	0	CONTRACT AWARD	RWK

USER INPUT CODE		SCHEDULE DATES							
START	COMPLETE	ES	EC	LS	LC	US	UC	BS	BC
1	1	0	0	0	0	900101	900101	0	0
1	1	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0
.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.
1	1	0	0	0	0	0	0	0	0

ES: Early Start
EC: Early Complete
LS: Late Start
LC: Late Complete
US: User Start
UC: User Complete
BS: Baseline Start
BC: Baseline Complete

Although the prototype does not use all of these fields, including all the fields in the task table facilitates importing data from and exporting data to CSNAS. Furthermore, an operational system may require some (or possibly all) of the fields not currently used. The task table is stored in a disk file. The filename consists of "P", the program ID number, and "TASK.ITB". For example, the tasks for the default schedule are stored in file "P000TASK.ITB".

The link table contains two fields (five characters each) to show the relationships between tasks. The first field contains the number of a task that precedes the task identified in the second field. For example, the fact that task 10 precedes task 20 is established by storing 10 in the first field and 20 in the second field. Table V shows a sample link table.

Table V. Link Table

PREDECESSOR	SUCCESSOR
10	20
20	30
20	40
.	.
.	.
.	.
90	290

A separate table for links is used for efficiency purposes. Although many tasks have only one predecessor and one successor, some tasks may have many predecessors and/or many successors. The link table

is stored in a disk file. The filename consists of "P", the program ID number, and "LINK.ITB". For example, the links for the default schedule are stored in file "P000LINK.ITB".

The working tables are identical in structure to the tables associated with each acquisition program. The files for the working tables are "CURRCHAR.ITB", "CURRTASK.ITB", and "CURRLINK.ITB". The transfer table contains a single field of 70 characters. Details for using the table are described in the section on the CSNAS interface.

User Interface

A menu-driven interface was implemented to guide the user through the process of developing, modifying, and analyzing schedules. Prompts are used to explain system options and request data input. Color is used to improve the appeal of the prototype. All menus contain a maximum of six choices (including exit), and the user selects an option by typing a single letter. Equivalent operations (such as Show Data) in different menus use the same command. The prototype evaluates all inputs and advises the user of invalid selections.

Menus and prompts were built using Guru's screen management facility. This facility allows the user to create forms which consist of boxes of text. Boxes are defined with screen position, background color, foreground color, and special effects such as borders and reverse video. Text can be positioned anywhere on the screen. Inputs can be limited to a specific number and type of characters. For example, input may be limited to a single alphabetic character, four digits, or any combination of ASCII characters.

CSNAS Interface

The prototype interfaces with CSNAS through an ASCII data file. This requires the user to export data to the ASCII file and exit to the operating system. Then CSNAS is started, and the ASCII file is loaded into CSNAS for schedule computations, reports, plot generation, etc. If desired, the user can save any changes made by CSNAS. The prototype can import the changed data for further manipulation such as performing schedule analysis. Although this process is undesirable for an operational system, it is sufficient to demonstrate that the system can interface with other software packages using data files.

The PC version of CSNAS uses a flat data file which stores data by position in the file. For example the first five characters of odd numbered lines contain the task number. The next five characters contain the duration in workdays. A full description of the CSNAS data format is available in the CSNAS User's Guide (AFALC, 1987:247-248).

Guru provides facilities to import and export data from database tables in six different formats (MDBS, 1987:2:45). None of the six formats were sufficient to import and export directly from the table to a CSNAS data file. Thus, a transfer table is used to store and transfer data. Data to be exported is extracted from the appropriate tables and, using string operators, concatenated together to form lines that match the CSNAS format. Each line created is added to the transfer table. The data in the transfer table is exported to a file using Guru's unquoted ASCII format. Data may be imported for further analysis/modification by reversing the process. Data is imported from the CSNAS data file into the transfer table. Then data is extracted from each line of

the transfer table and stored in the appropriate working table. The transfer table is also used to export a copy of the program characteristics to an ASCII data file.

Input of Program Characteristic Values

Rules are used prompt the user for the values of relevant program characteristics. The user is asked to confirm or change values of program characteristics whenever the source of the value is "DEFAULT". The source of the value is changed to "USER" upon user acceptance of the default or input of a new value. Rules are used to change the values of characteristics which are irrelevant based on user inputs for related characteristics. For example, the existence of a security classification guide is marked "NA" with a source of "RULE NOCLASS" when the user states that the program is unclassified.

The actual prompts for program characteristic values are maintained in individual perform files. Perform files are procedural modules which do not require compilation. Although this increases the number of files used, the modularity facilitates maintenance of the prompts. New prompts and changes to existing prompts can be made without affecting other modules. Figure 3 shows an example prompt for a program characteristic. The user can select a response by entering the minimum number of distinguishing characters. The default response for the example below is "INLINE". Pressing <RETURN> accepts this choice, typing "R" selects "RETROFIT", and typing "B" selects "BOTH". This minimizes user typing and corrects for typographical errors as long as the first character is correct.

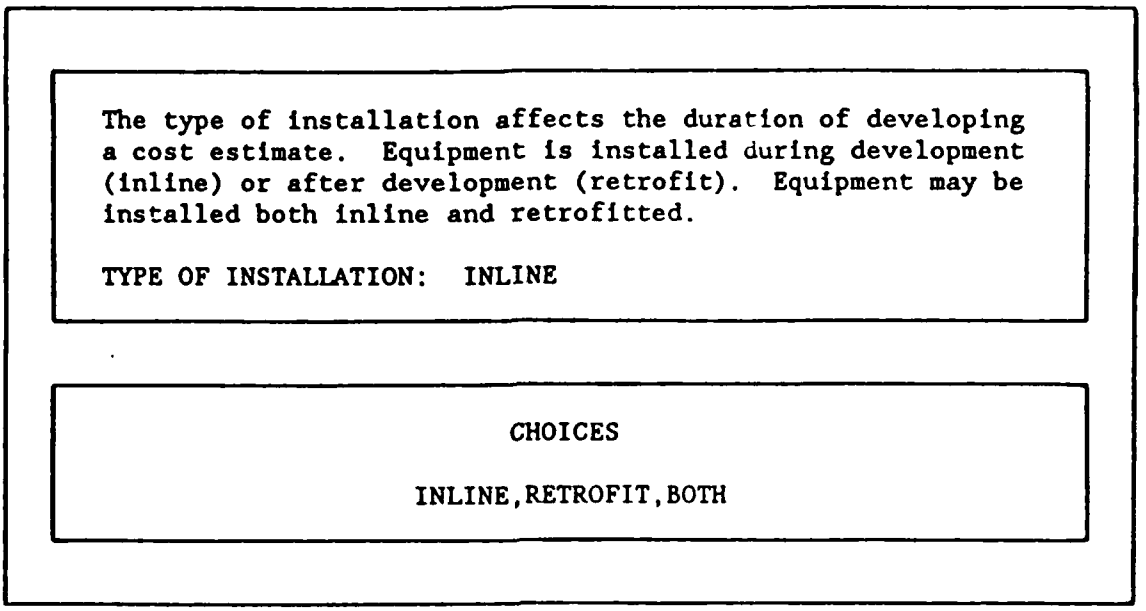


Figure 3. Example Prompt for a Program Characteristic Value

Schedule Generation

The centerpiece of the prototype is a rule set for generating a schedule based upon program characteristic values. The flow of data between the working tables and the rule sets is depicted in Figure 4. Rule set "Get Characteristics" prompts the user for the program characteristic values and stores the responses in the working characteristics table. Rule set "Build Schedule" uses the stored values to determine task applicability, durations, and relationships. Data for applicable tasks is stored in the working task and link tables.

The "Build Schedule" rule set builds the schedule from scratch. It begins with the start task and continues to add relevant tasks until the network is completed. The program characteristic values are used to determine if a task is needed and to calculate the duration of the

tasks. The rules make links between the task being added and all of its immediate predecessors (tasks which must be completed before the new task begins). Appendix D contains a listing of the rules used for schedule generation.

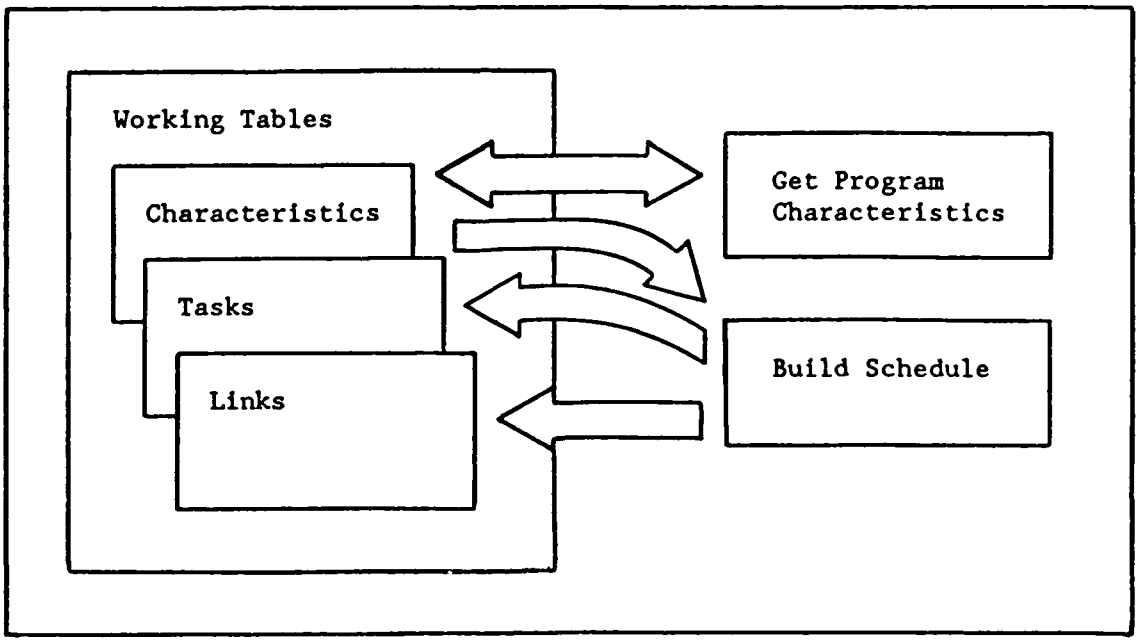


Figure 4. Data Flow Between Working Tables and Rule Sets

The prototype contains rules for 52 tasks. Up to 23 of these tasks may not apply to (be included in) a schedule for a specific program. Some tasks are mutually exclusive. For example, Justification and Approval Activities and Source Selection activities will never occur in the same network. Many tasks are included or omitted as a set. For example, five tasks are related to Source Selection. These complex inter-relationships make it difficult to determine the exact number of potential schedules which could be generated. A conservative estimate is that the prototype could generate up to 8000 different networks

considering only the presence or absence of tasks. More than 10,000 different schedules could be generated when taking into account the permutations created by different task durations.

A perform module (Guru procedural code) is used by the rules to create, initialize, and add tasks to the task data table. The rule then modifies the task duration, description, and office of primary responsibility (OPR). Task numbers start with 10 increment by 10. Thus, the first task invoked is given a task number of 10, the second task is given a task number of 20, and so on.

The prototype creates the start task and works forward to the finish task. New tasks are added when all of their predecessors have been created. This process continues until a complete schedule is generated. This mimics one way that a person might build a schedule. One advantage of this approach is that the first task is always 10, the last task is always the largest task number, and the magnitude of the task number gives an indication of where the task occurs within the topology of the network.

The key parts of a rule are the premise clause (IF statements) and the action clause (THEN statements). When the conditions in the premise clause are true, Guru executes the statements in the action clause. The premise of rules for tasks which will always occur are written so that the rule will always fire (execute THEN statements) at the appropriate time during schedule development. The premise of rules for tasks which may not occur are written so that the rule will fire only when the task applies to (should be included in) the schedule being generated. This process is best illustrated by considering a few rules in the prototype.

Figure 5 shows the rule for the start task. "ASKDONE" is a variable which is always set to true before consulting (invoking) the rule set to generate a schedule. Thus, the start task will always occur. Since all other tasks depend on the start task or one of its successors, the start task will always be the first task created. "PERFORM ADDTASK" calls the perform module to create a new task in the current task table (CURRTASK)

```
IF:     ASKDONE
THEN:   PERFORM ADDTASK
        CURRTASK.DUR = "    0"
        CURRTASK.DESCR = "DRAFT PMD                    "
        CURRTASK.OPR = "PM & PEM    "
        CURRTASK.US = vstart
        CURRTASK.UC = vstart
        tdpmd = CURRTASK.ID
```

Figure 5. Rule for "DRAFT PMD"

The next three lines change the duration, description, and OPR of the task respectively. The next two lines change the user input start date and user input complete date to the program start date (vstart) input by the user. Variable "tdpmd" in the last line is used to hold the task number for task "DRAFT PMD". This assignment also tells the other rules that a task has been created for "DRAFT PMD".

Figure 6 shows the rule for a task which may not apply to a schedule for a specific acquisition program. Variable "vthreat" contains the value for the program characteristic "SYSTEM THREAT". If a threat does not exist, this rule will never execute, and the "THREAT INPUT" task

will not be included in the network. If a threat does exist, the rule will not fire until variable "tfpmd" is KNOWN (given a value). Variable "tfpmd" is used to store the task number for the task "FINAL PMD". Thus, this rule fires after the "FINAL PMD" task has been created only if a system threat exists. If the premise is true, a new task is created and given the next task number. The duration, description, and OPR are changed as shown, and variable "tthreat" is assigned the task number for "THREAT INPUT".

```
IF:     vthreat = "YES" and KNOWN(tfpmd)
THEN:   PERFORM ADDTASK
        CURRTASK.DUR = "   23"
        CURRTASK.DESCR = "THREAT INPUT            "
        CURRTASK.OPR = "PM & FTD    "
        tthreat = CURRTASK.ID
        ATTACH 1 TO CURRLINK
        CURRLINK.PRED = tfpmd
        CURRLINK.SUCC = tthreat
```

Figure 6. Rule for "THREAT INPUT"

The final three lines create the link between "THREAT INPUT" and "FINAL PMD" by adding a new record to the link table (CURRLINK) and changing the values of the predecessor and successor fields. Note that this approach only makes links to predecessors when the rule is executed. The rule only contains knowledge about what the task depends on. It contains no knowledge about what follows the task.

Figure 7 shows the rule for a task which always applies but depends on a task which may not occur. This rule executes when both an "IPR" task has been created, and either a "THREAT INPUT" task has been created

or the system is not threatened. An "IPR" task will always occur in the schedule. The second part of the premise will always be true because "THREAT INPUT" will occur unless the system is not threatened (vthreat <> "YES"). Thus, the premise will always be true at some point during schedule development. When the premise is true, a new task is created and added CURRTASK.

```
IF:     KNOWN(tipr) and
            (KNOWN(tthreat) or vthreat <> "YES")
THEN:   PERFORM ADDTASK
        duration = 140
        IF vscomp = "MAJOR" THEN
          duration = duration + 20
        ENDIF
        IF vjoint = "NONE" THEN
          duration = duration - 30
        ENDIF
        CURRTASK.DUR = TOSTR(duration,5,0)
        CURRTASK.DESCR = "DRAFT SPEC              "
        CURRTASK.OPR = "PM & RWE     "
        tdspec = CURRTASK.ID
        ATTACH 1 TO CURRLINK
        CURRLINK.PRED = tipr
        CURRLINK.SUCC = tdspec
        IF KNOWN(tthreat) THEN
          ATTACH 1 TO CURRLINK
          CURRLINK.PRED = tthreat
          CURRLINK.SUCC = tdspec
        ENDIF
```

Figure 7. Rule for "DRAFT SPEC"

This rule also demonstrates the use of program characteristic values to calculate task duration in the action clause. The default duration is 140 days. If the system complexity is major (vscomp = "MAJOR"), then the duration is increased by 20 days. If the program

does not involve other services (vjoint = "NONE"), then the duration is reduced by 30 days. "TOSTR(duration,5,0)" converts the duration from a number to a string of length five with zero decimals. The duration, description, and OPR are changed as shown. Variable "tdspec" is given its appropriate value, and a link is made between "IPR" and "DRAFT SPEC". Finally, if "THREAT INPUT" does exist, a link will be made between "THREAT INPUT" and "DRAFT SPEC".

The premise of rules which depend on one or more tasks which may not exist can become fairly complex. Figure 8 shows the most complex rule in the prototype. The first clause of the premise tests whether "DRAFT CRISP" has occurred or Mission Critical Computer Resources are not used or Program Management Responsibility Transfer will not occur. The second clause tests if "AP APPROVAL" (acquisition plan approval) has occurred or program work is in the scope of a current contract. The third clause tests if "DRAFT RFP" has occurred, or both a draft RFP will not be used and "SOURCES SOUGHT SYNOP" has occurred, or both a sources sought synopsis is not needed and "DRAFT SOW" has occurred. The final clause of the premise tests if "WBS APPROVAL" (approval of the work breakdown structure) has occurred or the dollar value of the contract is low enough that WBS approval is not required.

Each condition within a clause is mutually exclusive (if one part of the clause is false, then the other is true). Thus, this rule will always fire after those tasks which will occur have been created. When the rule fires, a new task is created and added to CURRTASK. The duration, description and OPR are changed. The task number is assigned to "tfsow". The remainder of the action clause creates links to tasks

```
IF:     (KNOWN(tdcrisp) or vmccr <> "YES" or vpmrt <> "YES")
        and (KNOWN(tapa) or vinscope = "YES") and
        KNOWN(tdilsp) and
        (KNOWN(tdrfp) or (vdrfpn <> "YES" and
          (KNOWN(tsss) or (vsssn <> "YES"and
            KNOWN(tdsow))))) and
        (KNOWN(twbsa) or (vrdte <= 2 and vprod <=2))
THEN:   PERFORM ADDTASK
        CURRTASK.DUR = "   40"
        CURRTASK.DESCR = "FINAL SOW                "
        CURRTASK.OPR = "PM & RWE    "
        tfsow = CURRTASK.ID
        ATTACH 1 TO CURRLINK
        CURRLINK.PRED = tdilsp
        CURRLINK.SUCC = tfsow
        IF KNOWN(tdcrisp) THEN
          ATTACH 1 TO CURRLINK
          CURRLINK.PRED = tdcrisp
          CURRLINK.SUCC = tfsow
        ENDIF
        IF KNOWN(tapa) THEN
          ATTACH 1 TO CURRLINK
          CURRLINK.PRED = tapa
          CURRLINK.SUCC = tfsow
        ENDIF
        IF KNOWN(tdrfp) THEN
          ATTACH 1 TO CURRLINK
          CURRLINK.PRED = tdrfp
          CURRLINK.SUCC = tfsow
        ELSE
          IF KNOWN(tsss) THEN
            ATTACH 1 TO CURRLINK
            CURRLINK.PRED = tsss
            CURRLINK.SUCC = tfsow
          ELSE
            ATTACH 1 TO CURRLINK
            CURRLINK.PRED = tdsow
            CURRLINK.SUCC = tfsow
          ENDIF
        ENDIF
        IF KNOWN(twbsa) THEN
          ATTACH 1 TO CURRLINK
          CURRLINK.PRED = twbsa
          CURRLINK.SUCC = tfsow
        ENDIF
```

Figure 8. Rule for "FINAL SOW"

which are known to exist. Connections are made to "DRAFT CRISP", "AP _ APPROVAL", "DRAFT RFP" and "WBS APPROVAL" if they exist. If "DRAFT RFP" does not exist, a link is made to "SOURCES SOUGHT SYNOP" if it exists. If both "DRAFT RFP" and "SOURCES SOUGHT SYNOP" do not exist, a link is made to "DRAFT SOW". The third clause is so complex because "DRAFT SOW" always occurs and is a predecessor to a task which does not normally occur. The third clause is necessary to insure that a link will always be made to "DRAFT SOW" even if both "SOURCES SOUGHT SYNOP" and "DRAFT RFP" do not occur.

Each task in the network is handled by a single rule. The premise of the rule is written to create the task at the appropriate point in schedule development. Tasks which always exist have premises which are always true at the appropriate point in schedule development. Tasks which may not occur have premises that are true only when program characteristic values indicate that the task should be included in the network. The actions of the rule create the task, calculate task duration using program characteristic values, and make links to all immediate predecessors.

This process makes each rule relatively independent of the other rules. The only dependencies are those created by task links. Thus, all task data (duration, description, OPR, schedule dates, etc.) can be changed within the action clause of a rule without affecting other rules. When adding or deleting rules, the developer/maintainer need only concern himself with the task involved and its immediate predecessors and successors. If a task is deleted (the rule for the task removed from the rule set), the developer/maintainer must insure that its immed-

iate predecessors still have successors and its immediate successors still have predecessors. When adding tasks (creating a new rule), the developer/maintainer must create the premise so that the task will be created at the appropriate point in the network. The action must create the task, calculate task duration, and make links to preceding tasks. Succeeding tasks must be modified to insure that proper links are made to the new task.

The rule-base for generating the schedule was built using the process just described. Rules were written based on information gathered during the knowledge acquisition phase. The Phase I network was used only to determine links and durations for tasks which were not covered in the knowledge engineering phase. The topology of the Phase I network was not used as a guide for rule development. Indeed, the actual topology of the generated schedules were unknown until CSNAS plots were generated.

Based on this thesis, a rule set to generate a network schedule containing fifty tasks can be built in one or two months. This assumes that the developer works full-time on the project and has full cooperation from (access to) the individuals who will provide the knowledge for the system. The rule set will be a prototype which must be refined and maintained. The time to reach maturity for the rule set is unknown.

Schedule Modification

A rudimentary facility for modifying an existing schedule by changing program characteristic values was implemented. The user can create a new schedule (or simply modify the existing one) by changing program

characteristic values. This lets the user generate multiple versions of a program schedule by changing key program characteristic values. Thus, alternative approaches can be evaluated by generating a schedule for each alternative. For example, schedules might be generated for both sole source and competitive acquisitions. The two schedules could then be compared to see which is better given the risk involved and other appropriate factors.

If the user wants a new schedule, the system makes a copy of the schedule to be modified. Otherwise, the system modifies the existing schedule when the user saves the changes. Guru's browse facility is used to allow the user to review all program characteristic values and make changes as desired. When completed, the user can save the changes or ask the system to generate a new schedule based on the new program characteristic values. A more elaborate interface similar to that used to input program characteristic values would be developed for the operational system.

Alternative Approaches

The approach described for generating schedules is one of four considered. The prototype generates schedules forward from the start task as described above. Schedules could also be generated working backward from the finish task, creating tasks independently, or using frames.

The same approach used to work forward from the start task could be used to work backward from the finish task. The finish task would be created first. Then a task would be created only if it applied and the task which follows it had already been completed. For example, "CON-

TRACT AWARD" is the finish task and depends on "SOURCE SELECTION". Thus, "CONTRACT AWARD" would be created first. "SOURCE SELECTION" would be created after "CONTRACT AWARD", and a link made between the two. Note that this case requires the rule to know what tasks are dependent on the task created. The rule would contain no knowledge about what tasks precede the new task. The numbering would be backwards, but numbering is irrelevant to schedule computation and could be easily changed if desired. This approach merely takes a different viewpoint from that used for the demonstration prototype.

A schedule could be built by generating tasks independently. Tasks which always occur would be created automatically. Tasks which depend on program characteristic values would be created (if they apply) regardless of what other tasks exist. The rules for these tasks would not contain any link information. Once all the tasks were created, a separate rule base could create links between the tasks that exist. This approach would separate link information entirely from task creation. Thus, maintenance, addition, and deletion of rules could become more difficult than using the modular approach described.

A frame-based system could be used to generate schedules. Each task could be represented as an instance of a generic task frame. The frame could contain slots for all of the data maintained in the tables of the rule-based system. Methods could be written to determine if the task applied, create the task, calculate the task duration, and make links to other tasks. A frame-based system would further enhance the modularity of the tasks. However, PC-based frame tools were not readily available, and rules are easier to encode.

VII. Prototype Evaluation

Overview

The prototype accepts more than six trillion possible combinations of program characteristic values and can generate thousands of schedules. Testing of all possible inputs is impossible. Thus, the system was evaluated in two ways. First, tests were run to increase the confidence that the system would generate a schedule regardless of the combination of program characteristic values. Then the prototype was demonstrated to potential developers and users of an operational system. Each person who witnessed a demonstration was asked to evaluate the prototype by answering a questionnaire.

Prototype Testing

The system was implemented using modular components. Each component/collection of components was tested as it was built. The user interface was tested (using stubs for each option) by selecting every option available. The CSNAS interface was tested by exporting and importing data files. Exported data files were loaded into CSNAS and tested for compatibility.

The rule set for input of program characteristic values was tested by using it to prompt for program characteristic values and reviewing the stored data to see that it matched the inputs. Testing the rule set for schedule generation proved to be more difficult. Thirteen program characteristics (containing 49 choices) directly affect the existence of 23 tasks. The remaining program characteristics affect the duration

of tasks. Initial efforts were spent on generating a schedule based on default program characteristic values. Then, program characteristic values which affected task applicability were changed and new schedules generated. Finally, random input of program characteristic values was used.

Approximately twenty schedules were generated using different program characteristic values. When the prototype failed to generate a complete schedule, Guru's trace facility was used to locate the problem. Proper schedules were not generated for two reasons. Some tasks were incorrectly omitted due to an incorrect premise statement for the rule. These errors were corrected by modifying the premise. Some tasks were missing a predecessor and/or a successor. These errors were corrected by modifying the actions of appropriate rules to insure that links were established. Most problems were located and fixed within an hour.

The rule set for schedule generation was also evaluated by manually reviewing each rule. Rules for tasks that always occur were reviewed to insure that their premises were always true at the proper point in schedule development. Rules for tasks that may not occur were reviewed to insure that they would execute only when the task should be included in the network. Finally, cross-checks were made between rules to insure that proper links would be made to create a complete network regardless of which tasks were used.

Approximately ten more networks were generated after evaluating the rules. Most of these networks were generated using program characteristic values provided by potential users during demonstrations of the system. Schedule length for all test runs and demonstrations ranged

from 18 months to 34 months. Schedules were generated by the prototype in less than 15 minutes.

User Evaluation

The prototype was demonstrated to 18 people. Schedules were generated, modified, and exported to CSNAS to demonstrate system operation. Plots of five schedules generated by the system were shown to the evaluators following system operation. Everyone was asked to complete an evaluation questionnaire (see Appendix E) after the demonstration was completed. Figure 9 summarizes the responses to the first four questions.

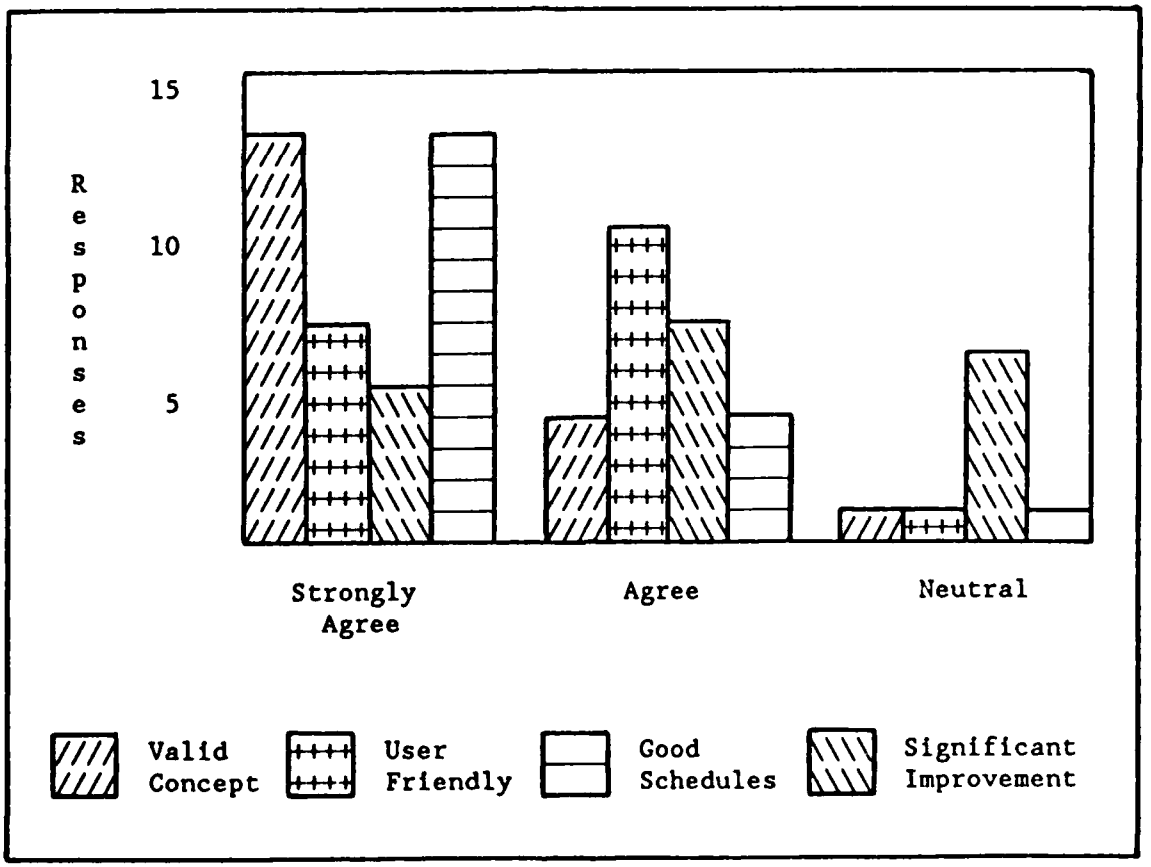


Figure 9. Summary of Responses to Primary Criteria

Evaluations were completed by people from the Aeronautical Systems Division (ASD), the Air Force Acquisitions Logistics Command (AFALC), and The Applied Science Corporation, Incorporated (TASC). TASC provides scheduling support to ASD. Sixteen of the eighteen evaluators (88 percent) had used network scheduling more than three times, had used automated project management tools (primarily CSNAS), and had used model networks. All eighteen (100 percent) believe that network scheduling has merit. The prototype received no negative comments. Table VI cross-tabulates questions 1 and 4.

Table VI. Cross-tabulation of Concept and Improvement

		Improvement is Significant		
		Strongly Agree	Agree	Neutral
Concept Is Valid	Strongly Agree	61%	11%	0%
	Agree	5%	11%	5%
	Neutral	5%	0%	0%

Summary

The prototype accepts more than six trillion possible combinations of program characteristic values and can generate thousands of schedules. The prototype was tested as each component/group of components was implemented. It generates tailored schedules in less than 15 min-

utes. Evaluations by eighteen people with experience in scheduling and the use of model networks contained no negative comments.

VIII. Conclusions and Recommendations

Overview

The prototype successfully demonstrates the feasibility of building an Intelligent Scheduling Assistant (ISA). The prototype was demonstrated to people from the Aeronautical Systems Division (ASD), the Air Force Acquisitions Logistics Command (AFALC), and The Applied Science Corporation, Incorporated (TASC). The demonstrations led to presentation of the concept at the ASD/Industry Scheduling Workshop. AFALC plans to use the concept for CSNAS acquisitions logistics model networks (Appendix F).

Conclusions

The prototype successfully demonstrates the feasibility of building ISA. User evaluations (Figure 9, Chapter VII) were consistently favorable. Ninety-five percent of the evaluators recommended development of an operational system or further research. Although the prototype's user interface was limited to menus and a few verbose prompts, ninety-five percent of the evaluators rated the potential for user-friendly operation as good to excellent. Sixty-six percent rated the quality of schedules generated by the prototype as reasonable. Ninety-five percent rated the prototype as an improvement to model networks.

The lack of a consensus on the quality of schedules generated by the prototype is derived from two factors. First, most of the people who evaluated the system did not examine the schedules closely. Second, schedule quality is difficult to quantify. However, schedules generated

are meant to be recommendations which must be reviewed and approved by members of the program management team. Therefore, the absence of negative evaluations is significant considering the limited knowledge of the prototype and the intended use of the results.

Cross-tabulation of concept and improvement (Table VI, Chapter VII) shows strong support for both. More than sixty percent of the evaluators recommended development of an operational system and agreed that the prototype was much better than existing model networks.

The concept is translatable to other network schedules using the procedures outlined in this thesis. The first step is to identify all tasks which may be included in a given schedule. Each task is analyzed to determine when it applies, how durations are estimated, and what relationships exist. A single rule can be created to handle each task. The premise of the rule is written so that the task is created at the appropriate time during schedule development. The actions of the rule compute task durations and establish links to immediate predecessors This process can be used to generate schedules without prior knowledge of the final network topology. Furthermore, only the predecessors of tasks need to be defined.

Schedule generators would be easy to build for the construction industry where tasks, durations, and relationships are well-defined The quality of the generated schedules would probably be very high under such circumstances. Schedule generators can be built for other model networks and where no previous work has been done to generate schedules The time required to build the initial prototype would range from one to two months for a network with fifty tasks.

Interfaces can be written to export schedule data to other software programs. The same technique used to create the CSNAS data file can be used to export data in an appropriate format for database management systems, other project management tools, or other software program which allow data interchange.

Schedules generated using this approach are only as good as the knowledge contained in the system. All generated schedules should remain subject to review by members of the program management team. The knowledge in the system must be continually reviewed and refined. The time to reach maturity is unknown.

The prototype has generated a lot of interest in the ISA concept. It was demonstrated to AFALC and the AI group of the Air Force Logistics Command (AFLC). Based on the demonstration, AFALC plans to apply the concept to CSNAS model networks for acquisition logistics. AFLC is supporting the project with training and M.1 (a knowledge-engineering language developed by Teknowledge). The concept was presented at the ASD/Industry Scheduling Workshop held 16-19 February 1988 in Los Angeles, California and will be presented to TASC in March 1988.

Recommendations for Future Research

The prototype was designed to demonstrate the feasibility of generating schedules using a knowledge based approach. However, schedule generation is only one part of the scheduling problem. A significant amount of effort is needed to maintain schedules. Thus, methods for modifying schedules by adding, deleting, or modifying tasks are essential. Analysis of schedules is another key element to successful use of network schedules. Thus, research into resolving schedule conflicts

using knowledge-based techniques is needed. Adding lines of reasoning to the current system would be valuable to people who review schedules generated by the prototype. Finally, automated procedures for generating rules for ISA would be beneficial.

Modifying Tasks.

Users will quickly become disenchanted with any computer program which does not allow them to easily modify schedules which are generated. Methods for adding, deleting, or modifying tasks in the schedule are essential to successful implementation of an operational ISA. Although such changes can be made using CSNAS or other project management tools, this is not an acceptable solution in the long term. The remainder of this section describes the kind of interface needed to facilitate schedule modification.

The user should be allowed to add any task which is deemed relevant to his program. These may be tasks that should ultimately be added to the rule base for schedule generation, or they may be tasks that are peculiar to the program in question. ISA should prompt the user for all information needed to add the task without requiring the user to concern himself with the details of making the changes. For example, ISA should automatically assign a task number and prompt the user for a description, duration, and office of primary responsibility (OPR). ISA should ask the user which tasks must occur before the task can begin and which tasks cannot be done before the task ends. ISA should handle the mechanics of creating the proper links. The user should also be allowed to input group codes and schedule dates as desired.

The user should be allowed to delete tasks which are deemed irrelevant to the program. The rule base for schedule generation may require modification to account for the cause of the deletion, or the deletion might be peculiar to the program. ISA should analyze links to previous tasks and advise the user on how tasks should be reordered. For example, if "NSR" is deleted, ISA should recommend that "IPR PREP" should follow "ASSESS COSTS". ISA would make the link only if the user confirms the proposed modification. Otherwise, ISA should prompt the user for desired links until the network is complete.

The user should be able to modify task data such as description, duration, and schedule dates. The user should also be able to restructure the network. The system should prompt the user for the task to be modified or moved. When moving tasks, the system should recommend appropriate changes in links to the user. For example, if the user wanted to show a task was complete although the predecessors to the task were not complete, the system should help the user decide how to restructure the network. This would insulate the user from the details for moving tasks.

Analyzing Schedules.

System requirements include computation of schedule timing and resolution of conflicts. Schedule timing is currently computed using CSNAS. The ability to compute schedule timing without exiting the system is a needed improvement. An ability to recommend alternatives to resolve schedule conflicts would dramatically increase the usefulness of the system. The remainder of this section describes possible approaches to providing these requirements.

Schedule timing could be computed within the system by creating a direct interface to CSNAS or developing internal algorithms. The first approach would require software which executes a sequence of keystrokes to load and run CSNAS. A file containing the keystrokes could be generated and invoked by the system. This approach would avoid re-inventing the wheel for schedule computation. The second approach would use programming languages, databases, or spreadsheets to compute schedule timing.

Alternatives for resolving schedule conflicts could be generated using a rule set. Consider the case where the schedule fails to meet its target date for completion. The system could examine the network to determine which tasks lie on the critical path. Rules could be used to generate recommendations for reducing the schedule. The recommendation would be accompanied by an explanation which included an assessment of the risk to the program if the change is made. The change would be made only upon user confirmation.

Providing Reasoning.

A record of the lines of reasoning used to generate a schedule would be useful to people who review the schedule. The reviewer could refer to the line of reasoning whenever he does not understand why a task was included (or omitted), how a duration was estimated, or how relationships were established. If the reviewer disagreed with the line of reasoning, then he would contact the person responsible for the schedule to recommend changes.

Lines of reasoning could be maintained in a database. Each rule could be modified to add text descriptions of why the task applies and

how the duration was calculated to the database. Text descriptions of why tasks are not in the network could also be added to the database. The lines of reasoning could then be exported along with the CSNAS data file and program characteristic values.

Automating Rule Generation.

The rules contained in the Intelligent Scheduling Assistant are fairly simple to construct. Automating rule generation would facilitate the development of rule sets for other model networks. The system could prompt the user for all possible tasks and knowledge needed to decide the applicability, duration, and links for each task. The system could use this knowledge to generate rules for schedule development.

Summary

This thesis has successfully demonstrated the feasibility of building an Intelligent Scheduling Assistant. It has generated interest in using knowledge-based techniques to improve scheduling efforts in ASD and AFALC. The concepts developed in this thesis are directly translatable to other network schedules. This thesis has spawned ideas for for further research into using artificial intelligence to improve schedule development and maintenance.

Appendix A: Problem Assessment Interviews

This appendix contains the results of the problem assessment interviews. Interviews were conducted informally using a problem assessment questionnaire developed by Teknowledge, Inc. (an international knowledge engineering company). Each numbered item represents an area of focus. Under each area is an approximation of the question asked. Suggested responses (if given) are contained in parentheses. Actual responses are preceded by an "*", and the number of individuals offering the response as listed at the right margin.

Problem assessment interviews were directed toward the Phase I network. People who had not used the network were asked to consider how they would use the network. Some people offered multiple responses for specific questions and some declined to respond to specific questions. All references to "program" mean defense acquisition program.

1. Problem Definition

What are the problems associated with model networks/network scheduling?

*	Time-consuming	6
*	Lack of acceptance	5
*	Phase I network requires drastic tailoring	5
*	Every program is unique	4
*	Lack of resources	4
*	Managing schedule instead of program	3
*	No one person knows everything	3
*	Lack of priority/motivation	2
*	Political decisions	2
*	Scheduling guide lacks sufficient information	2
*	Lack of experience	2
*	Too many regulations	1
*	PM doesn't know organizational commitment	1
*	Changes occur	1

How do scheduling problems impact the organization?

*	Inefficient management	6
*	Time-consuming to coordinate	3
*	Time-consuming to research regulations	3
*	Workload-intensive	2
*	Minimal impact	1

What benefits would a solution to the problems have (save money, save time, improve productivity)?

*	Improve productivity	9
*	Save time	4
*	Improve logical consistency of schedules	2
*	Improve management	2
*	Help define needs	1
*	Weed out non-players	1
*	Lessons learned	1

2. Current Practice

What are the current procedures used to tailor the Phase I network or develop schedules?

*	Review/identify task applicability	13
*	Coordinate with program management team	12
*	Review/develop estimated task durations	12
*	Get copy of network	9
*	Review/develop network logic	9
*	Modify/maintain schedule	6
*	Review/develop OPR	4
*	Learn about program characteristics	3
*	Program already in progress	2
*	Select most important tasks	2
*	Schedule in tiers	2
*	Determine fixed dates	1
*	Resolve conflicts	1
*	See if goal fits network	1
*	Tend to add activities more than delete activities	1
*	Hire contractor	1
*	Reviewed by corporate review group	1

What expertise is needed to tailor/generate schedules?

*	Functional area knowledge	15
*	Program experience	3
*	Scheduling	1

Where is the expertise (people, regulations)?

*	People (functional, PM)	15
*	Regulations, policy letters, operating instructions	11

How long does it take a skilled scheduler to develop a top-level schedule?

*	One week	4
*	Hours (days to coordinate)	2
*	Two to three weeks	2
*	One month	2
*	Two days	1

What is the difference in performance between skilled schedulers and average schedulers?

*	Skilled is better by magnitude in time and quality	14

How do failures occur?

*	Omissions	6
*	Lack of experience	5
*	Unrealistic expectations	4
*	Poor decisions/assumptions	3
*	Misconceptions	3
*	Lack of coordination	3
*	Quit due to frustration	2
*	Radical differences between Phase I network and program	1
*	Lack of support	1
*	Disagreements	1
*	Failure to schedule at all	1

How often do failures occur?

*	All the time	5
*	Occasionally	2

What are the costs associated with failure?

*	Increased costs, schedule delays	8
*	Major impact	6
*	Kill program	2
*	Can't assess program status	1

How should a knowledge-based system generate schedules?

*	Same approach as currently used	8
*	Based on real data	1
*	Larger view of program	1

How does information flow in the organization (any bottlenecks)?

*	Good	6
*	Bottlenecks	3
*	Problems getting data at start	1
*	Poor	1

3. Problem Characterization

What is the goal of the solution (generate satisfactory, optimal, or multiple schedules)?

*	Generate satisfactory schedules	10
*	Generate optimal schedules	5
*	Generate multiple schedules	3
*	No satisfactory schedules	1

Are schedules selected from pre-defined solutions or constructed without knowledge of the final solution?

*	Select from pre-defined solutions	11
*	Construct	3
*	Durations pre-defined	1
*	Precedence unknown	1

How many potential solutions are there (tens, hundreds, millions)?

*	Hundreds	8
*	Millions	5
*	Tens	2
*	Good and bad	1

Does the expert use a subset of all possible solutions?

*	Small subset	15
*	Iterative	1

How much input data is needed (tens, hundreds, thousands)?

*	Hundreds	9
*	Thousands	3
*	Tens	2

What kind of reasoning is required (certainty, uncertainty, search)?

*	Uncertainty	15
*	Some certainty	4

What kinds of reasoning strategies are used?

*	Gather program characteristics first	13
*	Generate schedules by analogy	2
*	Use experience	4
*	Layout roadmap	1
*	Iterative process	1
*	Coordinate	1

4. Characterization of a Successful Knowledge-based System (KS)

What features are needed to make a KS successful (graphics, voice I/O, touch-screen panel, integration with current software, video-disc display, very fast response, zero incorrect solutions)?

*	User-friendly	7
*	Not too slow	7
*	Graphics	6
*	Easy to understand	1
*	On-line documentation	1
*	Integrate with briefing tools	1
*	Present meaningful information	1
*	Intelligent interaction	1
*	Correlate to database	1
*	Highlight problems	1
*	Capture lessons learned	1
*	Touch-screen panel	1
*	Videodisc displays	1
*	Explanations	1

What are the potential benefits of a KS (conserve expertise, improve productivity, training, reduce cost)?

*	Improve productivity	12
*	Conserve expertise	14
*	Training use	9
*	Reduce cost	6
*	Evaluate impacts	1
*	Lessons learned	1

Are the risks/costs acceptable?

*	Acceptable	4
*	Limited investment	1

What should the design assume/include to avoid pitfalls?

*	Assume inexperienced user	2
*	Assume computer illiteracy	2
*	Adapt to background of user	1
*	Provide for maintainability	1

*	Provide glossary/help	1
*	Avoid technical jargon	1
*	Allow user to override	1
*	Must be relevant	1
*	Provide immediate feedback	1

Would a KS be accepted by organizational personnel?

*	With some initial reluctance	6

5. Defining the Prototype

What should the demonstration prototype do?

*	Solve specific problem	9
*	Solve generic problem	7
*	Provide explanation	1
*	Determine durations	1

What kind of knowledge does the prototype need?

*	Functional	7
*	Program management	7
*	Program (characteristics)	3
*	Regulatory	1
*	Threshold values	1
*	Priorities	1

What other software should the prototype interface with?

*	CSNAS	4
*	Scheduling software	3
*	AMS	2
*	CHART	1
*	ALP	1
*	Enable	1
*	Spreadsheet	1
*	None	1

Who needs to be convinced that a KS is appropriate?

*	Upper-level management (dictate use)	8
*	PMs	2
*	Workers	1

What are the important acceptance criteria?

*	Produce useful results	8
*	Show productivity improvement	4
*	Not time-consuming	2
*	Intelligent interaction	2

	1
* Useful in daily work	1
* Easy to learn	

6. Defining the Operational System

What environment should the operational system use?

	15
* Personal computer	4
* Mainframe	

What are long-term maintenance and enhancement requirements?

	10
* Relatively moderate changes	

7. Resources Required

How many different problems should the demonstration prototype solve?

	5
* Three	3
* Seven	1
* One	1
* Two	1
* Ten	1
* Twenty to thirty	

How many experts must be consulted?

	4
* Six to twelve	2
* Twenty to forty	1
* Three to four	1
* Fifty to one hundred	

Appendix B: Tasks

This appendix contains the tasks defined during the knowledge engineering phase of the thesis. All tasks and relationships are contained in Table B-1. The task number matches the number of the corresponding event in the Phase I model network. Letter identifiers are included when an event from the Phase I network was divided into multiple tasks. Table B-2 contains 23 tasks which may not apply to a schedule for a specific program.

TABLE VII. Tasks and Relationships

TASK	DESCRIPTION	PREDECESSORS	SUCCESSORS
1	DRAFT PMD	None	4 12
2	THREAT INPUT	5	18
3a	DRAFT ILSP	13a	3b 36 38
3b	COORDINATE ILSP	3a	48
4	ASSESS COST SCHED AS WBS	1	5 8 10
5	FINAL PMD	4	2 13a
7a	DRAFT CRISP	13a	7b 28 36 38
7b	COORDINATE CRISP	7a	48
8	NEW START REVIEW	4	9 10
9	SECURITY CLASS GUIDE	8 13a	11
10	IPR PREP	4 8	14
11	DD254 PREP & APPROVAL	9 14	40
12	AFLC PAD	1	40
13a	AFSC FORM 56 RECEIPT	5	3a 7a 9 13b
13b	AFSC FORM 56 RESPONSE	13a	14 24
14	IPR	10 13b	11 17 18 19 20a 21 22 23 27a
15	DRAFT SOW	20a 23	28 29 37
16	COST ESTIMATE	20a	27b 34 36
17	DEVELOP PMP	14	40
18	DRAFT SPEC	2 14	37 38

TABLE VII. Tasks and Relationships (Continued)

TASK	DESCRIPTION	PREDECESSORS	SUCCESSORS
19	DEVELOP TEMP	14	40
20a	WBS PREP	14	15 16 20b
20b	WBS APPROVAL	20a	36
21	MOA/MOU	14	40
22	SAFETY REQUIREMENTS	14	40
23	PROGRAM SCHEDULES	14	15 27b 33
24	DRAFT AP	13b 29	27b
27a	ASP PREP	14	27b
27b	ASP	16 23 24 27a	30a 31 33 34 35 37 43a
28	DATA PACKAGE PREP	7a 15	37 40
29	SOURCES SOUGHT SYNOP	15	24 36 37
30a	SS PLAN PREP	27b	30b 43c
30b	SS PLAN APPROVAL	30a	43b 45
31	J&A PREP	27b	39
33	PROGRAM BASELINE	23 27b	40
34	COST BASELINE	16 27b	40
35	AP APPROVAL	27b	36 39
36	FINAL SOW	3a 7a 15 20b 29 35 37	40
37	DRAFT RFP	15 18 27b 28 29	36 38
38	FINAL SPEC	3a 7a 18 37	40
39	J&A APPROVAL	31 35	40
40	ASD FORM 117	11 12 17 19 21 22 28 33 34 36 38 39	41
41	RFP PACKAGE PREP	40	44a
43a	SS STANDARDS PREP	27b	43b
43b	SS STANDARDS APPROVAL	30b 43a	47
43c	SS PROCEDURES	30a	47
44a	3-LTR & ASD REVIEWS	41	44b 45
44b	AFSC REVIEW	44a	47
45	RFP RELEASE	30b 44a	46
46	CONTRACTOR RESPONSE	45	47
47	SOURCE SELECTION	43b 43c 44b 46	48
48	CONTRACT AWARD	3b 7b 47	None

TABLE VIII Tasks Which may not Apply

TASK	DESCRIPTION
2	THREAT INPUT
7a	DRAFT CRISP
7b	COORDINATE CRISP
8	NEW START REVIEW
9	SECURITY CLASS GUIDE
11	DD254 PREP & APPROVAL
20b	WBS APPROVAL
21	MOA/MOU
24	DRAFT AP
27a	ASP PREP
27b	ASP
28	DATA PACKAGE PREP
29	SOURCES SOUGHT SYNOP
30a	SS PLAN PREP
30b	SS PLAN APPROVAL
31	J&A PREP
35	AP APPROVAL
37	DRAFT RFP
39	J&A APPROVAL
43a	SS STANDARDS PREP
43b	SS STANDARDS APPROVAL
43c	SS PROCEDURES
44b	AFSC REVIEW

The remainder of this appendix contains brief descriptions offices of primary responsibility (OPR), applicability, and duration estimations for each task.

DRAFT PMD: This task represents the receipt of a draft Program Management Directive (PMD). The OPR is the Program Manager (PM) and Program Element Monitor (PEM). This task is always required. Its duration is fixed at 0 days.

THREAT INPUT: This task represents the time required to obtain threat information concerning the program. The OPR is the PM and the Foreign Technology Division (FTD). This task is not required unless a threat to the system exists. Its duration is fixed at 1 month.

DRAFT ILSP: This task represents the time required to draft the Integrated Logistics Support Plan (ILSP). The OPR is the Deputy Program

Manager for Logistics (DPML). This task is always required but should not be on the critical path. Its duration is fixed at 3 months.

COORDINATE ILSP: This task represents the time required to coordinate the ILSP. The OPR is the DPML. This task is always required but should not be on the critical path. Its duration is fixed at 9 months.

ASSESS COST SCHED AS WBS: This task represents the time required to assess cost, schedule, acquisition strategy, and work breakdown structure. The OPR is the PM and the cost estimator (RWPE). This task is always required. Its duration is estimated as follows:

1. Assume six week baseline for medium R&D, big Production, 100% in-line installation
2. Subtract one week for small R&D (RDT&E < $50M)
3. Add three weeks for big R&D (RDT&E > $250M).
4. Subtract one week for small Production (production unit cost < $250M)
5. Add two weeks for joint service program
6. Add one week for group cost A (plan, design, modify)
7. Add 7 percent if 100% retrofit installations
8. Add 15 percent if both in line and retrofit installations
9. Maximum duration of eleven weeks

FINAL PMD: This task represents the time required to prepare, coordinate and sign the Final PMD. The OPR is the PM and the PEM. This task is always required. Its duration is fixed at 1 month.

DRAFT CRISP: This task represents the time required to draft the Computer Resources Integrated Support Plan (CRISP). The OPR is the PM and the engineer (RWE). This task is not required unless Mission Critical Computer Resources (MCCR) are used and Program Management Responsibility Transfer (PMRT) is planned. Its duration depends on the type of computer processing used and type of acquisition. Its duration is estimated as follows:

1. Assume 4 week baseline for serial processing and non-developmental item (NDI) acquisition
2. Add 3 week if other than serial processing parallel processing, neural networks, etc.)
3. Add 4 weeks if new development

COORDINATE CRISP: This task represents the time required to coordinate the CRISP. The OPR is the PM and RWE. This task is not required unless MCCR are used and PMRT is planned. Its duration depends on Air Training Command (ATC) requirements, joint service requirement, special intelligence (SI) requirements, and the source of maintenance and logistics support. Its duration is estimated as follows:

1. Assume 26 week baseline for no ATC training requirements, single using command, no SI requirements, and Air Force maintenance and logistics support.

2. Add 3 weeks for ATC training requirements.
3. Add 4 weeks for multiple using commands.
4. Add 4 weeks for SI requirements.
5. Add 2 weeks for contractor maintenance and logistics support.

NEW START REVIEW: This task represents the time required to prepare and brief the Corporate Review Group (CRG) on new work efforts. The OPR is the PM. This task is not required for follow-on work. Its duration is fixed at 3 weeks.

SECURITY CLASS GUIDE: This task represents the time required to prepare and approve a Security Classification Guide (SCG). The OPR is the PM and RWE. This task is not required for unclassified programs. Its duration depends on the level of classification, type of access, and the presence/absence of an existing SCG. Its duration is estimated as follows:

1. 4 months for new SCG for secret program with normal access.
2. 1 month to update existing SCG for secret program with normal access.
3. 5 months for new SCG for secret program with special access.
4. 2 months to update existing SCG for secret program with special access.
5. 2 months for new SCG for confidential program with normal access.
6. 2 weeks to update existing SCG for confidential program with normal access.
7. 3 months for new SCG for confidential program with special access.
8. 6 weeks to update existing SCG for confidential program with special access.

IPR PREP: This task represents the time required to prepare for the Internal Program Review (IPR). The OPR is the PM. This task is always required. Its duration is fixed at 2 months.

DD254 PREP & APPROVAL: This task represents the time required to prepare and approve DD Form 254 which authorizes contractors access to classified data. The OPR is the PM and RWE. This task is not required for unclassified programs. Its duration is fixed at 1 month and 2 weeks.

AFLC PAD: This task represents the time spent waiting for the Air Force Logistic Command Program Acquisition Directive (AFLC PAD). The OPR is the DPML. This task is always required. Its duration is fixed at 3 weeks.

AFSC FORM 56 RECEIPT: This task represents the time spent waiting for the AFSC Form 56. The OPR is the PM. This task is always required. Its duration is fixed at 1 month.

AFSC FORM 56 RESPONSE: This task represents the time required to review and respond to the AFSC Form 56. The OPR is the PM. This task is always required. Its duration is fixed at 1 week.

IPR: This task represents the time required for the IPR. This task is always required. The OPR is the PM. Its duration is fixed at 1 day.

DRAFT SOW: This task represents the time required to draft the Statement of Work (SOW). The OPR is the PM and RWE. This task is always required. Its duration depends on joint service involvement. Its duration is estimated as follows:

1. 30 days for single service program
2. 90 days for joint service program

COST ESTIMATE: This task represents the time required to develop a cost estimate. The OPR is the PM and RWPE. This task is always required. Its duration depends on the type/amount of dollars spent, production unit cost, joint system involvement, cost group, and type of installation. Its duration is estimated as follows:

1. Assume six week baseline for medium R&D, big Production, 100% in-line installation.
2. Subtract one week for small R&D (RDT&E < $50M).
3. Add three weeks for big R&D (RDT&E > $250M).
4. Subtract one week for small Production (production unit cost < $250K).
5. Add two weeks for joint service program.
6. Add one week for group cost A (plan, design, modify).
7. Add 7 percent if 100% retrofit installations.
8. Add 15 percent if both in-line and retrofit installations.
9. Maximum duration of eleven weeks.

DEVELOP PMP: This task represents the time required to develop the Program Management Plan (PMP). The OPR is the PM. This task is always required. Its duration is fixed at 2 months.

DRAFT SPEC: This task represents the time required to draft the technical specification. The OPR is the PM and RWE. This task is always required. Its duration depends on system complexity and joint service involvement. Its duration is estimated as follows:

1. Assume 140 day baseline for minor system complexity and single service program.
2. Add 20 days for major system complexity.
3. Add 30 days for joint service program.

DEVELOP TEMP: This task represents the time required to develop the Test and Evaluation Master Plan (TEMP). The OPR is the PM and the tester (RWNT). This task is always required. Its duration is fixed at 120 days.

WBS PREP: This task represents the time required to prepare the Work Breakdown Structure (WBS). The OPR is the PM and RWPE. This task is always required. Its duration is fixed at 1 week.

WBS APPROVAL: This task represents the time required to approve the WBS. The OPR is the PM and RWPE. This task is not required if the acquisition cost is less than $2M. Its duration depends on the amount/type of dollars spent and existence/absence of a waiver (if applicable). Its duration is estimated as follows:

1. Assume 1 week baseline for CAT 2 (RDT&E <= $200M and Production <= $1B).
2. Add 20 weeks for CAT 1 (RDT&E > $200M or Production > $1B) unless waived by OSD/CAIG.

MOA/MOU: This task represents the time required to prepare, coordinate, and sign Memorandum of Agreement/Memorandums of Understanding (MOA/MOU). The OPR is the PM and program control (RWPP). This task is not required unless other government agencies are involved in the program. Its duration is fixed at 130 days.

SAFETY REQUIREMENTS: This task represents the time required to form safety groups, review contractor analyses, and write SOW/SPEC paragraphs. The OPR is the safety officer (RWS). This task is always required. Its duration depends on operational safety requirements and joint service involvement. Its duration is estimated as follows:

1. Assume 2 week baseline for no operational safety requirements and single service program.
2. Add two weeks for operational safety requirements.
3. Add two weeks for joint service program.

PROGRAM SCHEDULES: This task represents the time required to prepare program schedules. The OPR is the PM. This task is always required. Its duration is fixed at 30 days.

DRAFT AP: This task represents the time required to draft the acquisition plan (AP). The OPR is the PM and the contracting officer (RWK). This task is not required for work that is within the scope of an existing contract. Its duration depends on the amount of dollars spent. Its duration is estimated as follows:

1. 5 days for informal plan (cost <= $100K).
2. 20 days for formal plan (cost > $100K).

ASP PREP: This task represents the time required to prepare for the acquisition strategy panel (ASP). The OPR is the PM and RWK. This task is required whenever an AP is prepared. Its duration is fixed at 1 month.

ASP: This task represents the time required for the ASP. The OPR is the PM and RWK. This task is required whenever an AP is prepared. Its

duration depends on the type/amount of dollars spent. Its duration is estimated as follows:

1. Assume 1 day baseline for ASD approval
2. Add 1 day for AFSC approval (RDT&E >= $200M or Production >= $1B)

DATA PACKAGE PREP: This task represents the time required to prepare the data package. The OPR is acquisition support (RWB). This task is not required for certain study efforts and NDI items which are not modified. Its duration depends on the type of work, system complexity, joint service involvement, and type of acquisition. Its duration is estimated as follows:

1. 71 days for new work, major complexity, and single service program
2. 62 days for new work, minor complexity, and single service program
3. 62 days for follow on work and single service program.
4. 28 days for single service, QRC program with major complexity
4. 21 days for single service, QRC program with minor complexity.
5. Add 21 days for joint service program with major complexity.
6. Add 14 days for joint service program with minor complexity.

SOURCES SOUGHT SYNOP: This task represents the time required to prepare, publish, and receive replies to the sources sought synopsis. The OPR is the PM and RWK. This task is required whenever the cost exceeds $25K unless the PM can justify proceeding without it. Its duration depends on the type of contract pursued. Its duration is estimated as follows:

1. 27 days for sole source acquisitions.
2. 42 days for competitive acquisitions.

SS PLAN PREP: This task represents the time required to prepare the source selection plan. The OPR is the PM. This task is not required for sole source acquisitions or follow-on work. Its duration depends on the type/amount of dollars spent. Its duration is estimated as follows:

1. 9 weeks for ultimate approval at SAF/AQ (RDT&E > $100M or Production > $500M).
2. 6 weeks for ultimate approval at ASD/CC ($50M - $100M RDT&E or $100M - $500M Production).
3. 3 weeks for approval at two-letter (RDT&E < $50M and Production < $100M or delegated by ASD/CC).

SS PLAN APPROVAL: This task represents the time required to coordinate and approve the source selection plan. The OPR is the PM. This task is not required for sole source acquisitions or follow-on work. Its duration depends on the type/amount of dollars spent. Its duration is estimated as follows:

1. 6 weeks for ultimate approval at SAF/AQ (RDT&E > $100M or Production > $500M).
2. 13 days for ultimate approval at ASD/CC ($50M - $100M RDT&E or $100M - $500M Production)
3. 3 weeks for approval at two-letter when delegated by ASD/CC.
4. 1 week for approval at two-letter (RDT&E < $50M and Production < $100M).

J&A PREP: This task represents the time required to prepare Justification & Approval documentation. The OPR is RWK. This task is required only for other than full and open competition. Its duration is fixed at 1 week.

PROGRAM BASELINE: This task represents the time required to develop the program baseline. The OPR is the PM. This task is always required. Its duration depends on joint service involvement. Its duration is estimated as follows:

1. 3 months for single service program
2. 6 months joint service program

COST BASELINE: This task represents the time required to develop the cost baseline. The OPR is the PM and RWPE. This task is always required. Its duration depends on the designation of the program, joint service involvement, and presence/absence of an existing cost baseline. Its duration is estimated as follows:

1. Assume 3 week baseline for "OTHER THAN DESIGNATED" program, single service program, and no existing cost baseline.
2. Add 1 week for revision of existing cost baseline.
3. Add 1 week for "SAR" designation.
4. Add 2 days for "DESIGNATED" designation.
5. Add 1 week for joint service program.

AP APPROVAL: This task represents the time required to coordinate and approve the AP. The OPR is the PM and RWK. This task is not required for work that is within the scope of an existing contract. Its duration depends on the amount of R&D dollars spent. Its duration is estimated as follows:

1. 4 days for approval at ASD or below (RDT&E < $5M).
2. 124 days for approval at SAF (RDT&E > $5M)

FINAL SOW: This task represents the time required to finalize the SOW. The OPR is the PM and RWE. This task is always required. Its duration is fixed at 40 days.

DRAFT RFP: This task represents the time required to prepare and receive a response to the draft request for proposal (RFP). The OPR is the PM and RWK. This task is required for competitive procurements exceeding $25M unless the PM can justify otherwise (routine program, standard contract). Its duration is fixed at 17 days.

FINAL SPEC: This task represents the time required to finalize the technical specification. The OPR is the PM and RWE. This task is always required. Its duration depends on the system complexity and joint service involvement. Its duration is estimated as follows:

1. Assume 110 day baseline for minor system complexity and single service program.
2. Add 15 days for major system complexity.
3. Add 25 days for joint service program.

J&A APPROVAL: This task represents the time required to approve the Justification & Approval documentation. The OPR is RWK. This task is required only for other than full and open competition. Its duration depends on the amount of dollars spent and the amount of controversy involved. Its duration is estimated as follows:

1. 30 days for cost less than $30M.
2. 60 days for routine approval at SAF/AQ (cost > $30M).
3. 90 days for controversial programs approved at SAF/AQ (cost > $30M).

ASD FORM 117: This task represents the time required to coordinate and approve ASD Form 117 (purchase request checklist). This task is always required. The OPR is the PM. Its duration is fixed at 3 weeks.

RFP PACKAGE PREP: This task represents the time required to prepare the RFP package. The OPR is RWK. This task is always required. Its duration is fixed at 14 days.

SS STANDARDS PREP: This task represents the time required to prepare the source selection standards. This task is not required for sole source acquisitions or follow-on work. Its duration depends on the type/amount of dollars spent. Its duration is estimated as follows:

1. 9 weeks for ultimate approval at SAF/AQ (RDT&E > $100M or Production > $500M).
2. 6 weeks for ultimate approval at ASD/CC ($50M - $100M RDT&E or $100M - $500M Production).
3. 3 weeks for approval at two-letter (RDT&E < $50M and Production < $100M or delegated by ASD/CC).

SS STANDARDS APPROVAL: This task represents the time required to coordinate and approve the source selection standards. The OPR is the PM. This task is not required for sole source acquisitions or follow-on work. Its duration depends on the type/amount of dollars spent. Its duration is estimated as follows:

1. 6 weeks for ultimate approval at SAF/AQ (RDT&E > $100M or Production > $500M).
2. 13 days for ultimate approval at ASD/CC ($50M - $100M RDT&E or $100M - $500M Production).

3. 3 weeks for approval at two-letter when delegated by ASD/CC.
4. 1 week for approval at two-letter (RDT&E < $50M and Production < $100M).

SS PROCEDURES: This task represents the time required to prepare source selection procedures and train the source selection team. The OPR is the PM. This task is not required for sole source acquisitions or follow-on work. Its duration depends on the type/amount of dollars spent. Its duration is estimated as follows:

1. Assume 10 day baseline to prepare the procedures.
2. Add 4 weeks to schedule and conduct training for source selections approved by ASD and higher (RDT&E > $50M or Production > $100M unless approval delegated to RW).

3-LTR & ASD REVIEWS: This task represents the time required to prepare for and execute reviews of the RFP package at ASD and below. The OPR is the PM. This task is always required. Its duration depends on the value of the RFP package. Its duration is estimated as follows:

1. Assume 15 day baseline for preparation and 3-LTR review.
2. Add 5 days for ASD review (RFP > $25M).

AFSC REVIEW: This task represents the time required to execute review of the RFP package at AFSC. The OPR is the PM. This task is not required unless the RFP exceeds $75M. Its duration is fixed at 5 weeks.

RFP RELEASE: This task represents the time required to finalize and release the RFP. The OPR is RWK. This task is always required. Its duration is dependent on the type of contract and RFP value. Its duration is estimated as follows:

1. Assume 5 day baseline for final modifications.
2. Add 55 days for single step sealed bid contract.
3. Add 40 days for two step sealed bid contract.
4. Add 40 days for priced orders under BOA.
5. Add 40 days for sole source contract exceeding $25M.
6. Add 30 days for sole source contract under $25M.
7. Add 40 days for competitive contract exceeding $3.5M.
8. Add 30 days for competitive contract under $3.5M.
9. Add 25 days for long lead contract, letter contract, or unpriced order under BOA.

CONTRACTOR RESPONSE: This task represents the time spent waiting for contractor responses. The OPR is the contractor. This task is always required. Its duration is dependent on the type of contract and RFP value. Its duration is estimated as follows:

1. 45 days for single step sealed bid contract.
2. 60 days for two step sealed bid contract.
3. 60 days for priced orders under BOA.
4. 60 days for sole source contract exceeding $25M.

5. 50 days for sole source contract under $25M.
6. 60 days for competitive contract exceeding $3.5M.
7. 50 days for competitive contract under $3.5M.
8. 50 days for long lead contract or letter contract.
9. 30 days for unpriced order under BOA.

SOURCE SELECTION: This task represents the time required to select the source The OPR is the PM. This task is always required. Its duration is dependent n the type of contract and RFP value. Its duration is estimated as follows:

1. 95 days for priced orders under BOA.
2. 80 days for sole source contract exceeding $25M.
3. 68 days for sole source contract under $25M.
4. 67 days for competitive contract exceeding $25M.
5. 65 days for competitive contract in the $3.5M - $25M range.
6. 37 days for competitive contract under $3.5M.
7. 68 days for long lead contract, letter contract, or unpriced order under BOA.

CONTRACT AWARD: This task represents the time required to write, review, approve and award the contract. The OPR is RWK. This task is always required. Its duration is dependent on the type of contract and RFP value. Its duration is estimated as follows:

1. 54 days for single step sealed bid contract.
2. 64 days for two step sealed bid contract.
3. 35 days for priced orders under BOA.
4. 35 days for sole source contract exceeding $25M.
5. 27 days for sole source contract under $25M.
6. 43 days for competitive contract exceeding $3.5M.
7. 32 days for competitive contract under $3.5M.
8. 27 days for long lead contract, letter contract, or unpriced order under BOA.

Appendix C. Program Characteristics

This appendix contains listings of the program characteristics identified during the knowledge engineering phases of the thesis. Table C-1 contains all the program characteristics and their acceptable values. Table C-2 cross-references the program characteristics with the tasks (number from Table B-1) which use the program characteristic values.

Table IX. Program Characteristics and Acceptable Values

CHARACTERISTIC	ACCEPTABLE VALUES
PROGRAM NAME	String
SYSTEM THREAT	Yes, No
RDTE DOLLARS ($M)	2, 5, 10, 25, 50, 100, 200, 250
PRODUCTION DOLLARS ($M)	2, 5, 10, 50, 100, 500, 1000
PRODUCTION UNIT COST ($K)	250
COST GROUP	A, B
TYPE OF INSTALLATION	Inline, Retrofit, Both
OTHER SERVICES	Acquisition, Other, Both, None
MCCR INVOLVED	Yes, No
PMRT PLANNED	Yes, No, NA
NDI ACQUISITION	Yes, No
TYPE OF PROCESSING	Serial, Other, NA
MAINT/LOG SUPPORT	Air Force, Contractor, NA
ATC TRAINING	Yes, No, NA
SPECIAL INTELLIGENCE	Yes, No
PROGRAM CLASSIFICATION	Unclassified, Confidential, Secret
EXISTING SCG	Yes, No, NA
WBS APPROVAL WAIVER	Yes, No, NA
MOA/MOU NEEDED	Funding, Other, None
OPERATIONAL SAFETY	Yes, No
DATA PACKAGE NEEDED	Yes, No
RFP VALUE ($K)	25, 3500, 25000, 75000
SS DELEGATED TO RW	Yes, No, NA

Table IX. Program Characteristics and Acceptable Values (Continued)

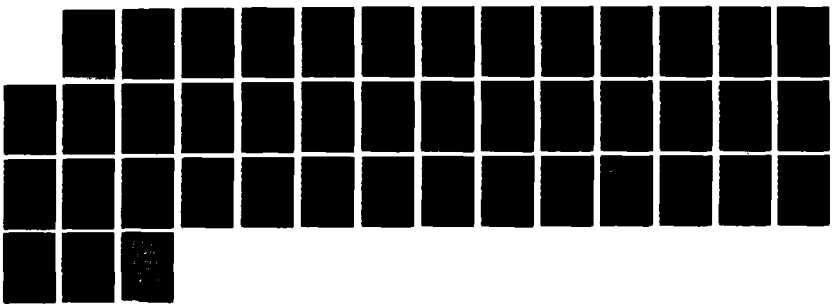
CHARACTERISTIC	ACCEPTABLE VALUES
TYPE OF CONTRACT	One Step Sealed Bid, Two Step Sealed Bid, Priced Order Under BOA, Unpriced Order Under BOA, Long Lead, Letter Contract, Sole Source, Competitive
QRC PROGRAM	Yes, No
EXISTING COST BASELINE	Yes, No
PROGRAM DESIGNATION	SAR, Designated, Other
IN SCOPE OF CONTRACT	Yes, No
CONTROVERSIAL	Yes, No
FOLLOW-ON PROGRAM	Yes, No
SYSTEM COMPLEXITY	Yes, No
SOURCES SOUGHT SYNOPSIS NEEDED	Yes, No, NA
DRAFT RFP NEEDED	Yes, No, NA
PROGRAM START DATE	Date
CONTRACT AWARD DATE	Date

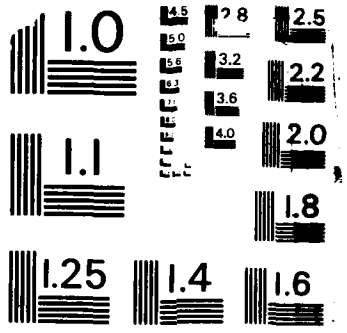
Table X. Cross-reference of Characteristics and Tasks

CHARACTERISTIC	TASKS USED BY
PROGRAM NAME	None
SYSTEM THREAT	2 18
RDTE DOLLARS ($M)	4 16 20b 24 27b 29 30a 30b 35 36 39 43a 43b 43c
PRODUCTION DOLLARS ($M)	20b 24 27b 30a 30b 35 39 43a 43b 43c
PRODUCTION UNIT COST ($K)	4 16
COST GROUP	4 16
TYPE OF INSTALLATION	4 16
OTHER SERVICES	4 7b 15 18 33 34 38
MCCR INVOLVED	7a 7b 28
PMRT PLANNED	7a
NDI ACQUISITION	7a
TYPE OF PROCESSING	7a
MAINT/LOG SUPPORT	7b
ATC TRAINING	7b
SPECIAL INTELLIGENCE	7b

AD-A190 481 ISA: A PROTOTYPE INTELLIGENT SCHEDULING ASSISTANT FOR DEFENSE ACQUISITION MANAGEMENT(U) AIR FORCE INST OF TECH WRIGHT-PATTERSON AF OH SCHOOL OF ENGI 2/2

UNCLASSIFIED J L MORAN MAR 88 AFIT/GE/ENG/88M-9 F/G 12/9 NL





MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU OF STANDARDS 1963-A

Table IX. Program Characteristics and Acceptable Values (Continued)

CHARACTERISTIC	ACCEPTABLE VALUES
TYPE OF CONTRACT	One Step Sealed Bid, Two Step Sealed Bid, Priced Order Under BOA, Unpriced Order Under BOA, Long Lead, Letter Contract, Sole Source, Competitive
QRC PROGRAM	Yes, No
EXISTING COST BASELINE	Yes, No
PROGRAM DESIGNATION	SAR, Designated, Other
IN SCOPE OF CONTRACT	Yes, No
CONTROVERSIAL	Yes, No
FOLLOW-ON PROGRAM	Yes, No
SYSTEM COMPLEXITY	Yes, No
SOURCES SOUGHT SYNOPSIS NEEDED	Yes, No, NA
DRAFT RFP NEEDED	Yes, No, NA
PROGRAM START DATE	Date
CONTRACT AWARD DATE	Date

Table X. Cross-reference of Characteristics and Tasks

CHARACTERISTIC	TASKS USED BY
PROGRAM NAME	None
SYSTEM THREAT	2 18
RDTE DOLLARS ($M)	4 16 20b 24 27b 29 30a 30b 35 36 39 43a 43b 43c
PRODUCTION DOLLARS ($M)	20b 24 27b 30a 30b 35 36 39 43a 43b 43c
PRODUCTION UNIT COST ($K)	4 16
COST GROUP	4 16
TYPE OF INSTALLATION	4 16
OTHER SERVICES	4 7b 15 16 18 22 28 33 34 38
MCCR INVOLVED	7a 7b 28 36 38 48
PMRT PLANNED	7a
NDI ACQUISITION	7a
TYPE OF PROCESSING	7a
MAINT/LOG SUPPORT	7b
ATC TRAINING	7b
SPECIAL INTELLIGENCE	7b 9

Table X. Cross-reference of Characteristics and Tasks (Continued)

CHARACTERISTIC	TASKS USED BY
PROGRAM CLASSIFICATION	9 11 12 40
EXISTING SCG	9
WBS APPROVAL WAIVER	20b
MOA/MOU NEEDED	21 40
OPERATIONAL SAFETY	22
DATA PACKAGE NEEDED	28 37 40
RFP VALUE ($K)	29 44a 44b 45 46 47 48
SS DELEGATED TO RW	30b 43b 43c
TYPE OF CONTRACT	29 30a 30b 31 39 40 43a 43b 43c 45 46 47 48
QRC PROGRAM	28
EXISTING COST BASELINE	34
PROGRAM DESIGNATION	34
IN SCOPE OF CONTRACT	24 27a 27b 30a 30b 31 33 34 35 36 37 39 43a 43b 43c 45 47 48
CONTROVERSIAL	39
FOLLOW-ON PROGRAM	8 9 28 30a 30b 31 39 40 43a 43b 43c 45 47 48
SYSTEM COMPLEXITY	18 28 38
SOURCES SOUGHT SYNOPSIS NEEDED	24 29 36 37
DRAFT RFP NEEDED	36 37 38 40
PROGRAM START DATE	1
CONTRACT AWARD DATE	48

Appendix D: Rules for Schedule Generation

This appendix contains the rules used for schedule generation. The rules are written in Guru. Variable "ASKDONE" is set to "TRUE" before consulting the rule set. All other variable names are in lower case letters. Variables beginning with the letter "t" are initialized to "UNKNOWN" before consulting the rules. Variables beginning with the letter "v" are initialized to the values of the program characteristics before consulting the rules.

```
RULE:   RDPMD
        PRIORITY:  90
        TEST:   E
        IF:     ASKDONE
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "    0"
                CURRTASK.DESCR = "DRAFT PMD                "
                CURRTASK.OPR = "PM & PEM     "
                CURRTASK.US = vstart
                CURRTASK.UC = vstart
                tdpmd = CURRTASK.ID
        REASON: The start event for the network is the Draft PMD.  Its
                actual duration takes months.  However, it is treated as
                a milestone event for schedule development.
        COMMENT: Draft PMD.  Sources:  LTC Hollingsworth, x54811; LTC
                Sikra, x53969.
                Event 1 in RW Phase I network, 30 Aug 85.


RULE:   RTHRT
        PRIORITY:  80
        TEST:   E
        IF:     vthreat = "YES" and KNOWN(tfpmd)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "   23"
                CURRTASK.DESCR = "THREAT INPUT             "
                CURRTASK.OPR = "PM & FTD     "
                tthreat = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tfpmd
                CURRLINK.SUCC = tthreat
        REASON: Threat Input follows receipt of the Final PMD and takes
```

```
        4 weeks to complete.  This task is not used if there is
        not a threat to the system being developed.
COMMENT: Threat Input.  Source, LT Rohner, x?????; LTC
        Hollingsworth, x54811.
        Event 2 in RW Phase I network, 30 Aug 85.


RULE:   RDILSP
        PRIORITY:  80
        TEST:   E
        IF:     KNOWN(tform56i)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "   65"
                CURRTASK.DESCR = "DRAFT ILSP                "
                CURRTASK.OPR = "DPML        "
                tdilsp = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tform56i
                CURRLINK.SUCC = tdilsp
REASON: Draft ILSP follows receipt of AFSC Form 56 takes 65 days
        to complete.
COMMENT: Draft ILSP.  Source, John Shawhan, x52108.
        Event 3 in RW Phase I network, 30 Aug 85.


RULE:   RCILSP
        PRIORITY:  80
        TEST:   E
        IF:     KNOWN(tdilsp)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "  195"
                CURRTASK.DESCR = "COORDINATE  ILSP        "
                CURRTASK.OPR = "DPML        "
                tcilsp = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tdilsp
                CURRLINK.SUCC = tcilsp
REASON: Coordinate ILSP follows Develop ILSP and takes 195 days
        to complete.
COMMENT: Coordinate ILSP.  Source, John Shawhan, x52108.
        Event 3 in RW Phase I network, 30 Aug 85.


RULE:   RACSAW
        PRIORITY:  90
        TEST:   E
        IF:     KNOWN(tdpmd)
        THEN:   PERFORM ADDTASK
                duration = 40
                IF vrdte < 50 THEN
                  duration = duration - 5
                ELSE
```

```
            IF vrdte > 250 THEN
              duration = duration + 15
            ENDIF
          ENDIF
          IF vunit < 250 THEN
            duration = duration - 5
          ENDIF
          IF vjoint = "NONE" THEN
            duration = duration - 10
          ENDIF
          IF vgroup = "A" THEN
            duration = duration + 5
          ENDIF
          IF vinst = "RETROFIT" THEN
            duration = duration * .07
          ELSE
            IF vinst = "BOTH" THEN
              duration = duration * .15
            ENDIF
          ENDIF
          IF duration > 55 THEN
            duration = 55
          ENDIF
          CURRTASK.DUR = TOSTR(duration,5,0)
          CURRTASK.DESCR = "ASSESS COST SCHED AS WBS"
          CURRTASK.OPR = "PM & RWPE    "
          tacsaw = CURRTASK.ID
          ATTACH 1 TO CURRLINK
          CURRLINK.PRED = tdpmd
          CURRLINK.SUCC = tacsaw
    REASON: Assess Cost, Schedule, Acquisition Strategy, and Work
            Breakdown Structure follows completion of Draft PMD.
            Its duration depends on RDTE cost, Production Unit Cost,
            Joint service involvement, Cost Group, and Type of
            Installation.  It takes no longer than 11 weeks to
            complete.
    COMMENT: Assess Cost, Schedule, Acquisition Strategy, and Work
            Breakdown Structure.  Source, John Holdren, x52651.
            Event 4 in RW Phase I network, 30 Aug 85.


RULE:   RFPMD
        PRIORITY:  90
        TEST:   E
        IF:     KNOWN(tacsaw)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "   23"
                CURRTASK.DESCR = "FINAL PMD                    "
                CURRTASK.OPR = "PM & PEM     "
                tfpmd = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tacsaw
```

```
            CURRLINK.SUCC = tfpmd
        REASON: The Final PMD follows Assessment of Cost, Schedule,
               Acquisition Strategy, and Work Breakdown Structure.
               It takes 4 weeks to complete.
        COMMENT: Final PMD.  Source:  LTC Hollingsworth, x54811.
               Event 5 in RW Phase I network, 30 Aug 85.


RULE:   RDCRISP
        PRIORITY:  80
        TEST:   E
        IF:     vmccr = "YES" and vpmrt = "YES" and KNOWN(tform56i)
        THEN:   PERFORM ADDTASK
                duration = 25
                IF vproc = "SERIAL" THEN
                  duration = duration - 5
                ENDIF
                IF vndi <> "NO" THEN
                  duration = duration + 20
                ENDIF
                CURRTASK.DUR = TOSTR(duration,5,0)
                CURRTASK.DESCR = "DRAFT CRISP              "
                CURRTASK.OPR = "PM & RWE     "
                tdcrisp = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tform56i
                CURRLINK.SUCC = tdcrisp
        REASON: Draft CRISP follows receipt of AFSC Form 56.  Its
               duration depends on the type of processing involved
               complete.  This task is not needed unless the system
               uses Mission Critical Computer Resources.
        COMMENT: Draft CRISP.  Sources: Capt Schmitt, x52665; LTC
               Hollingsworth, x54811.
               Event 7 in RW Phase I network, 30 Aug 85.


RULE:   RCCRISP
        PRIORITY:  80
        TEST:   E
        IF:     KNOWN(tdcrisp)
        THEN:   PERFORM ADDTASK
                duration = 150
                IF vatc = "YES" THEN
                  duration = duration + 15
                ENDIF
                IF vjoint = "NONE" THEN
                  duration = duration - 20
                ENDIF
                IF vsi = "YES" THEN
                  duration = duration + 20
                ENDIF
                IF vmaint = "CONTRACTOR" THEN
```

```
            duration = duration + 10
          ENDIF
          CURRTASK.DUR = TOSTR(duration,5,0)
          CURRTASK.DESCR = "COORDINATE  CRISP       "
          CURRTASK.OPR = "PM & RWE    "
          tccrisp = CURRTASK.ID
          ATTACH 1 TO CURRLINK
          CURRLINK.PRED = tdcrisp
          CURRLINK.SUCC = tccrisp
REASON: Coordinate CRISP follows develop CRISP.  Its duration
        depends on the ATC training requirements, Joint service
        involvement, Special Intelligence requirements, and the
        source of maintenance and logistics support.  This task
        is not needed unless the system uses Mission Critical
        Computer Resources.
COMMENT: Coordinate CRISP.  Sources: Capt Schmitt, x52665; LTC
        Hollingsworth, x54811.
        Event 7 in RW Phase I network, 30 Aug 85.


RULE:   RNSR
        PRIORITY:  90
        TEST:   E
        IF:     vfollow = "NO" and KNOWN(tacsaw)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "   15"
                CURRTASK.DESCR = "NEW START   REVIEW      "
                CURRTASK.OPR = "PM          "
                tnsr = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tacsaw
                CURRLINK.SUCC = tnsr
        REASON: The New Start Review follows Assessment of Cost,
                Schedule, Acquistion Strategy, and Work Breakdown
                Structure.  Its duration is 15 days including
                preparation.  It is not needed for follow-on programs.
        COMMENT: New Start Review.  Sources; LTC  Hollingsworth, x54811;
                LTC Sikra, x53969.
                Event 8 in RW Phase I network, 30 Aug 85.


RULE:   RSCG
        PRIORITY:  80
        TEST:   E
        IF:     vclass <> "UNCLASSIFIED" and KNOWN(tform56i) and
                (KNOWN(tnsr) or vfollow = "YES")
        THEN:   PERFORM ADDTASK
                IF vclass = "CONFIDENTIAL" and vscg = "NO" THEN
                  duration = 45
                ENDIF
                IF vclass = "CONFIDENTIAL" and vscg = "YES" THEN
                  duration = 10
```

```
                ENDIF
                IF vclass = "CONFIDENTIAL" and vsi = "YES" and \
                        vscg = "NO" THEN
                  duration = 68
                ENDIF
                IF vclass = "CONFIDENTIAL" and vsi = "YES" and \
                        vscg = "YES" THEN
                  duration = 30
                ENDIF
                IF vclass = "SECRET" and vscg = "NO" THEN
                  duration = 90
                ENDIF
                IF vclass = "SECRET" and vscg = "YES" THEN
                  duration = 23
                ENDIF
                IF vclass = "SECRET" and vsi = "YES" and \
                         vscg = "NO" THEN
                  duration = 113
                ENDIF
                IF vclass = "SECRET" and vsi = "YES" and \
                         vscg = "YES" THEN
                  duration = 45
                ENDIF
                CURRTASK.DUR = TOSTR(duration,5,0)
                CURRTASK.DESCR = "SECURITY    CLASS GUIDE "
                CURRTASK.OPR = "PM & RWE     "
                tscg = CURRTASK.ID
                IF KNOWN(tnsr) THEN
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tnsr
                  CURRLINK.SUCC = tscg
                ENDIF
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tform56i
                CURRLINK.SUCC = tscg
        REASON: Security Classification Guide preparation follows the
               New Start Review (if used) and receipt of AFSC Form 56.
               This task is not used if the program is unclassified.
        COMMENT: Security Classification Guide.  Sources: Dave Garcher,
               x53218; LTC Hollingsworth, x54811.
               Event 9 in RW Phase I network, 30 Aug 85.


RULE:   RIPRP
        PRIORITY:  90
        TEST:   E
        IF:     (vfollow = "YES" and KNOWN(tacsaw)) or KNOWN(tnsr)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "   45"
                CURRTASK.DESCR = "IPR PREP                "
                CURRTASK.OPR = "PM          "
                tiprp = CURRTASK.ID
```

```
            IF KNOWN(tnsr) THEN
              ATTACH 1 TO CURRLINK
              CURRLINK.PRED = tnsr
              CURRLINK.SUCC = tiprp
            ELSE
              ATTACH 1 TO CURRLINK
              CURRLINK.PRED = tacsaw
              CURRLINK.SUCC = tiprp
            ENDIF
      REASON: IPR preparation follows the New Start Review (if used).
              IPR preparation follows Assessment of Cost, Schedule,
              Acquisition Strategy, and Work Breakdown Structure for
              follow-on programs.  Its duration is 45 days.
      COMMENT: IPR Preparation.  Sources:  LTC  Hollingsworth, x54811;
              LTC Sikra, x53969.
               Event 10 in RW Phase I network, 30 Aug 85.


RULE:   RDD254
        PRIORITY:  70
        TEST:   E
        IF:     KNOWN(tscg) and KNOWN(tipr)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "   33"
                CURRTASK.DESCR = "DD 254 PREP & APPROVAL  "
                CURRTASK.OPR = "PM & RWE     "
                tdd254 = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tscg
                CURRLINK.SUCC = tdd254
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tipr
                CURRLINK.SUCC = tdd254
        REASON: DD Form 254 preparation and approval follows the
                Security Classification Guide and Internal Program
                Review.  Its duration is 33 days.  This task is not used
                in unclassified programs.
        COMMENT: DD Form 254.  Source:  Dave Garcher, x53218; LTC
                Hollingsworth, x54811.
                Event 11 in RW Phase I network, 30 Aug 85.


RULE:   RPAD
        PRIORITY:  90
        TEST:   E
        IF:     KNOWN(tdpmd)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "   15"
                CURRTASK.DESCR = "AFLC PAD                "
                CURRTASK.OPR = "DPML        "
                tpad = CURRTASK.ID
                ATTACH 1 TO CURRLINK
```

```
            CURRLINK.PRED = tdpmd
            CURRLINK.SUCC = tpad
        REASON: The AFLC Program Acquisition Directive follows the
               receipt of the Final PMD.  Its duration is 15 days.
        COMMENT: AFLC PAD.  Source:  LTC Hughes, x54852.
               Event 12 in RW Phase I network, 30 Aug 85.


RULE:   RFORM56I
        PRIORITY:  90
        TEST:   E
        IF:     KNOWN(tfpmd)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "   23"
                CURRTASK.DESCR = "AFSC FORM 56 RECEIPT    "
                CURRTASK.OPR = "PM          "
                tform56i = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tfpmd
                CURRLINK.SUCC = tform56i
        REASON: AFSC Form 56 is received after the Final PMD is
               completed.  Its duration is 23 days.
        COMMENT: AFSC Form 56 Receipt.  Sources:  LTC Hollingsworth,
               x54811;  LTC Sikra, x53969.
               Event 13 in RW Phase I network, 30 Aug 85.


RULE:   RFORM56O
        PRIORITY:  90
        TEST:   E
        IF:     KNOWN(tform56i)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "    5"
                CURRTASK.DESCR = "AFSC FORM 56 RESPONSE   "
                CURRTASK.OPR = "PM          "
                tform56o = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tform56i
                CURRLINK.SUCC = tform56o
        REASON: AFSC Form 56 must be reviewed and a response made in 5
               days.
        COMMENT: AFSC Form 56 Response.  Sources:  LTC Hollingsworth,
               x54811; LTC Sikra, x53969.
               Event 13 in RW Phase I network, 30 Aug 85.


RULE:   RIPR
        PRIORITY:  80
        TEST:   E
        IF:     KNOWN(tform56o) and KNOWN(tiprp)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "    1"
```

```
            CURRTASK.DESCR = "IPR                          "
            CURRTASK.OPR = "PM          "
            tipr = CURRTASK.ID
            ATTACH 1 TO CURRLINK
            CURRLINK.PRED = tform56o
            CURRLINK.SUCC = tipr
            ATTACH 1 TO CURRLINK
            CURRLINK.PRED = tiprp
            CURRLINK.SUCC = tipr
      REASON: The Internal Program Review follows the response to the
              AFSC Form 56 and IPR preparation.  Its duration is 1
              day.
      COMMENT: IPR.  Sources:  LTC Hollingsworth, x54811; LTC Sikra,
              x53969.
              Event 14 in RW Phase I network, 30 Aug 85.


RULE:   RDSOW
        PRIORITY:  80
        TEST:   E
        IF:     KNOWN(twbsp) and KNOWN(tsched)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "   90"
                IF vjoint = "NONE" THEN
                  CURRTASK.DUR = "   30"
                ENDIF
                CURRTASK.DESCR = "DRAFT SOW                "
                CURRTASK.OPR = "PM & RWE    "
                tdsow = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = twbsp
                CURRLINK.SUCC = tdsow
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tsched
                CURRLINK.SUCC = tdsow
        REASON: The Draft SOW follows WBS, Program Schedules, and Draft
                ILSP.  Its duration depends on joint service
                involvement.
        COMMENT: Draft SOW.  Source:  LTC Hollingsworth, x54811.
                Event 15 in RW Phase I network, 30 Aug 85.


RULE:   RCOSTEST
        PRIORITY:  70
        TEST:   E
        IF:     KNOWN(twbsp)
        THEN:   OBTAIN FROM CURRTASK FOR
                  DESCR = "ASSESS COST SCHED AS WBS"
                duration = CURRTASK.DUR
                PERFORM ADDTASK
                CURRTASK.DUR = duration
                CURRTASK.DESCR = "COST        ESTIMATE    "
```

```
            CURRTASK.OPR = "PM & RWPE    "
            tcostest = CURRTASK.ID
            ATTACH 1 TO CURRLINK
            CURRLINK.PRED = twbsp
            CURRLINK.SUCC = tcostest
      REASON: The Post-IPR Cost Estimate follows the IPR and WBS
              preparation.  Its duration is the same as the first cost
              estimate.
      COMMENT: Post-IPR Cost Estimate.  Source:  John Holdren, x52651.
              Event 16 in RW Phase I network, 30 Aug 85.


RULE:   RPMP
        PRIORITY:  80
        TEST:   E
        IF:     KNOWN(tipr)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "   45"
                CURRTASK.DESCR = "DEVELOP PMP              "
                CURRTASK.OPR = "PM          "
                tpmp = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tipr
                CURRLINK.SUCC = tpmp
        REASON: The Program Management Plan follows the IPR.  Its
                duration is 45 days.
        COMMENT: PMP.  Source:  LTC Hollingsworth, x52651.
                Event 17 in RW Phase I network, 30 Aug 85.


RULE:   RDSPEC
        PRIORITY:  80
        TEST:   E
        IF:     KNOWN(tipr) and (KNOWN(tthreat) or vthreat <> "YES")
        THEN:   PERFORM ADDTASK
                duration = 140
                IF vscomp = "MAJOR" THEN
                  duration = duration + 20
                ENDIF
                IF vjoint = "NONE" THEN
                  duration = duration - 30
                ENDIF
                CURRTASK.DUR = TOSTR(duration,5,0)
                CURRTASK.DESCR = "DRAFT SPEC               "
                CURRTASK.OPR = "PM & RWE    "
                tdspec = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tipr
                CURRLINK.SUCC = tdspec
                IF KNOWN(tthreat) THEN
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tthreat
```

```
            CURRLINK.SUCC = tdspec
          ENDIF
REASON: The Draft Specification follows the IPR and Threat Input
        (if used).  Its duration depends on System Complexity
        and Joint service involvement.
COMMENT: Draft Specification.  Source:  Phase I Scheduling
        Guide.
        Event 18 in RW Phase I network, 30 Aug 85.


RULE:   RTEMP
        PRIORITY:  80
        TEST:   E
        IF:     KNOWN(tipr)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "  120"
                CURRTASK.DESCR = "DEVELOP TEMP              "
                CURRTASK.OPR = "PM & RWNT    "
                ttemp = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tipr
                CURRLINK.SUCC = ttemp
        REASON: Development of the TEMP follows the IPR.  Its duration
                is 120 days.
        COMMENT: Develop TEMP.  Source:  Phase I Scheduling Guide.
                Event 19 in RW Phase I network, 30 Aug 85.


RULE:   RWBSP
        PRIORITY:  80
        TEST:   E
        IF:     KNOWN(tipr)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "    5"
                CURRTASK.DESCR = "WBS PREP                  "
                CURRTASK.OPR = "PM & RWPE    "
                twbsp = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tipr
                CURRLINK.SUCC = twbsp
        REASON: WBS Preparation follows the IPR.  Its duration is 5
                days.
        COMMENT: WBS Preparation.  Source:  John Holdren, x52651.
                Event 20 in RW Phase I network, 30 Aug 85.


RULE:   RWBSA
        PRIORITY:  80
        TEST:   E
        IF:     KNOWN(twbsp) and (vrdte > 2 or vprod > 2)
        THEN:   PERFORM ADDTASK
                duration = 10
```

```
            IF vrdte > 200 or vprod > 1000 THEN
              IF vwbs = "NO" THEN
                duration = 100
              ENDIF
            ENDIF
            CURRTASK.DUR = TOSTR(duration,5,0)
            CURRTASK.DESCR = "WBS APPROVAL                "
            CURRTASK.OPR = "PM & RWPE    "
            twbsa = CURRTASK.ID
            ATTACH 1 TO CURRLINK
            CURRLINK.PRED = twbsp
            CURRLINK.SUCC = twbsa
      REASON: WBS Approval follows the IPR and WBS Preparation.  Its
              duration depends on RDTE dollars, Production dollars,
              and OSD/CAIG waiver (if needed).
      COMMENT: WBS Approval.  Source:  John Holdren, x52651.
              Event 20 in RW Phase I network, 30 Aug 85.


RULE:   RMOA
        PRIORITY:  80
        TEST:   E
        IF:     vmoa <>  "NONE" and KNOWN(tipr)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "  130"
                CURRTASK.DESCR = "MOA/MOU                     "
                CURRTASK.OPR = "PM & RWPP    "
                tmoa = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tipr
                CURRLINK.SUCC = tmoa
        REASON: MOA/MOU follows the IPR.  Its duration is 130 days.
        COMMENT: WBS Preparation.  Source:  LTC Hollingsworth, x54811.
              Event 21 in RW Phase I network, 30 Aug 85.


RULE:   RSAFETY
        PRIORITY:  80
        TEST:   E
        IF:     KNOWN(tipr)
        THEN:   PERFORM ADDTASK
                duration = 20
                IF vsafe = "YES" THEN
                  duration = duration + 10
                ENDIF
                IF vjoint = "NONE" THEN
                  duration = duration - 10
                ENDIF
                CURRTASK.DUR = TOSTR(duration,5,0)
                CURRTASK.DESCR = "SAFETY      REQUIREMENTS"
                CURRTASK.OPR = "RWS          "
                tsafe = CURRTASK.ID
```

```
            ATTACH 1 TO CURRLINK
            CURRLINK.PRED = tipr
            CURRLINK.SUCC = tsafe
        REASON: Safety Requirements follow the IPR.  Its duration
               depends on Operational Safety and Joint service
               involvement.
        COMMENT: Safety Requirements.  Source:  Mr. Bigi, x59249.
               Event 22 in RW Phase I network, 30 Aug 85.


RULE:   RSCHED
        PRIORITY:  80
        TEST:   E
        IF:     KNOWN(tipr)
        THEN:   PERFORM ADDTASK
            CURRTASK.DUR = "   30"
            CURRTASK.DESCR = "PROGRAM     SCHEDULES    "
            CURRTASK.OPR = "PM           "
            tsched = CURRTASK.ID
            ATTACH 1 TO CURRLINK
            CURRLINK.PRED = tipr
            CURRLINK.SUCC = tsched
        REASON: Program Schedules follow the IPR.  Its duration is 30
               days.
        COMMENT: Program Schedules.Source: LTC  Hollingsworth, x54811.
               Event 23 in RW Phase I network, 30 Aug 85.


RULE:   RDAP
        PRIORITY:  70
        TEST:   E
        IF:     vinscope = "NO" and
            ((KNOWN(tform56o) and vsssn <> "YES") or KNOWN(tsss))
        THEN:   PERFORM ADDTASK
            IF vrdte > 0 or (vprod*1000) > 100 THEN
              duration = "   20"
            ELSE
              duration = "    5"
            ENDIF
            CURRTASK.DUR = duration
            CURRTASK.DESCR = "DRAFT AP                "
            CURRTASK.OPR = "PM & RWK    "
            tdap = CURRTASK.ID
            IF KNOWN(tsss) THEN
              ATTACH 1 TO CURRLINK
              CURRLINK.PRED = tsss
              CURRLINK.SUCC = tdap
            ELSE
              ATTACH 1 TO CURRLINK
              CURRLINK.PRED = tform56o
              CURRLINK.SUCC = tdap
            ENDIF
```

```
        REASON: Develop Acquisition Plan follows the Sources Sought
                Synopsis (if used) or receipt of AFSC Form 56.  Its
                duration depends on the type and amount of dollars to be
                spent.  This task is not needed for follow-on efforts.
        COMMENT: Develop Acquisition Plan.  Source: Jim Shaeffer, 52336.
                Event 24 in RW Phase I network, 30 Aug 85.


RULE:   RASPP
        PRIORITY:  80
        TEST:   E
        IF:     vinscope = "NO" and KNOWN(tipr)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "   30"
                CURRTASK.DESCR = "ASP PREP                    "
                CURRTASK.OPR = "PM & RWK     "
                taspp = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tipr
                CURRLINK.SUCC = taspp
        REASON: Acquisition Strategy Panel Preparation follows the IPR.
                Its duration is 30 days.  This task is not needed for
                follow-on efforts.
        COMMENT: Acquisition Strategy Panel Preparation.  Source: Jim
                Shaeffer, x52336.
                Events 25 and 27 in RW Phase I network, 30 Aug 85.


RULE:   RASP
        PRIORITY:  70
        TEST:   E
        IF:     KNOWN(tcostest) and KNOWN(tsched) and KNOWN(tdap) and
                KNOWN(taspp)
        THEN:   PERFORM ADDTASK
                IF vrdte < 200 and vprod < 1000 THEN
                  duration = "    1"
                ELSE
                  duration = "    2"
                ENDIF
                CURRTASK.DUR = duration
                CURRTASK.DESCR = "ASP                         "
                CURRTASK.OPR = "PM & RWK     "
                tasp = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tcostest
                CURRLINK.SUCC = tasp
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tsched
                CURRLINK.SUCC = tasp
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tdap
                CURRLINK.SUCC = tasp
```

```
          ATTACH 1 TO CURRLINK
          CURRLINK.PRED = taspp
          CURRLINK.SUCC = tasp
REASON: The Acquisition Strategy Panel follows Acquisition
        Strategy Panel Preparation, the Post-IPR Cost Estimate,
        development of Program Schedules, and development of the
        Draft Acquisition Plan.  Its duration depends on the
        amount and type of dollars spent.  This task is not
        needed for follow-on efforts.
COMMENT: Acquisition Strategy Panel.  Source: Jim Shaeffer,
        x52336.
        Events 25 and 27 in RW Phase I network, 30 Aug 85.


RULE:   RDATA
        PRIORITY:  80
        TEST:   E
        IF:     vdata = "YES" and
                (KNOWN(tdcrisp) or vmccr <> "YES" or vpmrt <> "YES")
                and KNOWN(tdsow)
        THEN:   PERFORM ADDTASK
                IF vfollow = "NO" and vscomp = "MAJOR" THEN
                  duration = 71
                ELSE
                  duration = 62
                ENDIF
                IF vqrc = "YES" THEN
                  IF vscomp = "MAJOR" THEN
                    duration = 28
                  ELSE
                    duration = 21
                  ENDIF
                ENDIF
                IF vjoint <> "NONE" THEN
                  IF vscomp = "MAJOR" THEN
                    duration = duration + 21
                  ELSE
                    duration = duration + 14
                  ENDIF
                ENDIF
                CURRTASK.DUR = TOSTR(duration,5,0)
                CURRTASK.DESCR = "DATA PACKAGE PREPARATION"
                CURRTASK.OPR = "RWB          "
                tdata = CURRTASK.ID
                IF KNOWN(tdcrisp) THEN
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tdcrisp
                  CURRLINK.SUCC = tdata
                ENDIF
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tdsow
                CURRLINK.SUCC = tdata
```

```
        REASON: Data Package Preparation follows development of the
                CRISP (if used) and the Draft SOW.  Its duration depends
                on type of effort, System Complexity, and Joint service
                involvement.
        COMMENT: Data Package Preparation.  Source: Linda Lorenz,
                x56421.
                Event 28 in RW Phase I network, 30 Aug 85.


RULE:   RSSS
        PRIORITY:  80
        TEST:   E
        IF:     vsssn = "YES" and KNOWN(tdsow)
        THEN:   PERFORM ADDTASK
                duration = 42
                IF vcntrct = "SOLE SOURCE" THEN
                  duration = duration - 15
                ENDIF
                CURRTASK.DUR = TOSTR(duration,5,0)
                CURRTASK.DESCR = "SOURCES     SOUGHT SYNOP"
                CURRTASK.OPR = "PM & RWK     "
                tsss = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tdsow
                CURRLINK.SUCC = tsss
        REASON: The Sources Sought Synopsis follows the Draft SOW.  Its
                duration depends on the type of contract used.  This task
                is not needed unless the RFP exceeds $25K or RDTE money
                is involved.
        COMMENT: Sources Sought Synopsis.  Sources: Carolyn Bowling,
                x52085; Jim Shaeffer, x52336.
                Event 29 in RW Phase I network, 30 Aug 85.


RULE:   RSSPP
        PRIORITY:  70
        TEST:   E
        IF:     vfollow = "NO" and vcntrct <> "SOLE SOURCE" and
                KNOWN(tasp)
        THEN:   PERFORM ADDTASK
                IF vrdte > 100 or vprod > 500 THEN
                  duration = "   45"
                ELSE
                  IF vrdte > 50 or vprod > 100 THEN
                    duration = "   30"
                  ELSE
                    duration = "   15"
                  ENDIF
                ENDIF
                CURRTASK.DUR = duration
                CURRTASK.DESCR = "SS PLAN PREP            "
                CURRTASK.OPR = "PM          "
```

```
            tsspp = CURRTASK.ID
            ATTACH 1 TO CURRLINK
            CURRLINK.PRED = tasp
            CURRLINK.SUCC = tsspp
REASON:  Source Selection Plan Preparation follows the
         Acquisition Strategy Panel.  Its duration depends on the
         type and amount of dollars spent.  This task is not used
         for sole source procurements.
COMMENT: Source Selection Plan Preparation.  Source: Ed Martin,
         x56624.
         Event 30 in RW Phase I network, 30 Aug 85.


RULE:    RSSPA
         PRIORITY:  70
         TEST:   E
         IF:     KNOWN(tsspp)
         THEN:   PERFORM ADDTASK
                 IF vrdte > 100 or vprod > 500 THEN
                   duration = "   30"
                 ELSE
                   IF vrdte > 50 or vprod > 100 THEN
                     IF vssd = "YES" THEN
                       duration = "   15"
                     ELSE
                       duration = "   13"
                     ENDIF
                   ELSE
                     duration = "    5"
                   ENDIF
                 ENDIF
                 CURRTASK.DUR = duration
                 CURRTASK.DESCR = "SS PLAN     APPROVAL     "
                 CURRTASK.OPR = "PM           "
                 tsspa = CURRTASK.ID
                 ATTACH 1 TO CURRLINK
                 CURRLINK.PRED = tsspp
                 CURRLINK.SUCC = tsspa
         REASON: Source Selection Plan Approval follows Source Selection
                 Plan Preparation.  Its duration depends on the type and
                 amount of dollars spent.  This task is not used for sole
                 source procurements.
         COMMENT: Source Selection Plan Approval.  Source: Ed Martin,
                 x56624.
                 Event 30 in RW Phase I network, 30 Aug 85


RULE:    RJAP
         PRIORITY:  70
         TEST:   E
         IF:     vcntrct = "SOLE SOURCE" and vfollow = "NO" and
                 KNOWN(tasp)
```

```
THEN:   PERFORM ADDTASK
        CURRTASK.DUR = "    5"
        CURRTASK.DESCR = "J&A PREP                "
        CURRTASK.OPR = "RWK          "
        tjap = CURRTASK.ID
        ATTACH 1 TO CURRLINK
        CURRLINK.PRED = tasp
        CURRLINK.SUCC = tjap
REASON: Justification & Approval Preparation follows the
       Acquisition Strategy Panel.  Its duration is 5 days.
       This  task is not needed unless a sole source procurement
       is planned.
COMMENT: J&A Preparation.  Sources: Al Miller, x58328; Jim
       Shaeffer, x52336.
       Event 31 in RW Phase I network, 30 Aug 85.


RULE:   RPBASE
        PRIORITY:  70
        TEST:   E
        IF:     KNOWN(tasp) or (KNOWN(tsched) and vinscope = "YES")
        THEN:   PERFORM ADDTASK
                IF vjoint = "NONE" THEN
                  duration = "   65"
                ELSE
                  duration = "  130"
                ENDIF
                CURRTASK.DUR = duration
                CURRTASK.DESCR = "PROGRAM     BASELINE     "
                CURRTASK.OPR = "PM           "
                tpbase = CURRTASK.ID
                IF KNOWN(tasp) THEN
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tasp
                  CURRLINK.SUCC = tpbase
                ELSE
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tsched
                  CURRLINK.SUCC = tpbase
                ENDIF
        REASON: The Program Baseline follows the Acquisition Strategy
               Panel (if used) or Program Schedules.  Its duration
               depends on Joint service involvement.
        COMMENT: Program Baseline.  Sources: LTC  Hollingsworth, x54811.
               Event 33 in RW Phase I network, 30 Aug 85.


RULE:   RCBASE
        PRIORITY:  70
        TEST:   E
        IF:     KNOWN(tasp) or (KNOWN(tcostest) and vinscope = "YES")
        THEN:   PERFORM ADDTASK
```

```
            duration = 25
            IF vecbase = "YES" THEN
              duration = 15
            ELSE
              duration = 10
            ENDIF
            IF vpdes = "SAR" THEN
              duration = duration + 10
            ELSE
              IF vpdes = "DESIGNATED" THEN
                duration = duration + 7
              ELSE
                duration = duration + 5
              ENDIF
            ENDIF
            IF vjoint <> "NONE" THEN
              duration = duration + 5
            ENDIF
            CURRTASK.DUR = TOSTR(duration,5,0)
            CURRTASK.DESCR = "COST        BASELINE    "
            CURRTASK.OPR = "PM & RWPE   "
            tcbase = CURRTASK.ID
            IF KNOWN(tasp) THEN
              ATTACH 1 TO CURRLINK
              CURRLINK.PRED = tasp
              CURRLINK.SUCC = tcbase
            ELSE
              ATTACH 1 TO CURRLINK
              CURRLINK.PRED = tcostest
              CURRLINK.SUCC = tcbase
            ENDIF
      REASON: The Cost Baseline follows the Acquisition Strategy Panel
              (if used) or the Cost Estimate.  Its duration depends on
              Joint service involvement.
      COMMENT: Cost Baseline.  Sources: LT Karpowich, x54011.
              Event 34 in RW Phase I network, 30 Aug 85.


RULE:   RAPA
        PRIORITY:  70
        TEST:   E
        IF:     KNOWN(tasp)
        THEN:   PERFORM ADDTASK
                IF vrdte > 5 THEN
                  duration = "  124"
                ELSE
                  IF vprod > 5 THEN
                    duration = "    4"
                  ELSE
                    duration = "    0"
                  ENDIF
                ENDIF
```

```
            CURRTASK.DUR = duration
            CURRTASK.DESCR = "AP APPROVAL                "
            CURRTASK.OPR = "PM & RWK     "
            tapa = CURRTASK.ID
            ATTACH 1 TO CURRLINK
            CURRLINK.PRED = tasp
            CURRLINK.SUCC = tapa
        REASON: Acquisition Plan Approval follows the Acquisition
               Strategy Panel.  Its duration depends on the type and
               amount of dollars spent.
        COMMENT: Acquisition Plan Approval.  Sources: Jim Shaeffer,
               x52336.
               Event 35 in RW Phase I network, 30 Aug 85.


RULE:   RFSOW
        PRIORITY:  70
        TEST:   E
        IF:     (KNOWN(tdcrisp) or vmccr <> "YES" or vpmrt <> "YES")
                and (KNOWN(tapa) or vinscope = "YES") and
                KNOWN(tdilsp) and (KNOWN(tdrfp) or
                  (vdrfpn <> "YES" and (KNOWN(tsss) or
                    (vsssn <> "YES"and KNOWN(tdsow))))) and
                (KNOWN(twbsa) or (vrdte <= 2 and vprod <=2))
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "   40"
                CURRTASK.DESCR = "FINAL SOW                  "
                CURRTASK.OPR = "PM & RWE     "
                tfsow = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tdilsp
                CURRLINK.SUCC = tfsow
                IF KNOWN(tdcrisp) THEN
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tdcrisp
                  CURRLINK.SUCC = tfsow
                ENDIF
                IF KNOWN(tapa) THEN
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tapa
                  CURRLINK.SUCC = tfsow
                ENDIF
                IF KNOWN(tdrfp) THEN
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tdrfp
                  CURRLINK.SUCC = tfsow
                ELSE
                  IF KNOWN(tsss) THEN
                    ATTACH 1 TO CURRLINK
                    CURRLINK.PRED = tsss
                    CURRLINK.SUCC = tfsow
                  ELSE
```

```
            ATTACH 1 TO CURRLINK
            CURRLINK.PRED = tdsow
            CURRLINK.SUCC = tfsow
          ENDIF
        ENDIF
        IF KNOWN(twbsa) THEN
          ATTACH 1 TO CURRLINK
          CURRLINK.PRED = twbsa
          CURRLINK.SUCC = tfsow
        ENDIF
REASON: The Final Statement of Work follows the Draft CRISP (if
        used), Acquisition Plan Approval (if used), the Draft RFP
        (if used) and approval of the Work Breakdown Structure
        (if used), or the Draft Statement of Work.  Its duration
        is 40 days.
COMMENT: Final SOW.  Sources: RW Phase I Scheduling Guide.
        Event 36 in RW Phase I network, 30 Aug 85.


RULE:   RDRFP
        PRIORITY:  70
        TEST:   E
        IF:     vdrfpn = "YES" and KNOWN(tdspec) and KNOWN(tdsow) and
                (KNOWN(tdata) or vdata <> "YES") and
                (KNOWN(tasp) or (vinscope = "YES" and
                (KNOWN(tsss) or vsssn <> "YES")))
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "   17"
                CURRTASK.DESCR = "DRAFT RFP                "
                CURRTASK.OPR = "PM & RWK    "
                tdrfp = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tdspec
                CURRLINK.SUCC = tdrfp
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tdsow
                CURRLINK.SUCC = tdrfp
                IF KNOWN(tdata) THEN
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tdata
                  CURRLINK.SUCC = tdrfp
                ENDIF
                IF KNOWN(tasp) THEN
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tasp
                  CURRLINK.SUCC = tdrfp
                ELSE
                  IF KNOWN(tsss) THEN
                    ATTACH 1 TO CURRLINK
                    CURRLINK.PRED = tsss
                    CURRLINK.SUCC = tdrfp
                  ENDIF
```

```
        ENDIF
REASON: The Draft Request For Proposal follows the Draft
        Statement of Work, the Draft Specification, Data Package
        Preparation (if used) and the Acquistion Strategy Panel
        (if used). Its duration is 17 days including response
        time. This task is not used for sole source procurements
        or competitive procurements less than $25M.
COMMENT: Draft RFP. Sources: Carolyn Bowling, x52085; Jim
        Shaeffer, x52336.
        Event 37 in RW Phase I network, 30 Aug 85.


RULE:   RFSPEC
        PRIORITY: 80
        TEST:   E
        IF:     KNOWN(tdilsp) and
                (KNOWN(tdcrisp) or vmccr <> "YES" or vpmrt <> "YES") and
                ((KNOWN(tdspec) and vdrfpn <> "YES") or KNOWN(tdrfp))
        THEN:   PERFORM ADDTASK
                duration = 150
                IF vscomp = "MINOR" THEN
                  duration = duration - 15
                ENDIF
                IF vjoint = "NONE" THEN
                  duration = duration - 25
                ENDIF
                CURRTASK.DUR = TOSTR(duration,5,0)
                CURRTASK.DESCR = "FINAL SPEC                "
                CURRTASK.OPR = "PM & RWE     "
                tfspec = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tdilsp
                CURRLINK.SUCC = tfspec
                IF KNOWN(tdcrisp) THEN
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tdcrisp
                  CURRLINK.SUCC = tfspec
                ENDIF
                IF KNOWN(tdrfp) THEN
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tdrfp
                  CURRLINK.SUCC = tfspec
                ELSE
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tdspec
                  CURRLINK.SUCC = tfspec
                ENDIF
        REASON: The Final Specification follows the draft ILSP the draft
                CRISP (if used), and the Draft RFP (if used) or the Draft
                Specification. Its duration depends on System Complexity
                and Joint service involvement.
        COMMENT: Final Specification. Sources: RW Phase I Scheduling
```

```
            Guide.
            Event 38 in RW Phase I network, 30 Aug 85.

RULE:   RJAA
        PRIORITY:  70
        TEST:   E
        IF:     KNOWN(tjap) and KNOWN(tapa)
        THEN:   PERFORM ADDTASK
                duration = 20
                IF vrdte > 10 or vprod > 10 THEN
                  IF vcntrv = "YES" THEN
                    duration = duration + 90
                  ELSE
                    duration = duration + 60
                  ENDIF
                ELSE
                  duration = duration + 30
                ENDIF
                CURRTASK.DUR = TOSTR(duration,5,0)
                CURRTASK.DESCR = "J&A APPROVAL              "
                CURRTASK.OPR = "RWK          "
                tjaa = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tjap
                CURRLINK.SUCC = tjaa
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tapa
                CURRLINK.SUCC = tjaa
        REASON: Justification & Approval Approval follows Justification
               & Approval Preparation and Acquisition Plan Approval.
               Its duration depends on the amount of dollars spent.
        COMMENT: J&A Preparation.  Sources: Al Miller, x58328; Jim
               Shaeffer, x52336.
               Event 39 in RW Phase I network, 30 Aug 85.

RULE:   RFRM117
        PRIORITY:  70
        TEST:   E
        IF:     (KNOWN(tdd254) or vclass = "UNCLASSIFIED") and
                KNOWN(ttemp) and (KNOWN(tmoa) or vmoa = "NONE") and
                KNOWN(tsafe) and KNOWN(tpbase) and KNOWN(tcbase) and
                KNOWN(tfsow) and KNOWN(tpad) and KNOWN(tpmp) and
                KNOWN(tfspec) and (KNOWN(tjaa) or
                  vcntrct <> "SOLE SOURCE" or vfollow = "YES") and
                    (vdrfpn = "YES" or KNOWN(tdata) or vdata <> "YES")
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "   15"
                CURRTASK.DESCR = "ASD FORM 117              "
                CURRTASK.OPR = "PM           "
                tform117 = CURRTASK.ID
```

```
        IF KNOWN(tdd254) THEN
          ATTACH 1 TO CURRLINK
          CURRLINK.PRED = tdd254
          CURRLINK.SUCC = tform117
        ENDIF
        ATTACH 1 TO CURRLINK
        CURRLINK.PRED = ttemp
        CURRLINK.SUCC = tform117
        IF KNOWN(tmoa) THEN
          ATTACH 1 TO CURRLINK
          CURRLINK.PRED = tmoa
          CURRLINK.SUCC = tform117
        ENDIF
        ATTACH 1 TO CURRLINK
        CURRLINK.PRED = tsafe
        CURRLINK.SUCC = tform117
        ATTACH 1 TO CURRLINK
        CURRLINK.PRED = tpbase
        CURRLINK.SUCC = tform117
        ATTACH 1 TO CURRLINK
        CURRLINK.PRED = tcbase
        CURRLINK.SUCC = tform117
        ATTACH 1 TO CURRLINK
        CURRLINK.PRED = tfsow
        CURRLINK.SUCC = tform117
        ATTACH 1 TO CURRLINK
        CURRLINK.PRED = tpad
        CURRLINK.SUCC = tform117
        ATTACH 1 TO CURRLINK
        CURRLINK.PRED = tpmp
        CURRLINK.SUCC = tform117
        ATTACH 1 TO CURRLINK
        CURRLINK.PRED = tfspec
        CURRLINK.SUCC = tform117
        IF KNOWN(tjaa) THEN
          ATTACH 1 TO CURRLINK
          CURRLINK.PRED = tjaa
          CURRLINK.SUCC = tform117
        ENDIF
        IF vdrfpn <> "YES" and vdata = "YES" THEN
          ATTACH 1 TO CURRLINK
          CURRLINK.PRED = tdata
          CURRLINK.SUCC = tform117
        ENDIF
REASON: ASD Form 117 coordination and approval follows DD254 (if
      used), TEMP, MOA/MOU (if used), Safety, ASP (if used),
      Data Package (if used), Program Baseline, Cost Baseline,
      Final SOW, Draft RFP (if used), and Final Specification.
      Its duration is 15 days.
COMMENT: ASD Form 117.  Sources: LTC  Hollingsworth, x54811.
      Event 40 in RW Phase I network, 30 Aug 85.
```

```
RULE:   RRFPPP
        PRIORITY:  70
        TEST:   E
        IF:     KNOWN(tform117)
        THEN:   PERFORM ADDTASK
                CURRTASK.DUR = "   14"
                CURRTASK.DESCR = "RFP PACKAGE PREP        "
                CURRTASK.OPR = "RWK          "
                trfppp = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tform117
                CURRLINK.SUCC = trfppp
        REASON: RFP Package Preparation follows ASD Form 117
               coordination and approval.  Its duration is 14 days.
        COMMENT: RFP Package Preparation.  Sources: Jim Shaeffer,
               x52336.
               Event 41 in RW Phase I network, 30 Aug 85.


RULE:   RSSSP
        PRIORITY:  70
        TEST:   E
        IF:     vfollow = "NO" and vcntrct <> "SOLE SOURCE" and
                KNOWN(tasp)
        THEN:   PERFORM ADDTASK
                IF vrdte > 100 or vprod > 500 THEN
                  duration = "   45"
                ELSE
                  IF vrdte > 50 or vprod > 100 THEN
                    duration = "   30"
                  ELSE
                    duration = "   15"
                  ENDIF
                ENDIF
                CURRTASK.DUR = duration
                CURRTASK.DESCR = "SS STANDARDS PREP        "
                CURRTASK.OPR = "PM           "
                tsssp = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tasp
                CURRLINK.SUCC = tsssp
        REASON: Source Selection Standards Preparation follows the
               Acquisition Strategy Panel.  Its duration depends on the
               type and amount of dollars spent.  This task is not used
               for sole source procurements.
        COMMENT: Source Selection Standards Preparation.  Source: Ed
               Martin, x56624.
               Event 43 in RW Phase I network, 30 Aug 85.
```

```
RULE:    RSSSA
         PRIORITY:  70
         TEST:    E
         IF:      KNOWN(tsssp) and KNOWN(tsspa)
         THEN:    PERFORM ADDTASK
                  IF vrdte > 100 or vprod > 500 THEN
                    duration = "   30"
                  ELSE
                    IF vrdte > 50 or vprod > 100 THEN
                      IF vssd = "YES" THEN
                        duration = "   15"
                      ELSE
                        duration = "   13"
                      ENDIF
                    ELSE
                      duration = "    5"
                    ENDIF
                  ENDIF
                  CURRTASK.DUR = duration
                  CURRTASK.DESCR = "SS STANDARDS APPROVAL    "
                  CURRTASK.OPR = "PM          "
                  tsssa = CURRTASK.ID
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tsssp
                  CURRLINK.SUCC = tsssa
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tsspa
                  CURRLINK.SUCC = tsssa
         REASON: Source Selection Standards Approval follows Source
                 Selection Standards Preparation and Source Selection
                 Plan Approval.  Its duration depends on the type and
                 amount of dollars spent.  This task is not used for sole
                 source procurements.
         COMMENT: Source Selection Plan Approval.  Source: Ed Martin,
                 x56624.
                 Event 43 in RW Phase I network, 30 Aug 85.


RULE:    RSSP
         PRIORITY:  70
         TEST:    E
         IF:      KNOWN(tsspp)
         THEN:    PERFORM ADDTASK
                  duration = "   10"
                  IF vrdte > 50 or vprod > 100 THEN
                    IF vssd <> "YES" THEN
                      duration = "   20"
                    ENDIF
                  ENDIF
                  CURRTASK.DUR = duration
                  CURRTASK.DESCR = "SS          PROCEDURES   "
```

```
            CURRTASK.OPR = "PM          "
            tssp = CURRTASK.ID
            ATTACH 1 TO CURRLINK
            CURRLINK.PRED = tsspp
            CURRLINK.SUCC = tssp
REASON:  Source Selection Procedures follows Source Selection
         Plan Preparation.  Its duration depends on the type and
         amount of dollars spent.  This task is not used for sole
         source procurements.
COMMENT: Source Selection Plan Approval.  Source: Ed Martin,
         x56624.
         Event 43 in RW Phase I network, 30 Aug 85.


RULE:    RASDREV
         PRIORITY:  70
         TEST:    E
         IF:      KNOWN(trfppp)
         THEN:    PERFORM ADDTASK
                  duration = "   15"
                  IF (vrfp/1000) > 25 THEN
                    duration = "   20"
                  ENDIF
                  CURRTASK.DUR = duration
                  CURRTASK.DESCR = "3-LTR & ASD REVIEWS     "
                  CURRTASK.OPR = "PM          "
                  tasdrev = CURRTASK.ID
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = trfppp
                  CURRLINK.SUCC = tasdrev
         REASON:  Three-letter and ASD Reviews follow RFP Package
                  Preparation.  Its duration depends on the RFP value.
         COMMENT: Three-letter and ASD Reviews.  Source: Jim Shaeffer,
                  x52336.
                  Event 44 in RW Phase I network, 30 Aug 85.


RULE:    RAFSCREV
         PRIORITY:  70
         TEST:    E
         IF:      KNOWN(tasdrev) and (vrfp/1000) > 75
         THEN:    PERFORM ADDTASK
                  CURRTASK.DUR = "   25"
                  CURRTASK.DESCR = "AFSC REVIEW             "
                  CURRTASK.OPR = "PM          "
                  tafscrev = CURRTASK.ID
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tasdrev
                  CURRLINK.SUCC = tafscrev
         REASON:  The AFSC Review follows the Three-letter and ASD
                  Reviews.  Its duration is 25 days.  This task is not
                  needed unless the RFP Package exceeds $75M.
```

```
        COMMENT: AFSC Review.  Source: Jim Shaeffer, x52336.
                 Event 44 in RW Phase I network, 30 Aug 85.

RULE:   RRFPR
        PRIORITY:  70
        TEST:   E
        IF:     (KNOWN(tsspa) or vfollow = "YES" or
                  vcntrct = "SOLE SOURCE" or vinscope = "YES") and
                KNOWN(tasdrev)
        THEN:   PERFORM ADDTASK
                IF vcntrct = "COMPETITIVE" THEN
                  IF (vrfp/1000) < 3.5 THEN
                    duration = "   35"
                  ELSE
                    duration = "   45"
                  ENDIF
                ENDIF
                IF vcntrct = "SOLE SOURCE" THEN
                  IF (vrfp/1000) < 25 THEN
                    duration = "   35"
                  ELSE
                    duration = "   45"
                  ENDIF
                ENDIF
                IF vcntrct = "LONG LEAD" or \
                        vcntrct = "LETTER CONTRACT" or \
                        vcntrct = "UNPRICE ORDER UNDER BOA" THEN
                   duration = "   30"
                ENDIF
                IF vcntrct = "PRICED ORDER UNDER BOA" THEN
                  duration = "   45"
                ENDIF
                IF vcntrct = "ONE STEP SEALED BID" THEN
                  duration = "   60"
                ENDIF
                IF vcntrct = "TWO STEP SEALED BID" THEN
                  duration = "   45"
                ENDIF
                CURRTASK.DUR = duration
                CURRTASK.DESCR = "RFP RELEASE              "
                CURRTASK.OPR = "RWK          "
                trfpr = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = tasdrev
                CURRLINK.SUCC = trfpr
                IF KNOWN(tsspa) THEN
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tsspa
                  CURRLINK.SUCC = trfpr
                ENDIF
        REASON: RFP Release follows the Three-letter and ASD Reviews and
```

```
        SS Plan Approval (if used).  Its duration depends on the
        Type of Contract and the RFP value.
COMMENT: RFP Release.  Sources: Jim Shaeffer, x52336, Jim
        Chapman, x?????, and Myron Phillips, x?????.
        Event 45 in RW Phase I network, 30 Aug 85.


RULE:   RCPROPR
        PRIORITY:  70
        TEST:   E
        IF:     KNOWN(trfpr)
        THEN:   PERFORM ADDTASK
                IF vcntrct = "COMPETITIVE" THEN
                  IF (vrfp/1000) < 3.5 THEN
                    duration = "   50"
                  ELSE
                    duration = "   60"
                  ENDIF
                ENDIF
                IF vcntrct = "SOLE SOURCE" THEN
                  IF (vrfp/1000) < 25 THEN
                    duration = "   50"
                  ELSE
                    duration = "   60"
                  ENDIF
                ENDIF
                IF vcntrct = "LONG LEAD" or \
                        vcntrct = "LETTER CONTRACT" THEN
                   duration = "   50"
                ENDIF
                IF vcntrct = "UNPRICE ORDER UNDER BOA" THEN
                   duration = "   30"
                ENDIF
                IF vcntrct = "PRICED ORDER UNDER BOA" THEN
                  duration = "   60"
                ENDIF
                IF vcntrct = "ONE STEP SEALED BID" THEN
                  duration = "   45"
                ENDIF
                IF vcntrct = "TWO STEP SEALED BID" THEN
                  duration = "   60"
                ENDIF
                CURRTASK.DUR = duration
                CURRTASK.DESCR = "CONTRACTOR  RESPONSE    "
                CURRTASK.OPR = "CONTRACTOR  "
                tcpropr = CURRTASK.ID
                ATTACH 1 TO CURRLINK
                CURRLINK.PRED = trfpr
                CURRLINK.SUCC = tcpropr
REASON: Contractor Response follows the RFP Release.  Its
        duration depends on the Type of Contract and the RFP
        value.
```

```
COMMENT: Contractor Response.  Sources: Jim Shaeffer, x52336,
       Jim Chapman, x?????, and Myron Phillips, x?????.
       Event 46 in RW Phase I network, 30 Aug 85.


RULE:   RSS
        PRIORITY:  70
        TEST:   E
        IF:     (KNOWN(tafscrev) or (vrfp/1000) <= 75) and
                KNOWN(tcpropr) and ((KNOWN(tsssa) and KNOWN(tssp)) or
                vcntrct = "SOLE SOURCE" or vfollow = "YES" or
                vinscope = "YES")
        THEN:   PERFORM ADDTASK
                IF vcntrct = "COMPETITIVE" THEN
                  IF (vrfp/1000) < 3.5 THEN
                    duration = "   37"
                  ELSE
                    IF (vrfp/1000) < 25 THEN
                      duration = "   65"
                    ELSE
                      duration = "   67"
                    ENDIF
                  ENDIF
                ENDIF
                IF vcntrct = "SOLE SOURCE" THEN
                  IF (vrfp/1000) < 25 THEN
                    duration = "   68"
                  ELSE
                    duration = "   80"
                  ENDIF
                ENDIF
                IF vcntrct = "LONG LEAD" or \
                       vcntrct = "LETTER CONTRACT" or \
                       vcntrct = "UNPRICE ORDER UNDER BOA" THEN
                  duration = "   68"
                ENDIF
                IF vcntrct = "PRICED ORDER UNDER BOA" THEN
                  duration = "   95"
                ENDIF
                CURRTASK.DUR = duration
                CURRTASK.DESCR = "SOURCE      SELECTION   "
                CURRTASK.OPR = "PM          "
                tss = CURRTASK.ID
                IF KNOWN(tafscrev) THEN
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tafscrev
                  CURRLINK.SUCC = tss
                ENDIF
                IF KNOWN(tsssa) THEN
                  ATTACH 1 TO CURRLINK
                  CURRLINK.PRED = tsssa
                  CURRLINK.SUCC = tss
```

```
            ENDIF
            IF KNOWN(tssp) THEN
              ATTACH 1 TO CURRLINK
              CURRLINK.PRED = tssp
              CURRLINK.SUCC = tss
            ENDIF
            ATTACH 1 TO CURRLINK
            CURRLINK.PRED = tcpropr
            CURRLINK.SUCC = tss
      REASON: Source Selection follows the AFSC Review (if used),
              Source Selection Standards Approval (if used), Source
              Selection Procedures (if used), and Contractor Response.
              Its duration depends on the Type of Contract and the RFP
              value.
      COMMENT: Source Selection.  Sources: Ed Martin, x56623; Jim
              Shaeffer, x52336; Jim Chapman, x?????; and Myron
              Phillips, x?????.
              Event 47 in RW Phase I network, 30 Aug 85.


RULE:   RAWARD
        PRIORITY:  70
        TEST:   E
        IF:     KNOWN(tcilsp) and KNOWN(tss) and
                (KNOWN(tccrisp) or vmccr <> "YES" or vpmrt <> "YES")
        THEN:   PERFORM ADDTASK
                IF vcntrct = "COMPETITIVE" THEN
                  IF (vrfp/1000) < 3.5 THEN
                    duration = "   32"
                  ELSE
                    duration = "   43"
                  ENDIF
                ENDIF
                IF vcntrct = "SOLE SOURCE" THEN
                  IF (vrfp/1000) < 25 THEN
                    duration = "   27"
                  ELSE
                    duration = "   35"
                  ENDIF
                ENDIF
                IF vcntrct = "LONG LEAD" or \
                        vcntrct = "LETTER CONTRACT" or \
                        vcntrct = "UNPRICE ORDER UNDER BOA" THEN
                   duration = "   27"
                ENDIF
                IF vcntrct = "PRICED ORDER UNDER BOA" THEN
                  duration = "   35"
                ENDIF
                IF vcntrct = "ONE STEP SEALED BID" THEN
                  duration = "   54"
                ENDIF
                IF vcntrct = "TWO STEP SEALED BID" THEN
```

```
    duration = "   64"
  ENDIF
  CURRTASK.DUR = duration
  CURRTASK.DESCR = "CONTRACT    AWARD       "
  CURRTASK.OPR = "RWK         "
  IF vaward <> "      " THEN
    CURRTASK.US = vaward
  ENDIF
  taward = CURRTASK.ID
  ATTACH 1 TO CURRLINK
  CURRLINK.PRED = tss
  CURRLINK.SUCC = taward
  ATTACH 1 TO CURRLINK
  CURRLINK.PRED = tcilsp
  CURRLINK.SUCC = taward
  IF KNOWN(tccrisp) THEN
    ATTACH 1 TO CURRLINK
    CURRLINK.PRED = tccrisp
    CURRLINK.SUCC = taward
  ENDIF
```

REASON: Contract Award follows Source Selection (if used) or Contractor Response. Coordination of the CRISP and ILSP are included for network completeness. However, they extend into the next stage. Its duration depends on the Type of Contract and the RFP value.

COMMENT: Contract Award. Sources: Jim Shaeffer, x52336, Jim Chapman, x?????, and Myron Phillips, x?????. Event 48 in RW Phase I network, 30 Aug 85.

Appendix E: ISA Evaluation Questionnaire

ISA EVALUATION

1. To what degree do you believe that the concept of an intelligent scheduling assistant as demonstrated by ISA is valid?

   a. Strongly agree: recommend developing operational system
   b. Agree: recommend further research
   c. Neutral
   d. Disagree: little merit
   e. Strongly disagree: abandon concept due to poor results

2. How would you evaluate the user interface?

   a. Excellent potential: extremely user-friendly
   b. Good potential: user-friendly
   c. Neutral
   d. Poor potential: not user-friendly
   e. Bad potential: would not be used

3. How would you evaluate the results of the system?

   a. Excellent: schedules are as good as best people do
   b. Good: schedules are reasonable
   c. Neutral
   d. Poor: schedules have deficiencies
   e. Bad: schedules are not usable

4. To what degree do you believe that an intelligent schedule assistant is an improvement to model networks?

   a. Strongly agree: much better
   b. Agree: better
   c. Neutral
   d. Disagree: worse
   e. Strongly disagree: much worse

5. What improvements would you recommend for the intelligent model network concept.

6. Have you ever used network scheduling before?

   a. More than 3 times
   b. One to three times
   c. Never

7. Have you ever used automated project management tools (CSNAS, Timeline, etc.)? If yes, please list project management tools that you have used:

8. Have you ever used model networks (RW Phase I, POINTS, CSNAS)? If yes, please list the model networks, number of times used, and results of use:

9. Do you believe that network scheduling has merit (yes/no)?

10. What further research in this area would you support (select all that apply):

   a. Development of a scheduler which generates networks using strictly a list of tasks, requirements and relationships.

   b. Development of an analyst which makes recommendations for modifying networks to solve resource and timing problems.

   c. Development of an assistant which automatically modifies networks at the users directions (determines how to reorder the network when tasks are added or relationships are modified).

   d. Please list any other you can think of.

Appendix F: AFALC Letter

DEPARTMENT OF THE AIR FORCE
HEADQUARTERS AIR FORCE ACQUISITION LOGISTICS CENTER
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-5000

FROM: AFALC/LSL

SUBJECT: DEMONSTRATION OF AI

TO: CAPTAIN MORAN

1. We were very impressed by the demonstration of your artificial intelligence/expert system for tailoring a model network. We had envisioned commissioning a pilot system, but your effort saved us both the cost and the effort of contracting for a pilot system.

2. This office created a series of model networks/schedules for the integration of acquisition logistics into the development and production of weapons systems. For several years our problem has been that there were not enough experts to help all of the acquisition logistics managers and the managers were not sufficiently expert in all of the separate integrated logistics support (ILS) areas to tailor their own networks. Your system demonstrates how the experts knowledge in an area can be captured in software to make their knowledge available to all of the managers.

3. As a result of your demonstrations, granted us from your own time, this office has now started working with AFLC/MM-AI toward the possible expansion of your system into the entire acquisition logistics arena. If our effort is as successful as your demonstrated system, the resultant software will eventually be deployed to all AFLC air logistics centers and AFSC product divisions for use by all acquisition logistics managers.

4. We thank you for the extra time you have dedicated to us in providing demonstrations of your system. We especially thank you for providing us with a pilot system that demonstrates how AI can be used to improve logistics within the weapons system program offices.

Albert L. Clark

ALBERT L. CLARK
CSNAS PROGRAM MANAGER

Bibliography

AFALC. CSNAS User Guide. Air Force Acquisition Logistic Command, Wright-Patterson AFB, OH, 15 July 1987.

ASD/AE. POINTS Schedule and Planning Handbook. Deputy for Aeronautical Equipment, Wright-Patterson AFB, OH, 15 April 1987.

Clark, Albert L. CSNAS Training Course. Air Force Acquisition Logistic Command, Wright-Patterson AFB, OH, 11-12 August 1987.

Fersko-Weiss, Henry. "Master Plans: Project Management Software," PC Magazine, 6/16: 153-209 (September 29, 1987).

Fox, B. R. and K. G. Kempf. "Complexity, Uncertainty and Opportunistic Scheduling," Proceedings of the Second Conference on Artificial Intelligence Applications. 487-492. Washington, D.C.: IEEE Computer Society Press, 1985.

Fox, Mark S. Constraint-Directed Search: A Case Study of Job-Shop Scheduling. PhD dissertation. Carnegie-Mellon University, Pittsburg, PA, December 1983, (AD-A138-307).

Fukumori, Koji. Fundamental Scheme for Train Scheduling (Application of Range-Constriction Search). Contract N00014-75-C-0643. Report AI-M-596. Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA, September 1980 (AD-A095-522).

Gansler, Jacques S. "Defense program instability: causes, costs and cures," Defense Management Journal, 22: 3-11 (Second Quarter 1986).

Gazdik, Igor. "Fuzzy-Network Planning - FNET," IEEE Transactions on Reliability, R-32,3:304-313 (August 1983).

Hayes-Roth, Barbara and others. PLANNER'S WORKBENCH: A Computer Aid to Re-planning. Report RAND/P-6688. Rand Corporation, Santa Monica, CA, October 1981 (AD-A113-331).

Hayes-Roth, Barbara and others. Modeling Planning as an Incremental, Opportunistic Process. Contract N00014-78-C-0039. Report RAND/N-1178-ONR. Rand Corporation, Santa Monica, CA, June 1979 (AD-A071-035).

MDBS. GURU Reference Manual Volume 1. Micro Data Base Management Systems, Lafayette, Indiana, May 1987.

Newman, P. A. and K. G. Kempf. "Opportunistic Scheduling for Robotic Machine Tending," IEEE 1985 Second Conference on Artificial Intelligence Applications. 168-175. Washington, D.C.: IEEE Computer Society Press, 1985.

Prerau, David S. "Selection of an Appropriate Domain for an Expert System," The AI Magazine, VI/2:26-30 (Summer 1985).

Sacerdoti, Earl D. "The Nonlinear Nature of Plans," Advance Papers of the Fourth International Joint Conference on Artificial Intelligence. 206-214. Los Altos, CA: William Kaufmann, Inc. September 1975.

Sage, Andrew P. and Chelsea C. White, III. "ADRIANE: A Knowledge-Based Interactive System for Planning and Decision Support," IEEE Transactions on Systems, Man, and Cybernetics. SMC-14,1:35-47 (January/February 1984).

Scanlon, Kathryn M. and Harold J. Schutt. The Program Manager's Support System (PMSS) An Executive Overview and Systems Description. Defense Systems Management College, Fort Belvoir, VA, 1 January 1987.

Tate, A., "Generating Project Networks," Proceedings of the 5th International Conference on Artificial Intelligence. 888-893. Cambridge, MA: IEEE Press, August 1977.

Vere, Steven A. "Planning in Time: Windows and Durations for Activities and Goals," IEEE Transactions on Pattern Analysis and Machine-Intelligence, PAMI-5,3:246-267 (May 1983).

Waterman, Donald A. A Guide to Expert Systems. Reading, MA: Addison-Wesley Publishing Company, 1986.

Wilkins, David E. and Ann E. Robinson. An Interactive Planning System. Contract F49620-79-C-0188, N00014-80-C-0300. Report TN-245. Artificial Intelligence Center, SRI International, Menlo Park, CA, July 1981 (AD-A103-019).

Winston, Patrick Henry. Artificial Intelligence. Reading, MA: Addison-Wesley, 1977.

Woffinden, MAJ Duard Stephen. EENG 543, Software Project Management. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, November 1987.

-----. Interactive Environment for a Computer-Aided Design System. MS Thesis, Naval Postgraduate School, Monterey, California, June 1984.

Zornes, Scotty E. Program Analyst. Personal interview. Aeronautical Systems Division, Air Force Systems Command, Wright-Patterson AFB, OH, 25 August 1987.

VITA

Captain Jerry L. Moran was born on 8 July 1957 in Manhattan, Kansas. He graduated from Manhattan High School in 1975. He received the degree of Bachelor of Science in Electrical Engineering from Kansas State University in May 1979. A recipient of the George C. Marshall award and Top 5 Percent Fellowship award, he was commissioned in the Regular Army in May 1979 through the ROTC program. He served three years as a platoon leader, company executive officer, and battalion intelligence officer with the 12th Engineer Battalion in West Germany. On 21 March 1983, he was transferred from the Corps of Engineers to the Signal Corps. Following attendance at the Tele-processing Operations Officer Course II at the Air Force Institute of Technology (AFIT), he was assigned to the United States Army Electronic Proving Ground (USAEPG), Fort Huachuca, Arizona. Besides serving as a Software and Evaluation Test Officer and Company Commander, he helped establish an artificial intelligence capability at USAEPG. Captain Moran entered the School of Engineering, AFIT, in September 1986.

Permanent Address: 514 N 5th
Manhattan, Kansas 66502

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/GE/ENG/88M-9	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION School of Engineering 6b. OFFICE SYMBOL (If applicable) AFIT/ENG	7a. NAME OF MONITORING ORGANIZATION
6c. ADDRESS (City, State, and ZIP Code) Air Force Institute of Technology (AU) Wright-Patterson AFB, OH 45433-6583	7b. ADDRESS (City, State, and ZIP Code)
8a. NAME OF FUNDING/SPONSORING ORGANIZATION See Box 19 8b. OFFICE SYMBOL (If applicable) See Box 19	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code) Wright-Patterson AFB, OH 45433-6503	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)

ISA: A PROTOTYPE INTELLIGENT SCHEDULING ASSISTANT FOR DEFENSE ACQUISITION MANAGEMENT

12. PERSONAL AUTHOR(S)
Jerry L. Moran, Captain, USA

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
MS Thesis	FROM _____ TO _____	March 1988	136

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Artificial Intelligence, Expert Systems, Scheduling, Project Management, Defense Acquisition Program Management
12	09		
05	01		

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

8a. NAME OF FUNDING/SPONSORING ORGANIZATION — 8b. OFFICE SYMBOL

Director of Program Control, Deputy for Reconnaissance/Strike & Electronic Warfare Systems, System Program Office — ASD/RWP

Director of Program Control, Comptroller — ASD/ACP

Thesis Chairman: Wade H. Shaw Jr., Captain, USA
Assistant Professor of Electrical Engineering

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
☐ UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS	UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE (Include Area Code)	22c. OFFICE SYMBOL
Wade H. Shaw Jr., Captain, USA	(513)-255-3576	AFIT/ENG

DD Form 1473, JUN 86 — Previous editions are obsolete. — SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

Block 19. ABSTRACT

This thesis investigated the application of artificial intelligence to defense acquisition management. The goal was to demonstrate how knowledge-based methods could improve scheduling of defense acquisition programs. The objectives were 1) to determine the kind of knowledge needed to tailor schedules; and 2) to develop a framework for using that knowledge to generate tailored schedules.

Scheduling of defense acquisition programs is a difficult problem for which expert systems are an appropriate solution methodology. This thesis identified 35 characteristics of a defense acquisition program which affect the applicability, duration, and relationships of tasks required to go from receipt of a Program Management Directive to contract award. It extends the model network concept used in the Aeronautical Systems Division and the Air Force Acquisition Logistic Command of the United States Air Force.

ISA, a prototype Intelligent Scheduling Assistant, successfully shows how knowledge used to tailor program schedules can be captured in a rule-based system. ISA uses the values of acquisition program characteristics to generate tailored schedules. The concept is applicable to any project schedule.

END

DATE

FILMED

DTIC

4/88