AFIT/GST/ENS/89J-3

ON-SITE ADAPTIVE DESIGN OF
A DECISION SUPPORT SYSTEM FOR
FIGHTER SQUADRON SCHEDULING

THESIS

James H. Jeter
Captain, USAF

AFIT/GST/ENS/89J-3

AFIT/GST/ENS/89J-3

ON-SITE ADAPTIVE DESIGN OF A DECISION SUPPORT SYSTEM FOR FIGHTER SQUADRON SCHEDULING

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 Operations Research

James H. Jeter, B.S.

Captain, USAF

June 89

Approved for public release; distribution unlimited

Preface

The purpose of this research was to develop a computer-based scheduling aid that worked. My goal was to field a functional decision support system that could be used on a daily basis by a crew-dog scheduler in a tactical fighter squadron. After several iterations, the Rhino Scheduling System addresses a significant portion of the scheduling decision process in the 89th TFS, Wright-Patterson AFB, Ohio.

This project was completed with the help, advice, and cooperation of a number of people. I want like to thank LtCol Skip Valusek, Maj Bruce Morlan, and Capt Will Bralick for their guidance and support while serving on my thesis committee. Capt Terry Tullia was a great help with the software. I would also like to thank Capt Bill (Centerline) Whittaker of the 89th TFS "Rhinos". Without his cooperation and input this project could never have been completed. And last, but most certainly not least, I want to thank my wife Debbie for her support and understanding during this difficult undertaking.

Table of Contents

Page

List of Figures

Abstract

The objective of this ~~research project~~ was to develop a decision support system (DSS) for the scheduler in a USAF tactical fighter squadron. A DSS supports the cognitive or mental processes of judgement and choice. The decision process supported in this project is the planning and programming of annual flying hours into monthly, weekly, and daily schedules. A kernel (or base) system was established, then extended and expanded with multiple iterations of the adaptive design approach. The adaptive design approach allowed the user to set system requirements in a gradual and interactive manner.

This thesis used as its subject the 89th Tactical Fighter Squadron located on Wright-Patterson AFB, OH. The close proximity of the 89th TFS and the Air Force Institute of Technology removed a common barrier to effective adaptive design: separation of the user and the builder. Through direct and frequent interaction, the 89th scheduler and the builder were able to develop and evolve a system that met the requirements of the user.

ON-SITE ADAPTIVE DESIGN OF A DECISION SUPPORT SYSTEM FOR FIGHTER SQUADRON SCHEDULING

I. Background

Fighter Squadron Scheduling

Computer support for the squadron scheduling function in a USAF fighter squadron is not a new concept. As computers became functional tools, their data handling capabilities were used to track and display many of the dates and qualifications important to aircrew scheduling. This electronic data management was a great improvement over the manual tracking of aircrew accomplishments, which had previously been posted on plexiglas greaseboards. Data processing did little, however, to improve the scheduling decision process itself.

Scheduling is a decision process in which resources, people, and events are sequenced into a day's activities in a logical and efficient way. The process begins at the start of the fiscal year when the squadron receives its allocation of flying hours. It culminates at the end of every operational day when the squadron safely lands its last scheduled sortie. The scheduling process is one of continual refinement and can be decomposed into a number of phases. These phases are:

1. Yearly Scheduling
2. Monthly Scheduling
3. Weekly Scheduling
4. Daily Scheduling

Yearly Scheduling

The first phase of the scheduling process begins when the squadron receives it's allocation of flying hours for the next fiscal year. Figure 1 illustrates important aspects of the yearly scheduling process.

The scheduler must decide how to distribute the allocated flying hours throughout the year while considering all known long range planning factors. Some of the factors to consider are weather conditions, deployments, exercises, manning, and host-country holidays. The goals and guidelines set by the wing and squadron leadership also enter into the budgeting process.

The result of this phase of the scheduling decision process is generally called the annual flying hour program. It is a result of the experience and

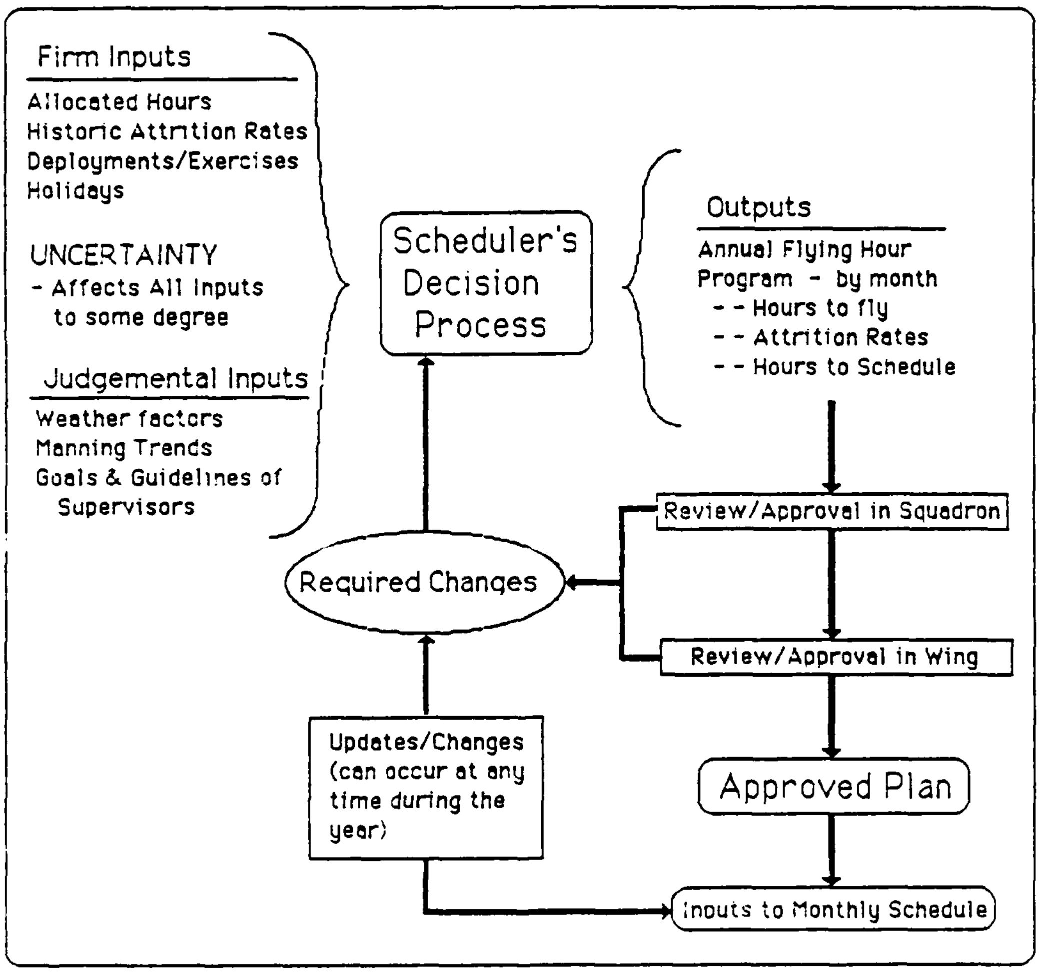

Figure 1. The Yearly Scheduling Process

judgment of the scheduler and his awareness of all of the factors that could affect the flying schedule in the coming year. The finished product provides a monthly breakdown of hours to schedule and provides milestones throughout the year. Because many of the factors upon which it's construction were based were somewhat tentative, the flying hour program is flexible and may be adjusted several times throughout the year.

Monthly Scheduling

The next phase of scheduling occurs when the hours assigned to a particular month are divided into approximate daily schedules. Figure 2 illustrates the monthly scheduling process.

The monthly plan usually takes the form of a calendar with each day noted as to the number and type of each sortie to be flown. Estimated departure times may be listed. Average sortie durations for each sortie type are used to calculate the total hours scheduled. A major factor to be considered at this point is attrition.

Attrition is the loss of scheduled sorties due to factors such as weather, maintenance, supply, aircrew availability, and other reasons. It is defined as the number of sorties lost (or cancelled) divided by the number of sorties scheduled. Historical attrition rates for all factors are maintained by maintenance and are used by the scheduler to calculate the flight time expected after attrition. Attrition is important to the scheduler because it is the flying hours actually flown (after attrition) that must be accounted for. Therefore, each month will have more sorties scheduled than are actually required so that sorties lost due to attrition do not adversely affect the annual flying hour plan.

The decision process required to produce a monthly schedule is more complicated and data intensive than that required for the yearly plan. The scheduler must begin to balance numerous and conflicting requirements for flying resources (flight time) against a limited supply. The scale of events that impact his decisions has become smaller as the period to be

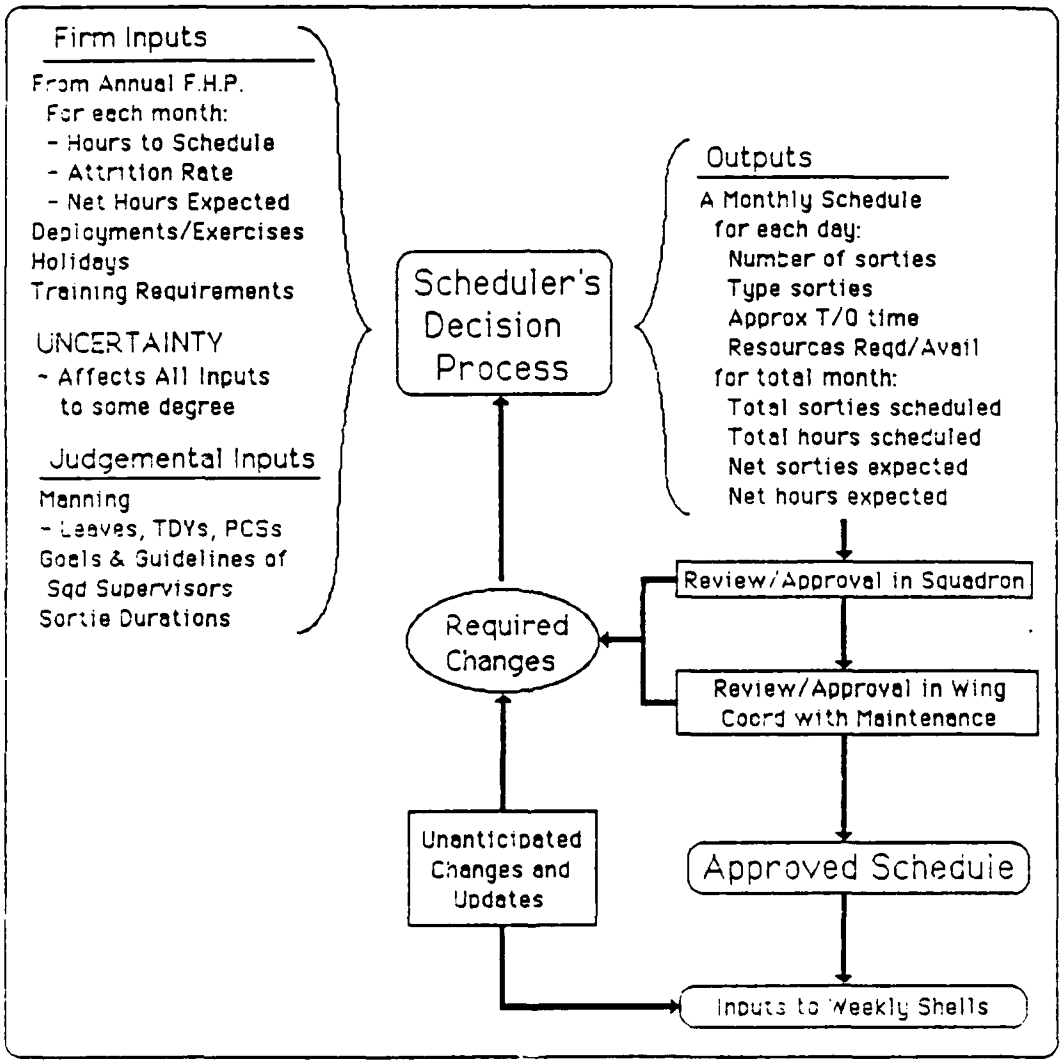

Figure 2. The Monthly Scheduling Process

scheduled has shrunk from twelve months to one. A trial and error approach is usually required as hours and sorties are added, subtracted, and adjusted until the month's total is as desired and the events scheduled meet the squadron's needs. As with the yearly schedule, the monthly plan must be reviewed and approved by the squadron leadership. This frequently results in changes to the scheduler's work.

The monthly flying schedule is usually produced one to two months prior to the start of the month in question. It will be used as a reference for routine requests such as air-to-ground bomb range times and air-to-air range times. Air refueling support may also be requested on the basis of a monthly schedule.

Weekly Scheduling

The next phase of the scheduling process is the production of the weekly "shells". The weekly scheduling process is illustrated in Figure 3.

A shell is a daily schedule with all planned flight operations data noted, but without any aircrew members assigned. A daily shell lists all sorties, the takeoff times, estimated durations, estimated arrival times, mission types, aircraft configurations, weapons configurations, and range or airspace times. It becomes the contract between maintenance and operations as to what will be produced by maintenance and what will be flown by operations.

The weekly shells are one week's worth of daily shells and are developed by squadron scheduling two weeks in advance. Maintenance provides the scheduler with the expected number of aircraft available for flight each day and any special requests for maintenance training. Special requests might be for integrated combat turns (ICT), functional check flights (FCF) or

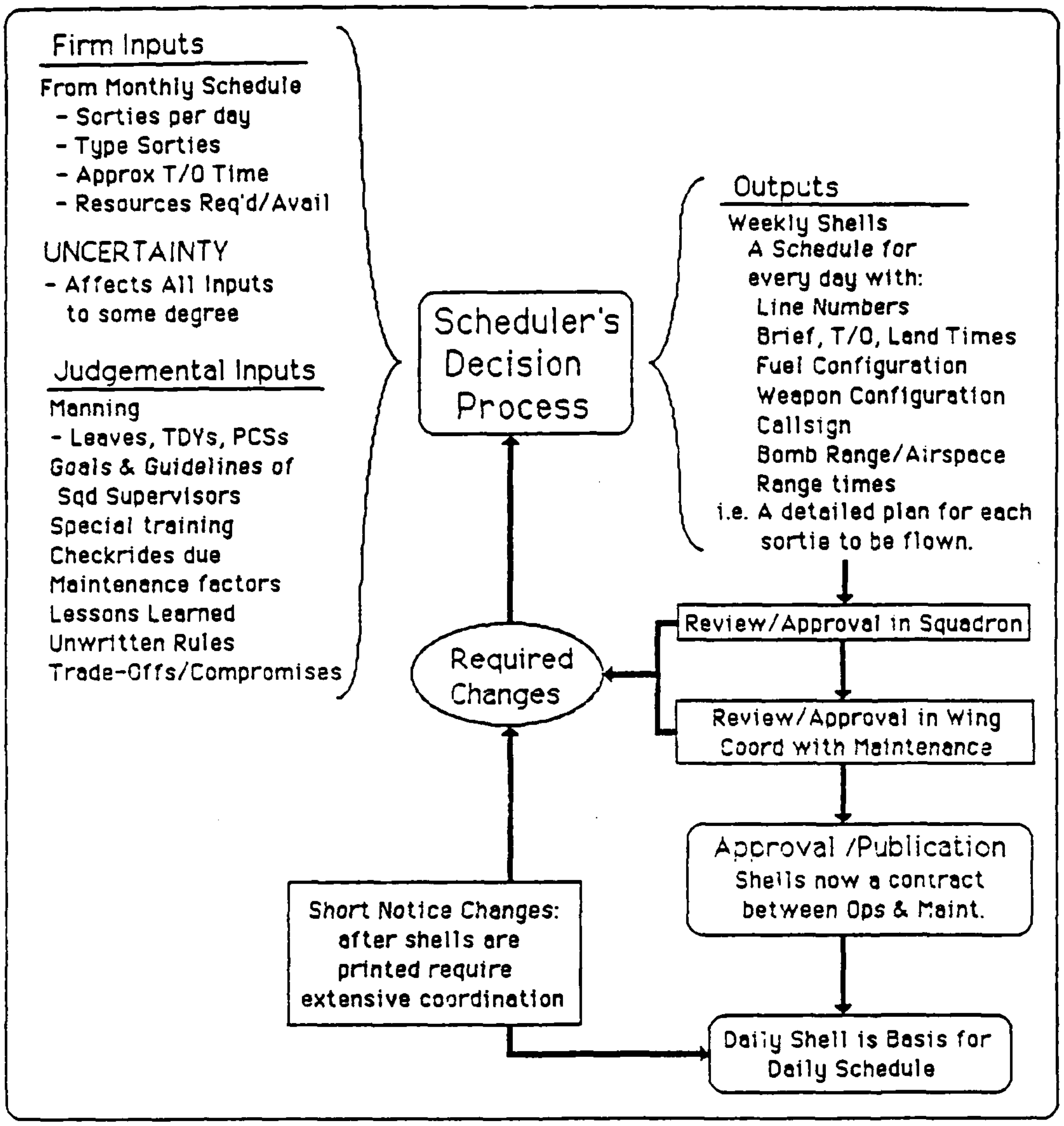

Figure 3. The Weekly Scheduling Process

aircraft delivery to or from depot level maintenance facilities located off station. Wing scheduling supplies the squadron scheduler with a list of assigned and available resources such as bomb ranges, airspace, and tankers.

The scheduler uses the data provided by wing and maintenance, the goals and priorities of the squadron supervisors, and his own judgment and experience to transform the rough outline from the monthly schedule into a detailed plan for one week. The decision process requires attention to detail and has numerous constraints. However, the scheduler does have some flexibility and there is generally no single "approved" solution to the problem he faces.

During the review and approval process, the scheduler must be able to defend his logic and decisions to squadron and wing supervisors. What is "best" or "optimum" can depend on personal bias, desires, and interpretation, so the scheduler may have to explain some of his decisions before the schedule is approved.

Daily Scheduling

The final and most difficult phase of the scheduling process is daily scheduling. Here the daily scheduler uses the guidance and priorities of the squadron supervisors to match names with events. Figure 4 is an illustration of the Daily Scheduling Process.

The objective of the daily schedule is maximum effective training, with initial and upgrade training primary and continuation training secondary. This training must be conducted under significant constraints and with limited resources. In addition to flight operations, the daily scheduler must assign the appropriate names to activities such as simulator training, academic training and squadron duties and meetings. The daily schedule is the means by which a squadron communicates, both to its own members and to people and organizations outside of the squadron, the details of the following day's operations. It should clearly and concisely describe all events to which squadron personnel or resources are committed.

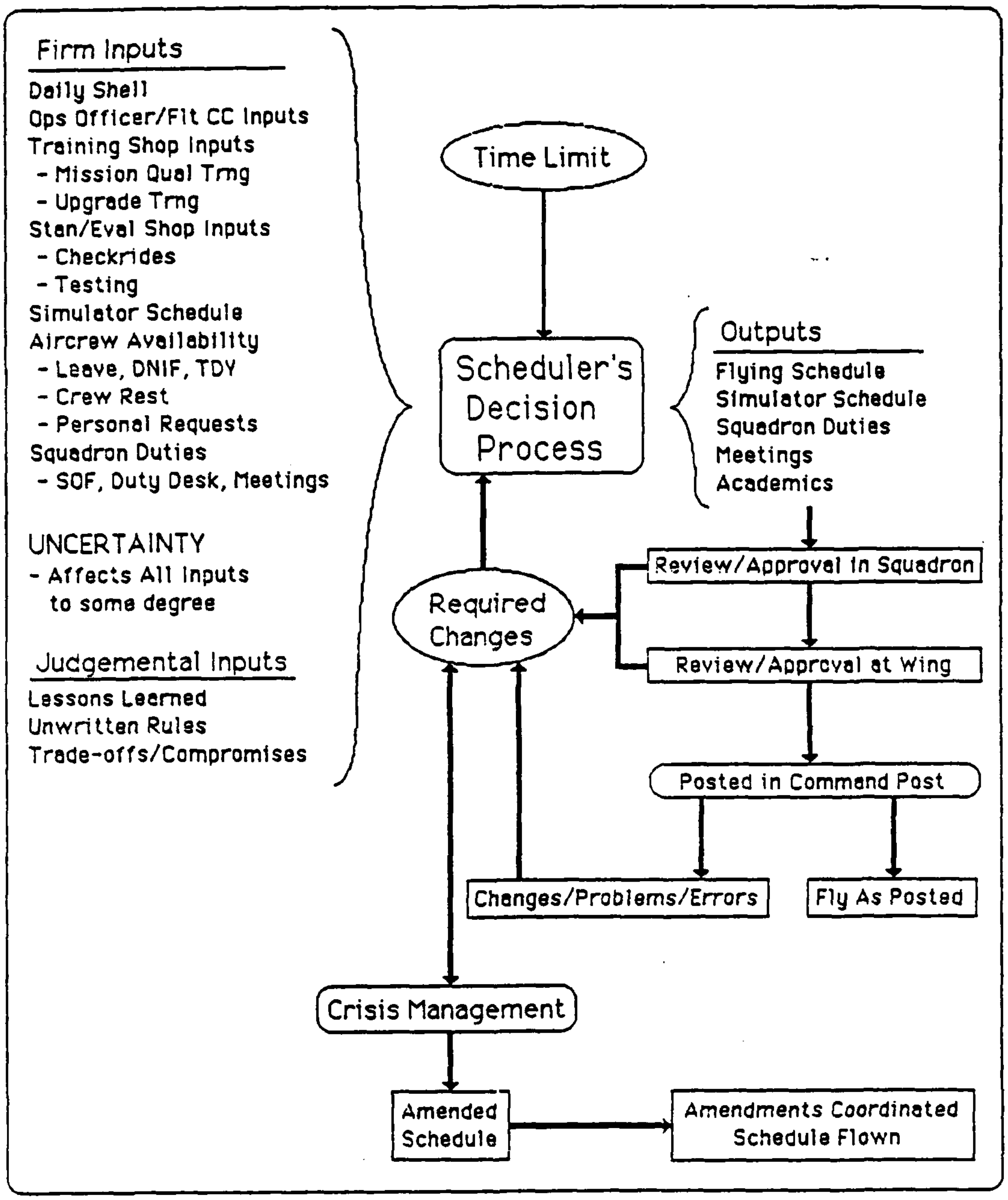

Figure 4. The Daily Scheduling Process

The first step in the daily scheduling decision process is data collection and review. Before he can effectively and correctly decide which aircrew members are appropriate for each required position in the squadron's schedule, the scheduler must know (or have access to) everything that will impact on his decisions. A scheduler can spend hours of valuable time updating his personal "database" before he assigns a single name to the schedule.

Once the scheduler has an idea of what events he is required to schedule and the resources (crewmembers) he has available, he can start the decision process that will match names with events. This decision process requires logic, attention to detail, and an excellent memory. It involves judgment, compromise, and creativity. Numerous combinations of names and events may be tried and abandoned as he looks for what he feels is the best solution. After two to four hours of work, he may have a feasible schedule on the board. An "optimum" schedule is, of course, the goal. However, the compromises forced on the scheduler by the numerous constraints, conflicting goals, limited resources, and a deadline, frequently result in a schedule that satisfies rather than optimizes. Optimal, as used in this context, suggests a schedule that maximizes training, meets all the squadron's goals, and violates none of the constraints. The major remaining hurdle left for

the scheduler is gaining operations officer approval for his work. This event often results in a required change to what may be a "house of cards" built by the scheduler. The change may be trivial and easily made, or it may be one that is difficult to make without reworking a major portion of the schedule. Often the result is a schedule that is late in completion, or that is hastily changed and is now less effective than before. After the schedule is completed and approved, the administrative chore of copying and distributing the schedule in the proper format remains. This duty requires a significant period of time and is occasionally the source of errors in quality control.

Crisis Management

A final area in which the scheduler can face serious problems is in crisis management. A crisis is generally the result of an unexpected change in conditions that call for a scheduling decision, and generally occurs in the daily scheduling phase after the original schedule has been approved and posted. A crisis can occur in a number of ways. A crewmember ean misread the schedule and not show up at the right time. People get ill and are unable to fly. Weather causes entire days to be lost, or can result in aircraft and crewmembers diverting to an alternate base. Maintenance problems can result in loss of valuable sorties. At practically every occurrence of a crisis, the scheduler will be asked by the operations officer for assistance in solving the problem. Crisis management can provide some of the scheduler's greatest challenges.

Problem Conception

The motivation to research a computer scheduling aid for a fighter squadron scheduler originated in the author's past experience. As a squadron duty scheduler, and then the chief of squadron scheduling, the author has experienced directly the frustrations of coordinating the activities of a forward based, dual-mission fighter squadron. It is a tough job, and little has been done to bring the power of the microcomputer into the scheduling office to help the schedulers make the myriad decisions required to schedule 24 aircraft and 60 men effectively.

Problem Statement

This research investigates the daily scheduling process in a U.S. Air Force tactical fighter squadron and how that decision process can be aided through a microcomputer-based decision support system that is developed by an on-site adaptive design approach.

Objective

The result of this thesis effort is a scheduling DSS developed using the adaptive design approach. The 89th Tactical Fighter Squadron, located at Wright-Patterson AFB, OH served as the subject for the project. The close proximity of the 89th TFS allowed close interaction between the user and the builder, thus removing a common barrier to multiple iterations of a DSS.

Sub-Objectives

(1) This project applied adaptive design procedures to the development of a decision support system to aid the decision process of squadron schedulers in a tactical fighter squadron.

(2) The validity of the adaptive design approach was investigated as a process for system development. The iterative feature of adaptive design was used to take a kernal system through multiple stages of development.

(3) An off-the-shelf software product was used. One task was evaluation and selection of the appropriate tool.

(4) Smooth implementation and user acceptance was a goal. The automated system developed had to be an improvement over the manual system it replaced. User acceptance depended on it.

(5) The system had to be user-friendly and flexible so that system abandonment was not likely after the research was completed.

(6) Replacement of the traditional User's Manual with Storyboards/Feature Charts/Hookbook/On-line Help was investigated.

Scope

Only normal peacetime flying conditions were addressed. Wartime, exercises, TDY deployments, and special events were not considered.

Chapter II discusses decision support system concepts, principles, and components. The adaptive design approach to DSS development is described, and the tools and techniques used to evolve an effective DSS are reviewed.

II. Methodology

DSS Definitions and Concepts

The meaning and purpose of decision support is not easily defined. An early description of DSSs describes them as "*interactive* computer-based systems that *help* decision makers utilize *data* and *models* to solve *unstructured* problems" (Sprague and Carlson, 1982:4). The key words were italicized to emphasize the unique aspects of DSS when compared to more traditional management information systems. An important feature of decision support systems is the requirement for human judgment and decision.

> The purpose of a DSS is to establish an integrated framework between the problem, the machine, and the decision maker. The DSS couples the speed and thoroughness of automation with the insight of human experience and, where appropriate, a proper blend of quantitative support (Davis, 1988:5).

The goal of decision support is not merely to process data and present results in a stack of computer output, but to aid the decision process of the user.

> Decision support implies the use of computers to:
> 1. Assist managers in their decision processes in semistructured tasks.
> 2. Support, rather than replace, managerial judgement.
> 3. Improve the effectiveness of decisionmaking rather than its efficiency (Keen and Morton, 1978:1)

DSS Components

Decision support systems are composed of three major subsystems. Each is critical to the effectiveness of the entire system and each must be integrated into the overall system architecture. These components are termed Dialog, Data, and Model.

The interactive nature of decision support systems requires that there exist an effective interface between the user and the system. This is the

Dialog component of a DSS. Sprague and Carlson state that it is the most important subsystem because it determines much of the power, flexibility, and usability of the entire DSS. "In fact, from the DSS users' point of view, *the Dialog is the System*" (Sprague and Carlson, 1982:29). "Because DSS usage is discretionary, the nature of the interface becomes a critical factor as either a barrier or a facilitative medium to any potential benefits of a DSS" (Young, 1989:112). Design of an effective dialog component is not a trivial task. An interface that communicates with the user in an effective, tolerant and forgiving manner requires 30 to 60 percent of the development (Davis, 1988:81).

The Data component of a DSS consists of a database and a database management system. Data must be available from sources both external and internal to the organization. Personal and unofficial data must be available to allow the user to employ experience and judgment (Sprague and Carlson, 1982:32). The system must be capable of effective data display, which suggests a graphical capability. Effective management of the data component is not generally an obstacle to DSS development because of the evolution of database management systems with MIS (management information systems).

The Model subsystem of a DSS serves to mathematically and analytically manipulate data to provide a basis for decisions by the user. Davis classifies quantitative techniques into five areas. They are mathematical programming, network optimization, network analysis, stochastic methods, and forecasting procedures (Davis, 1988:116-135). A model component can be as complicated as one of these traditional operations research techniques or as simple as a basic spreadsheet model. The decision process supported may require the DSS to contain a number of different models, depending on the problem, the data available to the decision-maker, or the decision process itself.

DSS System Configuration

Hardware

The choice of hardware for a decision support system is primarily a function of the size of the problem and its related data requirements. Ideally, the hardware configuration decision should be left until the end of the design process. This should help prevent the DSS from outgrowing the hardware.

Large data storage requirements may require the use of a mainframe or minicomputer. Larger computers may be used to simply store the data which can then be "fed" to a smaller workstation. However, the increase in power and storage capability of microcomputers makes them a practical and less costly alternative in many cases.

Software

The choice of software is a critical decision in DSS design because "...the character, power, and scope of a DSS can vary widely and is determined mainly by its functional software" (Young, 1989:33). Davis describes three approaches to software development: off-the-shelf packages; existing software modules patched together; and an integral system designed from scratch (Davis, 1988:162).

Off-the-shelf packages can be the quickest and most economical way to develop a DSS. However, a software package that has the flexibility to handle a variety of problems may not have enough power to handle specific problem areas adequately. There is also a risk that as the system is developed, requirements can overtake capabilities. To prevent this situation, the builder must attempt to forecast future system requirements insofar as possible. The features and capabilities of the prospective software package should also be demonstrated to ensure that they can meet the demands of any anticipated requirements (Davis, 1988:163).

A composite system patched together from commercially available functional packages may be the next best alternative. There are a number of products that perform very well in a narrow application, such as spreadsheets, data base management systems, graphics packages, and report generators. However, the task remains to integrate these packages effectively (Davis, 1988:163-164).

> The ideally integrated system is designed so that a common interface and language style enables the user to control the full functionality of the system without a sense of moving from one processing world into another (Young, 1989:60).

Davis makes the point that "products designed to permit interaction with other systems are becoming increasingly available..." (Davis, 1988:164).

The last resort in software selection should be use of a programming language to build a DSS from the ground up. The cost and time involved are distinct disadvantages to this approach. However, this may be the only way to meet all the requirments of a unique application (Davis, 1988:164-165).

DSS Design Principles

Intelligence, Design, Choice

Given the objective of supporting a decision maker, the builder/designer requires an understanding of the decision making process. In Simon's oft-quoted work, he states that the three main phases of decision making are "Intelligence", "Design", and "Choice". Intelligence is the recognition that a problem exists and includes the collection of relevant data. Design involves the search for feasible solutions. It concerns generation and analysis of possible courses of action. (Simon refers to the design of a solution to a problem, as opposed to the design of a computer system.) Choice is the selection and implementation of a particular course of action (Simon, 1960:1-3).

An effective DSS must support the user in all three phases. DSSs should be especially applicable to the design phase of the decision process, where the ability to examine numerous options can give the user more insight into the problem. The DSS designer needs specific tools and techniques for developing a system that captures the requirements of a specific user.

The ROMC Approach

Sprague and Carlson suggest the ROMC approach to give a user-oriented approach to the development process. The "R" is for representations, and refers to how the system will represent to the user the problem and all related data. The "O" is for the operations that will manipulate and analyze those representations. "M" is for memory aids that will connect representations and operations, and "C" is for the control mechanisms that the user employs in handling the system (Sprague and Carlson, 1982:96).

The ROMC approach is valid for DSS design because of the type of problems and the characteristics of the users that decision support is designed for. ROMC methodology gives the designer the guidance necessary to ensure that a DSS has the flexibility to adapt to different users and their different decision processes. "The most important characterisic of the ROMC approach is that it is a *process-independent* approach for identifying the necessary capabilities of a Specific DSS" (Sprague and Carlson, 1982:101). Process-independent means that regardless of the decision process that the system is to support, the developed system will effectively represent the data for the user, help him/her to remember and manipulate the data, and allow the user to control the decision aiding process.

Adaptive Design

An important tool for requirements determination is the adaptive design approach. "The very nature of DSS requires a different design technique from traditional transaction processing systems" (Sprague and Carlson, 1982:15). Researchers have documented obstacles within, among, and between participants in the requirements determination process (Valusek and Fryback, 1985:103). An inadequate statement of requirements can result in the fielding of a computer system that does not effectively address the problem and that is subsequently abandoned.

The traditional systems approach, also called the systems development life cycle, is a five step procedure. It consists of (1) requirements analysis; (2) general and detailed design; (3) system construction; (4) systems testing and installation; and (5) operations and maintenance (Young, 1989:165). This approach "freezes" user requirements at stage two and therefore severely restricts flexibility in the design process. It is unreasonable to ask a user to define all requirements for a decision support system when faced with an ill-structured problem and a poorly defined decision process in an environment of information bombardment.

Adaptive Design has the same essential steps as the traditional systems design approach, but is compressed into a short cycle that is repeated in an iterative manner. The design of a DSS begins with a small part of the problem and then grows with each iteration Feedback from users is incorporated into each new version of the system. In this way, user requirements can change and grow as the user and designer learn more about the problem and the decision process.

Three Phase DSS Development

In a recent work on DSS, Lawrence F. Young outlines a three phase approach to specific DSS development (Figure 5). This three phase approach provides an overall structure within which adaptive design tools and techniques can be applied. Phase one is preliminary assessment, phase two is base system development and assessment, and phase three is iterative development, usage, and assessment (Young, 1989:171).

Phase One

The first phase, preliminary assessment, consists of situation analysis and interface assessment and selection (Young, 1989:171). Situation analysis refers to examination of the problem, the user(s), and the organizational environment in order to determine if a DSS approach is appropriate. Interface assessment and selection involves review of available and appropriate software and hardware tools, followed by an estimation of the potential worth of a successful DSS application in light of it's cost. Phase one may conclude that the DSS approach is inappropriate and should be abandoned, or it may lead to phase two.

Phase Two

The second phase in Young's development process is base system design and assessment. The purpose is:

> 1. To relatively quickly create a working system that can serve as a basis for further problem analysis and modification; and
> 2. To enable immediate assessment through actual use to determine whether of not it appears worthwhile to continue to use and/or develop the DSS (Young, 1989:171).

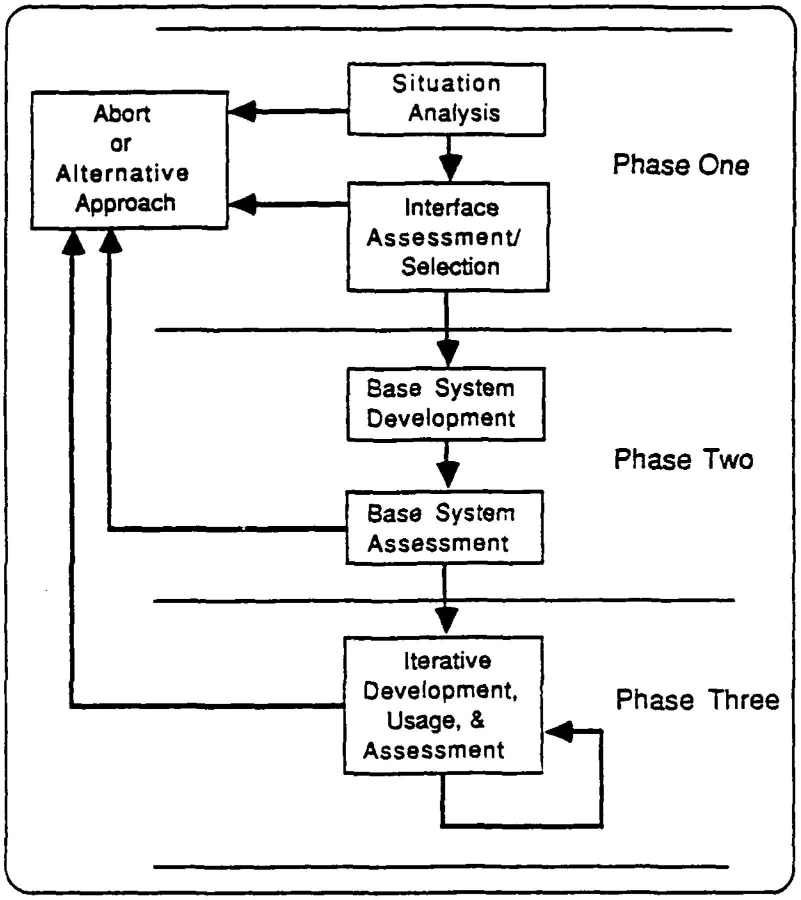

Figure 5. Three Phase DSS Development (Young, 1989:170)

Phase Three

The final stage is an iterative cycle of system development, use, and assessment. It begins with the base system and may continue as long as the DSS is deemed appropriate and effective. The first iteration will be modification or extension of the base system's representations and operations. With further iterations, the user should take a greater role in system extension.

> Because usage is integrated with further development, it is highly desirable that users can now carry out changes and additions to the system largely on their own with minimal assistance from specialist DSS builders (Young, 1989:198).

Adaptive Design Tools and Techniques

Young's three phase specific DSS development approach provides a framework for development and extension of a decision support system. However, the designer/builder requires proven tools and techniques to determine the most appropriate starting point for the base system and to ellicit further user requirements for system extension.

Kernel Selection

The question of where to start the iterative design process is termed kernel selection. The kernel problem can range from "the most mundane, time-consuming task that lends itself to automation" to tasks "which are ill-defined or may never have been done before" (Valusek, 1988:107). In the case of the former, the implementation of a system that produces visible results quickly is the goal. In the latter case, the objective is a system that gradually encompasses more of the problem as the user and designer learn more about the problem and the decision process. Kernel selection is critical to the development of an effective base system that can continue to evolve. Concept maps and storyboards are currently being researched as tools for aiding in kernel selection.

Concept Mapping

A concept map is a graphical representation of concepts and relationships among them. The purpose of a concept map is to communicate or represent a state of knowledge (Novak and Gowin, 1984:Chap 2). In the case of DSS design, concept mapping is being researched as a tool to be used to to quickly get a feel for the problem, it's related data, and the decision maker's perspective on the problem. It may eventually become the means by

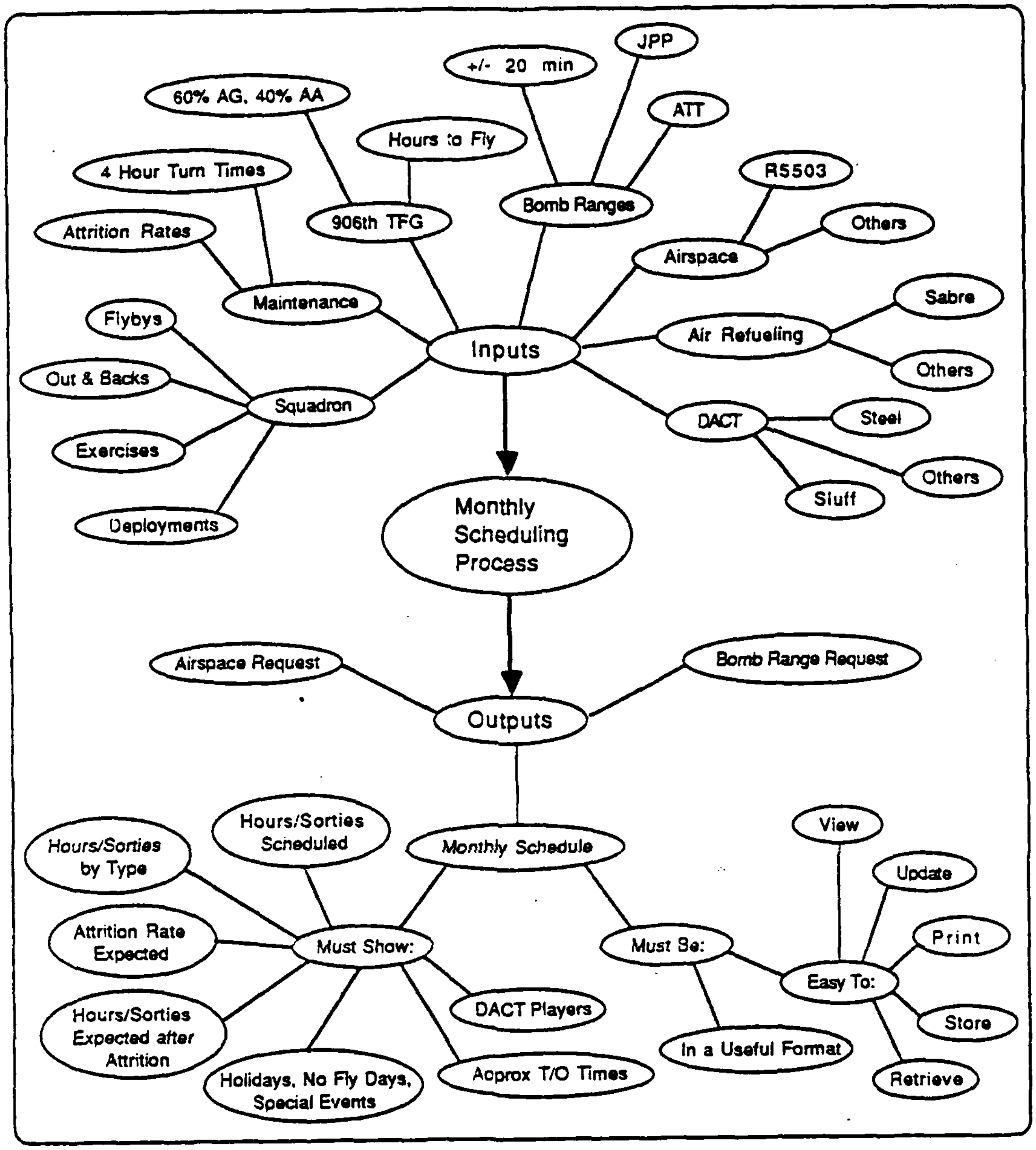

Figure 6. Concept Map of 89th Monthly Scheduling

which the user himself defines requirements for a future DSS. As an example of a concept map, Figure 6 came from a 90 minute discussion with the schedulers at the 89th TFS and represents the important factors in the monthly scheduling process.

The time saving aspect of concept mapping is critical. To reduce the impact on the user, the designer/builder should plan on spending no more than one hour with the user at any one time. Three one hour sessions should be the maximum required of the user for the builder to gather the required information. Concept mapping the problem domain will help the builder to produce storyboards.

Storyboarding

A storyboard is a representation of a screen display in the DSS under development. It can be hand-drawn or produced with appropriate software. The idea is:

> ...to develop -- in conjunction with the users -- a set of displays that represent each and every path users might take once the system is developed. The purpose of storyboarding is to represent all of the important functions that the system will perform as well as the output that it will generate (Andriole et al, 1987:5-6).

In phase one, base system development, concept mapping and storyboarding will promote the selection of an appropriate kernel. Later, as the DSS evolves and expands, storyboards can be used to communicate evolving requirements as the system develops. The hookbook concept is central to the idea of an evolving system.

Hookbook

The hookbook consists of structured notecards which allow the user to log thoughts and ideas about how to improve the system (Valusek, 1987:109). The idea is to provide the user a readily available means for input into future system requirements. A properly implemented adaptive design process will allow the system to grow and incorporate these user-defined requirements. Storyboards developed from the user's hookbook entries will provide the builder a statement of requirements.

The Appropriateness of DSS

There are different types of scheduling problems and a variety of settings in which they can be found. As a class, scheduling problems can be approached with a number of solution techniques. Linear programming, simulation, and expert systems all have potential contributions to make towards a scheduling problem's solution. However, there are two major features of scheduling in a U.S. Air Force fighter squadron that make these techniques inappropriate as a starting point for systems design. These two features are: 1) frequent and rapid change in parameters and priorities, and 2) a requirement for a "man-in-the-loop".

One of the major problems faced by a scheduler in a fighter squadron is change. The allocation of annual flying hours may change at any time due to reasons that range from national budgetary considerations to squadron or wing-level decisions. Monthly and weekly schedules may require adjustment for weather, squadron manning, or aircraft maintenance problems, among other reasons. Daily scheduling is affected by problems that range down to the individual crewmember level. The one constant in squadron scheduling is that nothing remains the same. All too often these changes require rapid adjustment of scheduling decisions to limit damage or take advantage of an opportunity.

The second major feature of fighter squadron scheduling is the requirement for human problem solving capability. A scheduler must frequently use judgment, creativity, tact, foresight, and intuition as he works on each phase of the scheduling problem. There are no equations, algorithms, or if-then rules that can completely replace the human mind in this decision process.

It is the presence of frequent change and the requirement for human decision making that limit the applicability of many mathematical or computer-based problem solution techniques. A solution technique must remain flexible if it is to remain useful. Schedulers do not have the ability or the time to reformulate an LP, rewrite computer code, or modify a knowledge base every time a change occurs.

Scheduling and DSS

The presence of change and the requirement for user involvement are not obstacles for a decision support system. Changes in the problem do not necessarily require changes in the DSS; often a change in the database is sufficient. If a change in the system is required, the mechanism of adaptive design aids system growth and evolution. The requirement for human judgement is not an obstacle to DSS, it is a strength. As defined, an effective DSS does not make the decision for the user, it helps the user make a better decision.

Recent DSS Research Projects

Several recent theses at the Air Force Institute of Technology have employed DSSs and their adaptive design with some success. A number of these research projects have used real-world scheduling problems as the basis for the research. However, a recurring theme of these research efforts has been the problems presented by the geographic separation of the user and the builder. The inability to frequently and directly interact has generally limited the evolutionary adaptive design methodology to one iteration. Limited contact with the user has also affected other aspects of the specific DSS application development process.

In his thesis on a DSS for AWACS programmed flying training, Schneider states:

> In this research effort the physical separation between the two [user and builder] degraded the effectiveness of adaptive design. Questions that could have been resolved simply, given face-to-face interaction, became stumbling blocks (Schneider, 1987:5-4,5).

Kopf, in his work on a decision aid for aircrew scheduling in the 6916th Electronic Security Squadron, had users located at Hellenikon Air Base near Athens, Greece.

> Due to the distance and lack of timely communications between the 6916ESS and AFIT, there was little builder-user interaction. As a result, the design and implementation processes were not as effective as they could have been (Kopf, 1987:5-10).

The extreme distance impacted on user evaluation of the DSS as well. "A system evaluation has not been completed because of the lack of interaction with the user" (Kopf, 1987:5-7).

The most ambitious scheduling DSS effort to date was that of Trapp and Grechanik. In the project Design Evolution of a Fighter Training Scheduling Decision Support System, Captains Trapp and Grechanik developed a DSS to support the scheduling of fighter training sorties at Holloman AFB, New Mexico. The DSS they created was designed to aid the scheduler in assigning the appropriate instructor to a student and to ensure that the student had accomplished all the prerequisite training for each flight. This specific DSS development was also plagued by separation.

> Lack of available thesis time and TDY funds limited the use of adaptive design to the first iteration. Unless co-located, user feedback is hard to get and usually impossible due to only telephone conversation. These authors conclude that the best way to do adaptive design is on-site in the user's organizational environment. Immediate feedback would be available to the builders from the most recent users (Trapp and Grechanik, 1987:71-72).

Trapp and Grechanik began their DSS development work at the daily phase of scheduling. They used as a starting point for the adaptive design process the instructor-tostudent matching decision, which they considered central to the overall daily scheduling decision process. In spite of the limitations cited above, they developed an effective, albeit abbreviated, DSS for squadron schedulers at Holloman AFB, New Mexico. With real-time user input and feedback, a second iteration may have encompassed the entire daily scheduling phase.

Chapter III describes the application of DSS design principles and procedures to the development of a specific DSS. The chapter begins with a short description of the scheduling environment in the project squadron, the 89th TFS.

III. Methodology Application

The 89th Tactical Fighter Squadron

This research used the 89th Tactical Fighter Squadron as its problem environment. The 89th TFS is a U.S. Air Force Reserve fighter unit equipped with 18 F-4D aircraft and manned by 12 full-time crewmembers and 32 part-time crewmembers. It is located on Wright-Patterson AFB, OH. The 89th TFS "Rhinos" have the primary mission of air superiority and a secondary mission of ground attack. They fly five days per week and are required to maintain the same Mission Ready (MR) status as active duty fighter squadrons. As a reserve unit, they have some unique scheduling problems while other problems experienced by active units are not a factor.

The 89th TFS does not offer the full range of scheduling problems that a Regular Air Force active duty fighter squadron might. It has only 18 aircraft in the squadron versus the 24 in most active squadrons; therefore, there are fewer aircrew members and fewer hours to fly. There is no F-4 flight simulator on base, which eliminates a significant scheduling and training requirement. The 89th is the only fighter unit on base, rather than one of three (the usual case). This greatly reduces the problem of coordination and competition among squadrons for the same resources.

As a reserve unit, the 89th has fewer PCSs, TDYs, and Leave requests compared to an active squadron. This suggests that aircrew training requirements are smaller and aircrew availability is more stable. However, the fact that part-time flyers are only available a few days a month adds complexity to the aircrew availability issue. Scheduling of part-time reservists becomes a high visibility issue if the squadron's mission ready rating starts to drop due to low sortie rates or lack of training.

The scheduling function in the 89th TFS is a two-man operation, separated into long-range scheduling and daily scheduling. The long-range scheduler is responsible for yearly through weekly scheduling, while the daily scheduler's duty involves the placing of names against events for each day's operations. The salient features of scheduling in the 89th will be discussed as the development of the specific DSS is described.

Specific DSS Development

This section discusses the application of the methodology reviewed in chapter two to the scheduling problem as it exists in the 89th TFS.

Phase One

The first step in this research project was determination of the willingness and suitability of the 89th TFS as a subject for study. A short visit to the squadron in July 1988 was encouraging. Both schedulers were willing to discuss the possibility of a computeraided scheduling system. A second meeting was set up for further discussion.

The second meeting took place on August 19 and was attended by the author, his thesis advisor, the long range scheduler, Captain Bill Whittaker, and the daily scheduler, LtCol Tom Kollin. The schedulers were briefed on the basics of DSSs, what they could expect during system development, and what would be expected of them. Following that, there was a 90 minute discussion on the scheduling problem in the 89th TFS, current procedures used to solve the problem, and specific characteristics required of a DSS application. The meeting concluded with an agreement that the 89th would serve as host for a scheduling DSS application.

This meeting was a critical part of the situation assessment process of phase one. The author, serving as designer/builder, examined the task, the users, and the environment and was able to determine that a DSS was indeed applicable. There was a great deal of information and knowledge elicited during this session. This information was recorded in a concept map that would later help determine the kernel for the base system.

The second part of phase one, interface assessment and selection, occurred over the next few weeks. A second visit to the 89th determined that Zenith Z-248 desktop computers were available with typical bundled software, including Enable, the integrated software package. However, the schedulers were not frequent or experienced computer users.

Enable has been used in prior DSS research projects as a DSS generator with varying degrees of success. Enable offers spreadsheet, database, word processing, graphics, and telecommunications capabilities in one integrated package. User-developed menus, macro commands, and a windowing capability are some of Enable's strong points. Trapp and Grechanik used Enable in their fighter scheduling DSS with some success. They were hampered, however, by an inability to use macros across applications, i.e. from spreadsheet to database. A subsequent version of Enable has apparently solved that problem.

Enable was not the only prepackaged software product considered for this research. Symphony, a product of the Lotus Development Company, and SmartWare, from Informix Software, were also considered. However, because of Enable's immediate availability, it's past successes as a DSS generator, and it's strong points listed above, it was selected as the DSS generator for this project.

Phase Two

The objective of this phase of the specific DSS development was to quickly produce a working system, called the base system, that could be subsequently analyzed, modified, and extended. The adaptive design tools of concept mapping and storyboarding were used to determine the appropriate kernel (starting point) for the base system.

Concept Map

The concept map from the initial meeting in August was examined for indicators of a suitable kernel. A tentative kernel appeared to be in the long range (yearly to weekly) scheduling arena. Two facts supported this conclusion.

First, the daily scheduler had an excellent "manual DSS" in the form of a greaseboard. This board provided, at a glance, scheduling information for the next 30-60 days for all members of the squadron. Any crewmember could easily review or update his availability for flying. He could also see how many sorties he was scheduled for, and how many he had flown to date. The scheduler was able to use the availability and sortie count for each crewmember to schedule each person so that minimum sortie levels required by regulation were maintained.

A computer-based scheduling aid would have to be extremely flexible and at the same time, extremely user-friendly to match the current system. The man-machine interface capabilities of Enable did not appear to facilitate this environment. Unless it was an improvement over the manual system, a DSS would not be accepted by the users.

Second, a chronological approach to the different phases of scheduling made good sense. Each phase of the scheduling process provides inputs to the next phase. To begin the DSS development cycle at the beginning of the scheduling cycle was a logical conclusion.

A second meeting was held October 6 with Capt Whittaker, the long range scheduler. A detailed discussion ensued as to what information and procedures were required to conduct long range scheduling. This one hour exchange helped to complete the concept map begun in the first session. Whittaker also provided copies of the various printed forms used in each phase of long range scheduling.

89th Yearly Scheduling

Yearly scheduling in the 89th TFS is dictated to a significant degree by direction from higher headquarters, the 906th Tactical Fighter Group. When the squadron receives its allocation of annual flying hours from the group, it has already been separated into quarterly blocks of flight time. However, these quarterly blocks are not equal; some consideration has already been given to factors which affect the squadron's scheduling capacity. In general, the squadron scheduler then simply divides each quarter into thirds to produce an approximate monthly flying hours goal. This practice of allocating flight time in three month blocks takes away some of the squadron's flexibility. However, squadron schedulers are still allowed to adjust individual month's flight time and in fact, must frequently do so.

89th Monthly Scheduling

The monthly scheduling process in the 89th is a time-consuming, difficult process. The scheduler starts with the decision of how many hours to schedule for the month. Adjustment of the figure from the yearly plan may be required to account for a previous month's schedule or for an expected overage or shortage of available aircrew members. The attrition figure is factored in to account for expected sorties lost. In the 89th, the schedulers use only weather attrition to calculate the number of sorties to schedule. The hours he must schedule equals the hours he must fly for the month, plus the hours he expects to lose due to weather.

Once the scheduler has decided how many hours he has to schedule, he must start to schedule actual sorties by day and type. The 89th TFS has the goal of approximately 60% of it's sorties being air-to-air and 40% being air-to-ground. Air-to-air (AA) sorties are planned for .9 hours duration and air-to-ground (AG) sorties planned for 1.1 hours. Other sortie types may be scheduled, such as air-to-air refueling (AAR), cross-country (XC), etc. Each has an associated sortie duration used to calculate total hours scheduled.

The 89th has certain events that remain fairly constant from week to week. These generally involve agreements with other fighter units to conduct air combat training at standard times and locations each week. Air refueling may also be a weekly training event. The scheduler generally builds a month's schedule around these routine events. The standard week in the 89th is normally Tuesday through Saturday, with no planned flying on Sunday and Monday.

Using a standard week as a basis, the scheduler adds sorties until he reaches the monthly goal. This is frequently a trial and error process, as different sortie types and durations are added, removed, and shifted until the goal is met. A running total of hours scheduled and percentage of AA and AG sorties must be maintained. When the scheduler has a balanced monthly plan that meets the constraints and goals, he can present his monthly schedule to the operations officer for approval.

An approved monthly schedule is one product of the monthly scheduling process. Two other important products are the airspace request for air combat training and the bomb range request for air-to-ground weapons training. These resources must be requested in writing at least 15 days prior to the month required.

The monthly schedule may be revised after it is initially approved for a number of reasons. Resources requested may not be available and adjustment in the monthly schedule may be required. The availability of part-time aircrew members, weather problems, and maintenance factors may also require changes. The scheduler may have to rework a single month several times before it is executed.

Storyboards

The next step was the production of storyboards to represent a possible instance of a base system. The initial thrust in storyboarding was toward the monthly scheduling phase. The reason the monthly phase was chosen rather than the yearly phase had to do with how the squadron received its yearly allocation of hours.

Seventeen storyboards depicting desired system capabilities were hand-drawn using Sprague and Carlson's ROMC approach for maximum effectiveness. Evaluation and modification of the storyboards by the designer/builder followed, and continued until it seemed that the decision process had been captured.

The author's past experience in scheduling was a great advantage in storyboard production. However, at times this past experience caused in the author a tendency to view the storyboards from a user's perspective, rather than from the designer/builder's.

Concept Map and Storyboard Review

The concept map and the storyboards depicting an automated monthly scheduling procedure were presented to Capt Whittaker on October 20. In general, the response was very favorable and the use of storyboards proved to be an effective way of eliciting reaction and input to the system development process. Whittaker had a total of ten changes to the concept map and storyboards. This initial feedback corrected some errors and misconceptions by the builder. The user also benefited by gaining an increased understanding of system design and intent.

In addition, there was a great deal of discussion on the most effective way to present a monthly schedule on a small monitor screen. The builder also emphasized the role of the hookbook function as a means of user input to the future system. This would be the first of many attempts to get the user to view the hookbook as a set of requirements for system expansion, rather than a way to correct or improve the current system.

The Kernel System

Over the next six weeks the kernel system was developed and tested by the builder. A significant portion of that time was spent learning Enable and it's capabilities and limitations. The dialog component of the DSS also demanded a great deal of time. The goal was to make the system easy for an unsophisticated user to control, yet retain the power and flexibility required for a viable system. Windows, macros, and custom menus were used to create a user-friendly system. An electronic hookbook facility was designed to permit on-line logging of user thoughts on system problems and extensions. On December 1, the Rhino Scheduling System (RSS) was installed at the 89th Tactical Fighter Squadron. Chapter four describes RSS in more detail.

The kernel system delivered to the 89th schedulers did not incorporate all of the features envisioned by the storyboards or required during the monthly scheduling process. Specifically, the ability to produce bomb range and airspace requests was missing completely. Also, the printed copies of the monthly schedule were not in an easily used format, and the hookbook facility would not always function. The problem with the hookbook was due to the limits on computer RAM memory. When several memory-hungry windows were open there would not be enough memory left for the hookbook. However, the requirement to rapidly develop a base system outweighed the builder's desire for perfect system performance.

User reaction to the kernel system was generally positive. After using the system for two weeks, Captain Whittaker had five inputs for the current system. These were passed on verbally; the hookbook facility was not used at all, even though it was available much of the time. There were no inputs or ideas for the next phase of the scheduling DSS, the weekly phase, only for the system fielded at the time.

Because the development of the DSS began with the monthly phase, RSS was not used on a daily basis. This was not unexpected; the scheduler is only concerned with monthly scheduling a few days per month. Unfortunately, this meant that the kernel system was not heavily used. However, the feedback from Captain Whittaker was generally valid and contributed to the next iteration of RSS.

The positive reaction of the user to the kernel system led to the conclusion that the DSS approach was a valid methodology for the 89th TFS scheduling problem and that further iteration should continue. The next version of the Rhino Scheduling System would correct deficiencies of the base system and better support the monthly scheduling process.

Phase Three

The third phase of Young's development process is an open-ended cycle of system modification or extension followed by use and evaluation. It begins with the base system developed in phase two.

Second Iteration

The second iteration of RSS, modification of the kernel system, was accomplished in two weeks and involved a greater use of Enable's database management system. The database system was required to manipulate the data entered in the monthly schedule so that RSS could produce the appropriate requests for bomb ranges and airspace. The format change was required to allow data transfer to the database from the spreadsheet. Also incorporated at this time were five man-machine interface (MMI) changes requested by the user. These inputs concerned how and what information the screen presented to the user, and were directed towards features of the current system, rather than a future, expanded system. As in phase three, these user inputs were passed verbally, rather than with the hookbook facility.

The result of this iteration was a much more capable monthly scheduling DSS, delivered on 12 January, 1989. The user starts with a figure for the number of hours to fly for the month and the historical attrition data, and finishes with a monthly schedule complete with range and airspace requests. User reaction to the system continued to be highly positive. However, RSS still did not meet the requirements as determined by the storyboards.

Third Iteration

A third iteration of the monthly scheduling phase of RSS was required to bring the system up to full operation. Specifically, there were two features still not totally functional.

The first involved a menu choice that allowed the user to set the date of the change to/from Daylight Savings Time. This was important because the takeoff and landing times are in "local" time while the range and airspace times are in Universal Coordinated Time (or "zulu" time).

The second feature that was incorporated was the distinction between "Building a Month" and "Modifying a Month". The storyboards called for the choice "Build" to erase whatever had previously existed in that month, i.e. to erase last year's month. "Modify" would be used to update or complete a month that had been recently constructed.

This third iteration also integrated some changes to the printing formats and included a minor change to the monthly schedule template. These MMI inputs all came from the user, Capt Whittaker.

Iteration three completed the monthly phase of scheduling in RSS. User reaction to the system continued to be very favorable, and so it was decided that iteration four would enter into the weekly scheduling phase.

Fourth Iteration

A brief overview of weekly scheduling procedures is presented as a basis for the fourth iteration. Although this would be the fourth iteration for the DSS, it was the first time the weekly scheduling phase would be included in the system.

89th Weekly Scheduling

The production of weekly shells occurs two weeks prior to the week in question. The scheduler uses the most recent monthly schedule and any new information that is available to produce a detailed plan for each day of the week. Each aircraft to be flown is listed along with appropriate times, fuels, weapons, mission type, callsign, and resources assigned.

The scheduler breaks the day into parts; a first "go", a second "go", and possibly a third. Most aircraft in the first go will "turn" to the second go, and so a typical day may be a ten turn eight, which means ten aircraft in the first go with eight in the second. This requires the scheduler to balance sortie types and aircraft configurations (fuel and weapons load) between "go's" to prevent excessive maintenance workload. Maintenance also requires three hours between scheduled takeoff times for each aircraft to allow sufficient time for servicing.

Because these shells become the contract between operations and maintenance, they have to be carefully prepared and reviewed to ensure that they accurately reflect the goals and capabilities of the squadron. This means that the shells may go through a number of revisions prior to approval and publication.

RSS Iteration Four

The fourth iteration was the first attempt at the weekly phase of scheduling and therefore represented a major step forward in system complexity. The initial concept map was reviewed by the builder and then a very rudimentary set of storyboards for the weekly scheduling path constructed. These storyboards were intended for the builder's reference only and were not reviewed by the user.

Using these storyboards, new menus and macro commands were developed. The database report generator was used to replicate the daily flying schedule, AFRES Form 119, used by the scheduler to produce the shells. Macros were written which would take the basic information for one day from the monthly schedule and embellish it with default values for various times, and fuel and weapons loads. Menu selections offered the user the options of viewing a day's shell on the screen, editing the shell, and printing the shell.

Capt Whittaker reviewed the computer-produced shell and provided the appropriate default values. He also stated requirements for user-selectable default values. These default values provided for system flexibility while at the same time allowing ease of use. This review required only fifteen minutes of the user's time.

Iteration four was completed and delivered on 25 April 1989. A short training session of about 30 minutes duration was conducted, during which the user made a request for an increased capability in the shell editing facility. Capt Whittaker was asked to use the system for a few weeks, note any problems, and consider possible improvements or extensions.

The next session with Whittaker took place on 11 May. He had eight inputs; four concerning the appearance of the printed daily shell, and four requesting more options when editing a daily shell. All but one of these were corrected on the spot. The final request required more time than was available. It was clear that the DSS had been in use on a daily basis. Capt Whittaker's attitude towards the system continued to be positive and

Session/Task	User Time Required	Cumulative User Time
Presentation of Project Concept	15 Min	0 + 15
Presentation of DSS Principles, Possibilities, and Requirements	30 Min	0 + 45
Knowledge Gathering on 89th TFS Scheduling Problems, Tasks, Products	90 Min	2 + 15
Knowledge Gathering on Long Range Scheduling (Yearly, Monthly, Weekly Phases)	60 Min	3 + 15
Review of Initial Set of 17 Storyboards	30 Min	3 + 45
Delivery of First Iteration of RSS Short Training Session	30 Min	4 + 15
Meeting to Elicit User Reaction to 1st Iteration and Any Inputs for Future Work	20 Min	4 + 35
Delivery of Second Iteration of RSS Short Training Session	30 Min	5 + 05
Meeting to Elicit User Reaction to 2nd Iteration and Any Inputs for Future Work	15 Min	5 + 20
Delivery of Third Iteration of RSS Short Training Session	20 Min	5 + 40
Review of First Cut of Printed Daily Shell	10 Min	5 + 50
Delivery of Fourth Iteration of RSS Short Training Session	30 Min	6 + 20
Meeting to Elicit User Reaction to 4th Iteration	20 Min	6 + 40

Figure 7. User Time Requirements

constructive. A fifth iteration would improve and expand the system; however, time constraints did not allow further work.

User Involvement in Adaptive Design

One of the strengths of the adaptive design approach is its minimum impact on the user in terms of his/her time and effort to specify and evolve the requirements for a DSS. Figure 7 is a table of the time required of

Capt Whittaker in the design and development process of RSS. The results indicate that this DSS went through four iterations yet required less than seven hours of the user's time to set system requirements.

This DSS was developed over a seven month period as an academic research project. If the designer/builder could have worked on this system full-time instead of sporadically, it could possibly have been completed in less than two months. This means that less than one hour per week of user time would have been required to develop and expand a DSS built to the user's specifications.

A discussion of the major elements of the Rhino Scheduling System are presented in Chapter IV. The results and conclusion of this research and development effort are in Chapter V.

IV. The Rhino Scheduling System

The Rhino Scheduling System (RSS) provides decision support for the squadron scheduler in the monthly and weekly scheduling phases. The system was built using the integrated software package Enable and makes extensive use of the spreadsheet module, the database module, menus, macros, and windows. The builder's goal was a system that was easy for an unsophisticated computer user to operate, yet had enough power and flexibility to actively aid the user's decision process and not merely automate some phase of it.

This chapter will discuss the three major DSS components of data, dialog, and model as incorporated in RSS. A more complete discussion of details of the system can be found in the RSS Maintainer's Manual located in Appendix B.

Data

Data is used in RSS in both phases of the scheduling process supported. In the monthly phase, the data is stored mostly in spreadsheet format. This is because most of the monthly scheduling process occurs in a spreadsheet environment. In the weekly stage, the database module is extensively used.

Monthly Phase

Each month of the year is stored in a spreadsheet format with a label of [Month]CAL.SSF where [Month] is a three-letter abbreviation for the month. An example would be FEBCAL.SSF for February. There are twelve such files that are reused in subsequent years. The user can enter and store a schedule for each day of the month.

Each month also has a spreadsheet file which contains summary information. This information includes the number of each type sortie scheduled, the total hours and sorties scheduled, the attrition rate used, and more. These twelve files are labeled with a three-letter abbreviation of the month (e.g. FEB.SSF).

In order to transfer the data contained in each monthly calendar to the database for manipulation and reporting, it must be in the appropriate format. Consequently, there are twelve more files which the user never sees, but are there to reformat the data from a presentation that is more easily viewed and understood by the user to one that is compatible with the database module. These files are labeled [Month]DB.SSF (e.g. FEBDB.SSF).

A convenient and time-saving feature of RSS is the use of a "standard" week that can serve as the core for an entire month's schedule. This standard week is stored in the spreadsheet file called STNDRDS.SSF and can be modified and saved by the scheduler whenever the situation dictates.

The historical attrition data for each month is stored in a single spreadsheet titled ATTRIT.SSF. The source of this data is a document published periodically by the maintenance organization. This spreadsheet must be manually updated by the scheduler whenever new information is received from maintenance.

The annual flying hour program is maintained in a spreadsheet file labeled FHP.SSF. This data file is manually updated by the scheduler whenever new guidance is received from the 906th TFG.

Weekly Phase

In the weekly phase of scheduling, RSS uses information entered and stored in the monthly phase to produce a scheduling shell for each day. The data for each month is stored in a database labeled [Month]DAT.DBF (e.g. FEBDAT.DBF), for a total of twelve. Each record in this database has a field for each entry made by the user in the monthly scheduling phase. It also has fields that are derived from the user-entered information. These fields are used to calculate the start and end times of bombing range and airspace training periods.

A second database used to produce shells is the one labeled SHELL1.DBF. This is a single database that will contain user-entered and derived fields for the specified day of a particular month. Before each daily shell is produced, the information for that day in loaded into SHELL1.DBF from say, FEBDAT.DBF. A macro command will then embellish this data by entering into other fields information that depends on decisions made by the user at other menus. These other fields will contain the appropriate entries for categories of information such as fuel loads, weapon configurations, time enroute, etc. The fully embellished SHELL1.DBF is used to produce the printed copy of the daily shell.

Dialog

The dialog component of RSS required the most time and effort during system development. The man-machine interface is critical to system acceptance and considerable effort was invested toward that end. Enable's menus, macros, and windowing capability were the primary tools used to produce an effective dialog between the DSS and the user.

Menus

RSS has a total of 36 builder-developed menus that allow the user to move through the system, operate on the data, and produce printed output. The menus allow the user to enter data, make choices that move to a new menu, or execute a macro command. Most of the menus occupy the entire screen, but when a menu is called while the user is operating on data (e.g.

a monthly schedule), the menu will appear in a smaller window. The window is placed so that the menu does not obstruct the user's view of the data or interfere with the decision process.

Macros

A macro command consists of individual Enable commands strung together and executed sequentially with a single keystroke. Macros allow the user to repeatedly perform the same operation on data without entering multiple commands. This capability is incorporated in RSS to allow the user to execute complicated commands without needing in-depth knowledge of Enable.

Macro commands are found in three places in RSS. The largest macro commands are stored in their own file with the label ${X}.MCM, where X is a different letter for each command. These large macros perform lengthy and complicated operations that may require several minutes to execute.

A second place where macros are stored is in the menus from which they are executed. The menus have limited space for storage of macros so only the short, less complicated commands are able to reside in a menu.

The third location for macros is in a spreadsheet. These macros cannot operate across modules (i.e. from spreadsheet into database) so they have limited applicability. Each data file for the monthly calendar ([Month]CAL.SSF) has a spreadsheet macro that helps to import the standard week from the STNDRDS.SSF file in the proper order.

Macro commands are the key element in RSS's friendliness. It would be impossible for an unsophisticated user to perform all the required operations to produce monthly and daily schedules without using macros.

Windows

Enable has the capability to have up to eight windows open at one time on the screen. RSS takes advantage of this to give the user the ability to view several pieces of information at the same time to aid the decision making process. Windows are used extensively in the monthly phase to present data from two to three spreadsheet storage files at the same time. As previously mentioned, windows are also used to size and locate called menus to avoid data obstruction.

Model

The quantitative component of RSS is neither elaborate nor complicated. The nature of the scheduling problem is such that any solution that satisfies, or at least does not violate the numerous constraints is generally deemed satisfactory. An optimum scheduling solution is very difficult to define, and usually subject to argument and opinion. The role of the model component in RSS is to show the scheduler the effect of his decisions, and to allow him to "what if" the problem until a satisfactory solution is found.

In the monthly scheduling module of RSS numerous Enable functions are embedded in the spreadsheets. These functions serve to build the correct representation of a month, tally the number of hours and sorties for each type of sortie, and to display the scheduler's progress towards the monthly flying hour allocation.

The weekly shell module of RSS has numerous IF...THEN statements incorporated into the macro commands which embellish the basic sortie information transferred from the monthly schedule. Based on the content of the monthly schedule and the user's inputs, these conditional statements enter the appropriate values into fields in the SHELL1.DBF database which is used to produce the daily shell. These statements may be thought of as a rudimentary rule-based expert system.

Chapter V. discusses the conclusions and the recommendations that resulted from this research and development effort.

IV. Conclusions and Recommendations

In this research effort a decision support system for fighter squadron scheduling was designed, built, and implemented by using the adaptive design approach. This chapter presents the conclusions and recommendations from this project.

The main objective of this project was to exercise the methodology by developing a decision support system to tackle a real-world problem faced by a decision-maker in an operational environment. In this regard the effort was a success. RSS began as a conceptual approach to the scheduling problem faced in a fighter squadron, progressed through concept maps and storyboards to a set of requirements, and evolved through user-builder interaction into a viable decision aid. RSS is a functional tool for the squadron schedulers in the 89th TFS.

A noteworthy feature of this research was the application of the adaptive design approach to DSS development in an environment where the builder and the user were co-located. Many past DSS research projects have had notional users or had the builder acting as the user. In those projects that had actual users, the user was usually in another state, and in one case, on another continent. Without the ability to frequently exchange thoughts, ideas, requests, and complaints, these projects were generally limited to one iteration. RSS evolved through four iterations.

By applying the tools and techniques of adaptive design in a realistic setting with a genuine user, some of the strengths and weaknesses of the methodology became clear. As the DSS developed and evolved, so did the author's perception of the problem, the user, the hardware and software, and the methodology.

The Problem

The scheduling problem in the 89th TFS was decomposed into four phases of scheduling: yearly, monthly, weekly, and daily. The DSS developed includes the monthly and the weekly phases. The yearly phase was too structured for the DSS approach and was not included. The daily phase appeared to be an excellent candidate for a DSS, however time constraints were a limiting factor. Given more time, the next step would have been to expand into daily scheduling. This expansion of RSS would involve more risk however, due to the excellent "manual DSS" currently in use and also because of hardware limitations (e.g. small screen size).

The scheduling problem in the 89th TFS, although not as complex as that of many active USAF fighter squadrons, served well as a test subject. However, the nature of the problem led to a DSS with a weak model component. The scheduler did not really want to analyze or optimize; he wanted to satisfy. Thus the requirements specified by the user led to a system with a very basic model component.

> Recommendation - Small organizations are excellent opportunities for DSS applications. However, the designer/builder must carefully analyze the decision process to be supported to ensure that he can provide the appropriate models and data along with an effective dialog component. Some indicators for model and data components are: what quantitative techniques are currently in use, can results of analysis be presented graphically, is an optimum solution desired, and what data sources are used and how and how often are they updated.

The User

A critical factor in the success of any user-driven requirements determination process is the attitude and motivation of the user. In this project, the user was approached by the builder with the proposal to aid his decision process with a DSS. Although the user was open-minded and willing to experiment with a new approach to his job, there was a lack of motivation towards system expansion. This was manifested by the preponderance of user inputs in the area of man-machine interface for the current system, rather than requirements and direction for future system capabilities and features.

A much better environment for system evolution would have existed if the user had come to the builder with a problem or decision to be supported. In this case, the user would obviously have some concept of requirements and specifications. However, until knowledge and awareness of decision support reaches the "grass-roots" level, many potential applications for a DSS will go untapped.

> Recommendation - A short course in DSS basics and potential applications should be included in the training received by the small computer managers assigned to base organizations. These people could then become the points of contact for DSS development in small units.

The Hardware and Software

The selection of hardware and software for this project was driven by the organizational environment of the application. The main objective of this development effort was an operational system in a real-world environment. To achieve this objective, hardware currently in-place had to be used. The squadron could not be expected to invest heavily in computer hardware or software. For that reason, the DSS was limited to installation on a Z-248 PC with a hard drive. A major limiting feature of this configuration is the small screen size. More advanced workstations with large monitors and greater resolution would be a better choice.

The choice of software was similarly constrained. The desire to evolve the system through multiple iterations forced an early start on system development. The Smartware package by Informix Software appeared to be a good candidate for a DSS generator, but was not immediately available. It also had a much larger cost; a consideration if it were to be purchased by the 89th TFS.

Recommendation - If builders are to develop state of the art DSSs, then they need advanced hardware and software to do so. Even if this equipment is available in the academic setting, it very likely will not be available in the application environment. A possible solution might be a lend-lease type agreement in which organizations could be supplied with this equipment on a low-cost basis. If the DSS is of sufficient value to the supported unit, then they could expend the funds necessary to buy their own hardware and software.

The Methodology

Adaptive Design

Adaptive design works most effectively when the builder and the user are collocated and can interact face-to-face. Without frequent interaction, the requirements determination process becomes slow and fragmented. The DSS is slow to expand and may never reach the expectations of the user.

User trust and acceptance of the delivered system is greatly enhanced by a closer relationship between the builder and the user. In this project, as the builder and user developed a working relationship, barriers to trust and communication fell away. The result was more user knowledge of the system and trust in its results.

> Recommendation - The development of DSSs should take place in the user's organization. This would greatly increase the user-builder interaction and reduce the time required to evolve an effective DSS. The demand on user's time should not increase however, because much of the increased interaction would be initiated by the user.

Concept Maps and Storyboards

Concept maps and storyboards are viable tools of the adaptive design approach. The concept map produced after the initial meeting proved to be a real asset when determining the kernel system. Two minor problems were experienced with storyboards.

The first occurred when the builder's own knowledge and previous experience in scheduling led to some assumptions and projections on the part of the builder. The result was that features were included in the storyboard representation of the system that were not desired by the user. It is a credit to the storyboard concept, however, that these unwanted features were identified by the user prior to incorporation in the working system.

The second problem occurred because the storyboards were constructed before the builder had learned the capabilities and limitations of the software. As a result, the storyboards suggested features that could not in fact be incorporated into the system. The most outstanding example of this is the electronic hookbook facility.

The hookbook concept was to be implemented in RSS as a selection on each menu. The idea was to make the hookbook as accessible as possible to the user, so that whenever an idea for system improvement or expansion occurred, the thought could be logged for future reference. The database module was used to record and store user hookbook entries.

The problem arose when the user attempted to make an entry in the hookbook when a window containing a large spreadsheet was already open. This is the case whenever the user is working on the monthly schedule. The new window for the hookbook could not be opened due to an insufficient amount of RAM available.

In attempting to resolve this problem, the builder discovered that Enable is a very RAM hungry program. Although the documentation states that up to eight windows can be displayed at the same time, this is not possible if one or more of the windows requires a great amount of memory. The monthly schedule files are approximately 65 kilobytes in size.

Several attempts to solve the problem were made: 1) reduce the memory demand of Enable, 2) decrease the memory requirements for data files, and 3) use a virtual disk to run Enable. None of these attempts resulted in enough available RAM to run the hookbook facility as specified in the storyboards. The end result was that the hookbook became a stack of labeled 3" X 5" cards given to the user.

> Recommendation - The commitment to a particular DSS generator should be postponed until a bound can be placed on established and forecast system requirements. This should preclude the possibility of the DSS capabilities being overcome by an evolving set of system requirements. The designer/builder needs a background in computer science to have the substantial knowledge base necessary to choose the most effective hardware and software.

The Hookbook

The hookbook is the means by which the user makes inputs to the builder for system enhancement and expansion. Ideally, the user should make hookbook entries whenever he has an idea or thought which relates to future system capabilities.

In this DSS application the user did not make use of either the electronic hookbook (in its limited capacity) or the structured notecards. With one exception, every input the builder received from the user was passed verbally during one of the many visits to the 89th TFS. Verbal inputs to the builder may not be an obstacle to system expansion if the system is developed in the user's organization and the builder is easily accessible.

Another point of interest is that almost all of the inputs received concerned the man-machine interface of the system as implemented at the time of the input. This user concentrated mostly on the dialog portion of the DSS, with fewer requests for changes to the data and model components.

These findings are likely related to the fact that this user was recruited into the DSS project and did not initiate it himself. Although a willing participant, he could not be considered a highly motivated one.

The Rhino Scheduling System is a functional DSS at the small organization level. It assists a fighter squadron scheduler in his decision process in the monthly and weekly scheduling phases. This DSS is the result of a flexible methodology applied to a real problem with a genuine user. The author hopes that RSS can make a continuing contribution to the 89th TFS.

RSS User Manual

The purpose of this manual is to aid the first-time user in the operation and control of RSS, the Rhino Scheduling System. RSS is not difficult to operate, so after a few trial runs the new user should have little trouble. The best way to learn how to use RSS is to have the system up and running while you work through this manual.

For information on how to install or maintain this system, refer to the RSS Maintenance Guide.

Starting RSS

To start the system, simply type **RSS** (or **rss**) at the C: prompt and Enable will load, followed by the display of the Main Menu. RSS is an application of the software package Enable, produced by The Software Group.

The Main Menu

The main menu will appear as shown in Figure A-1. The options open to the user are: 1) work on a monthly schedule or functions related to monthly scheduling, 2) work on the weekly shells, and 3) quit the program and return to the DOS level.

Menu Features

The menus are the means by which the user moves through the system and operates on the data. There are two basic types of menu options for the user. The first involves making a choice between listed options (as in Figure A-1); the second involves entering data in a highlighted box (as in Figure A-4). Many of the screen displays in RSS have default menus which can be called up by using the **Shift/F10** keys simultaneously. The **Esc** key will remove the default menu.

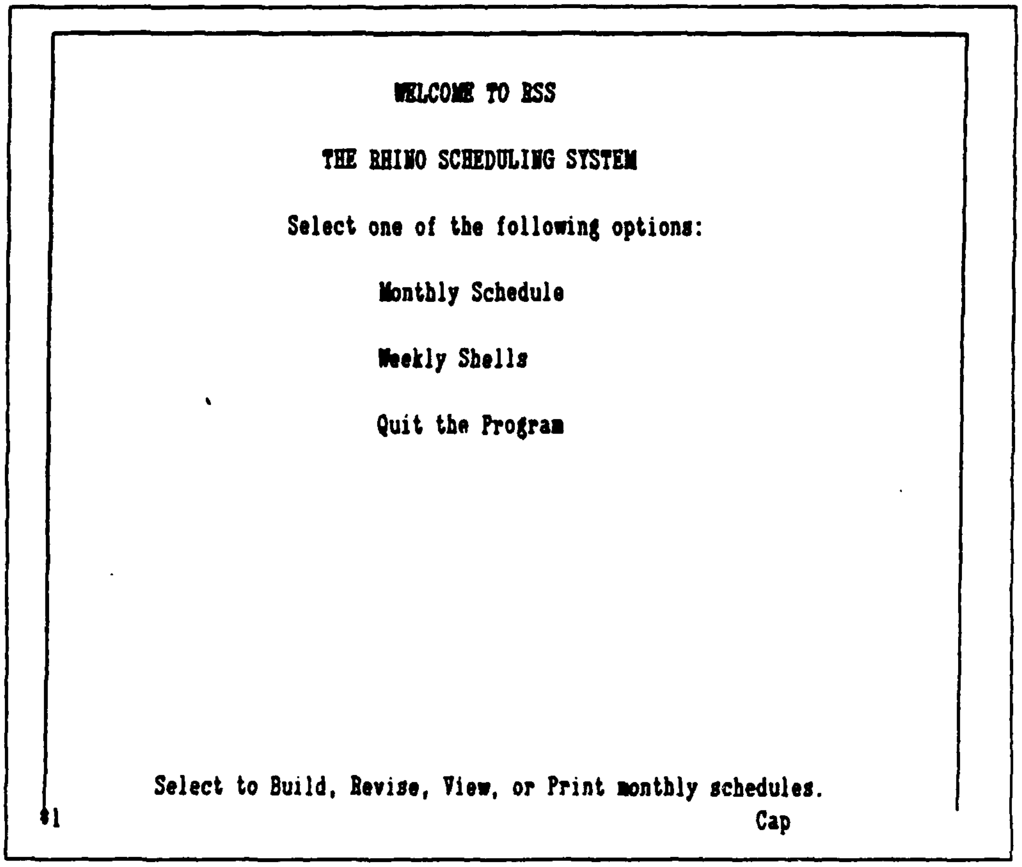

Figure A-1. Main Menu

Select Options

To make a choice between options listed on the screen, you can either: 1) use the cursor control keys (the arrow keys) to move down to your desired option and then press **Enter**, or 2) just press the key of the letter that is highlighted in the desired option (usually the first letter).

Enter Data

To enter data in a highlighted box, just type the characters from the keyboard. Follow the suggested format or you may get an error message in the information line. If you make an error, use the cursor control keys to back up and re-enter the data correctly. When you have the data correctly entered, hit the **Enter** key.

MONTHLY SCHEDULING OPTIONS

Select one of the following options:

Build a new month

Modify a month

View a month

Output products

Update memory

MAIN MENU

Select Build to start a new month from the beginning.

1 Cap

Figure A-2. Monthly Options Menu

Information Line

At the bottom of the menu there is an information line that will display a one line explanation of what the current menu choice does. (The current menu choice is the option that the cursor is highlighting.) For menus that call for the user to enter data, the information line will display an error message if the data is entered incorrectly.

Monthly Scheduling

Select *Monthly Schedule* at the *Main Menu* to begin the monthly scheduling process. This will bring up the Monthly Scheduling Options menu depicted in Figure A-2. Each option will be described in detail below.

BUILDING A NEW MONTH

Note: Building a new month will delete whatever is currently stored in that month. You may want to print a hardcopy of last year's month for your records prior to building a new schedule.

Build a new schedule?

Yes

MONTHLY OPTIONS

Select Yes to continue building a new month.

#1 Cap

Figure A-3. Building a New Month

Build a New Month

This option starts building a new month from an empty shell. If there is a schedule currently stored in that month, say from the same month last year, that information will be deleted so that you can start from a clean slate. When you select "Build a new month" the menu shown in Figure A-3 appears to explain this and to give you a chance to make an archive copy of last year's month.

If you select Yes from this menu, you will proceed to the menu depicted in Figure A-4 so that you can specify the month to be worked. If you select MONTHLY OPTIONS, you will be returned to the Monthly Options menu so that you can select Output products to make a copy of the month before you delete it.

SELECT MONTH

Which month will you be building?

Enter the month first, then the year.

Month:

Year:

Enter the month in a three letter abbreviation (e.g. OCT).

#1 Cap

Figure A-4. Select Month

Select Month

In this menu (Figure A-4), you specify the month and year to be scheduled. Use the standard military three-letter abbreviations for the month (recommend that you set CAPS LOCK on the keyboard while using RSS), and enter only the last two digits for the year (e.g. 89).

Programmed Hours

In this menu (Figure A-5), you enter the flying hours for the month called for by the Annual Flying Hour Program. Enter the figure in the data entry box in the format specified in the information line and hit **Enter**.

If you don't recall the number of hours, use the arrow keys to cursor down to View FHP and hit **Enter**. This will call up the Annual Flying Hour Program as stored in the computer. (See Update Memory for more

PROGRAMMED HOURS

How many hours does the Flying Hour Program call for
this month? If you have that figure, enter it below.

Hours:

If you don't have that figure at hand, cursor down to
View FHP to bring up the Annual Flying Hour Program.

View FHP

Note: Hit 'R' when done Viewing FHP.

Enter the hours to fly this month in the format XXX.X (e.g. 265.5).

#1 Cap

Figure A-5. Programmed Hours

information.) When you have made a mental note on the number of hours you need to fly, hit R or r to Return to the menu to enter the appropriate value.

Attrition Rate

This menu (Figure A-6) is very similar to the one above. In the same manner as the programmed hours, enter the appropriate attrition rate or select View Attrition to look at stored attrition data.

Local/Zulu Time

In this menu (Figure A-7) you can input the date for the change from Standard Time to Daylight Savings Time. Except for October and April, you should select No to proceed. If the month you are working has a time change, select Yes to set the effective date of that time change.

ATTRITION RATE

What attrition rate is required for this month? If you have the figure, enter it below.

Attrition Rate:

If you don't have that figure at hand, cursor down to View Attrition to view the *historical attrition* data.

View Attrition

Note: Hit 'R' when done viewing the attrition.

Enter the attrition rate for the month in the format X.XX

1 Cap

Figure A-6. Attrition Rate

Setting Time Change Date

This menu (Figure A-8) comes up if you selected Yes in the above menu. Enter the date that the time change becomes effective and hit **Enter**. You will not see the affects of this right away, since the monthly schedule is done in Local time. But the range and airspace requests are in Zulu time, so an incorrect or nonexistent time change date can cause problems with range times.

Working a Month

After selecting No in Figure A-7 or entering a date in Figure A-8, a macro command (a long string of individual Enable commands) will call up two 'windows' for you. This process takes about 80 seconds. The window at the top left of the screen contains the calendar month that you specified

LOCAL/ZULU TIME

Do you expect a change in Local Time this month?

Note: The change to/from Daylight Savings
Time should normally be a factor only
in October or April.

No

Yes

Select if there will NOT be a change to/from Standard/Daylight Savings.

#1 Cap

Figure A-7. Local/Zulu Time

earlier. The window at the right contains summary information for that month. Figure A-9 is a depiction of how that screen should look.

The Calendar Window

The left window will be called the calendar window, and is a spreadsheet file. What you see in that window is one day of the month, in this case the first day. You should enter data only under the columns marked Go, Type, *, T/O, and Remarks. Each line under these headings stands for a flight of aircraft. For example, line 6 in Figure A-10 shows a flight that will launch in the second go, that will fly air-to-air, will be a 2-ship, will takeoff at 1425 local, and will meet Tribe flight in the area.

You can enter up to 12 flights per day. In the Go column, you can indicate as many go's as desired, but will probably only use 1, 2, or 3 most

SETTING TIME CHANGE DATE

Enter the date that the time change takes effect.

Date:

Enter the date in numbers (e.g. 4 or 25).

#1 **Cap**

Figure A-8. Setting Time Change Date

of the time. In the Type column, use the abbreviations shown on the right window (i.e. AA, AG, XC, OB, RR, LL, or AAR). You can mix and match with AAR, e.g. AAR/AG. List the AAR first, and if it is night air refueling use AARN (not NAAR). In the # column, enter the number in the flight. For T/O, enter the takeoff time in Local military time. Note that a preceding zero will not appear (e.g. entering a 0900 time will show up as 900.) In the Remarks column you can enter up to 10 characters. Any more than that and they will be truncated.

If you have some sorties to be flown off station, and want them to appear on the monthly schedule but don't want them to show up on the daily shells or the monthly range and airspace requests, leave off the entry in the Go column. This way, the sorties will be counted for the month's sortie

```
C3:                                                              READY

   A B  C   D   E  F      G            A    B C D E  F   G H
  1MAY                              1Month: MAY
  2      Go  Type  #  T/O   Remarks  2
  3   Mon                            3 Type  ASD   #    Hours
  4    1                             4 AA   0.9 #    =   0.0 ERR %
  5                                  5 AG   1.1 #    =   0.0 ERR %
  6                                  6 XC   3.0 #    =   0.0
  7                                  7 OB   6.0 #    =   0.0
  8                                  8 RR   2.0 #    =   0.0
  9                                  9 LL   1.1 #    =   0.0
  10                                 10 AAR 1.5 #    =   0.0
  11                                 11
  12                                 12Scheduled: Sorties    0
  13                                 13           Hours    0.0
  14                                 14Programmed Hours:  333.0
                                     15Planned Attrition:  0.05
                                     16Hours to Schedule: 349.7
                                     17Hours Remaining:   349.7 233 A
                                     18                         127 G
                                     19Date Last Updated: APR 4
  Use 'Shift/F10' to call up Menu, 'Esc' to remove it.        Cap
```

Figure A-9. The Working Month

and hours count, but won't appear anywhere else. An example might be a mini-deployment to Miramar for DACT training.

<u>Moving Around in the Calendar Window</u>

The limited window size prevents you from seeing more than one day of the month at a time. Figure A-11 shows how the month is laid out in the spreadsheet. The upper left hand day is always the 1st of the month. There are seven days across the top of the calendar and four to five weeks down, depending on the month. The day of the date (i.e. whether the 1st is a Monday, Tuesday, etc.) is automatically set.

To move around in the month, use **Tab**, **Shift/Tab**, **PgDn**, and **PgUp** keys. The **Tab** key will move you to the right one day at a time. **Shift/Tab** will move you back to the left in the same way. **PgDn** will move you down one week

```
V3:                                                              READY

     V  W  X   Y     Z  AA      AB          A   B  C  D  E  F     G  H
  1                                       1Month: MAY
  2        Go  Type  #  T/O    Remarks    2
  3     Thu    AA    6  OPEN   MIRAMAR    3 Type  ASD   #    Hours
  4      4  1  AA    3   955              4 AA    0.9 * 140 = 126.0  60 %
  5         1  AG    4  1010              5 AG    1.1 *  94 = 103.4  40 %
  6         2  AA    2  1425    TRIBE     6 XC    3.0 *   0 =   0.0
  7         2 AAR/AA 2  1455    SABRE     7 OB    6.0 *   1 =   6.0
  8         2  AG    2  1510              8 RR    2.0 *   0 =   0.0
  9         3  AA    4  1825              9 LL    1.1 *   0 =   0.0
 10                                       10 AAR  1.5 *   8 =  12.0
 11                                       11
 12                                       12Scheduled: Sorties   243
 13                                       13           Hours   247.4
 14                                       14Programmed Hours:  265.0
                                          15Planned Attrition:  0.05
                                          16Hours to Schedule: 278.3
                                          17Hours Remaining:    30.9  21 A
                                          18                          11 G
                                          19Date Last Updated: APR 4
  Use 'Shift/F10' to call up Menu, 'Esc' to remove it.       Cap
```

Figure A-10. An Example Working Month

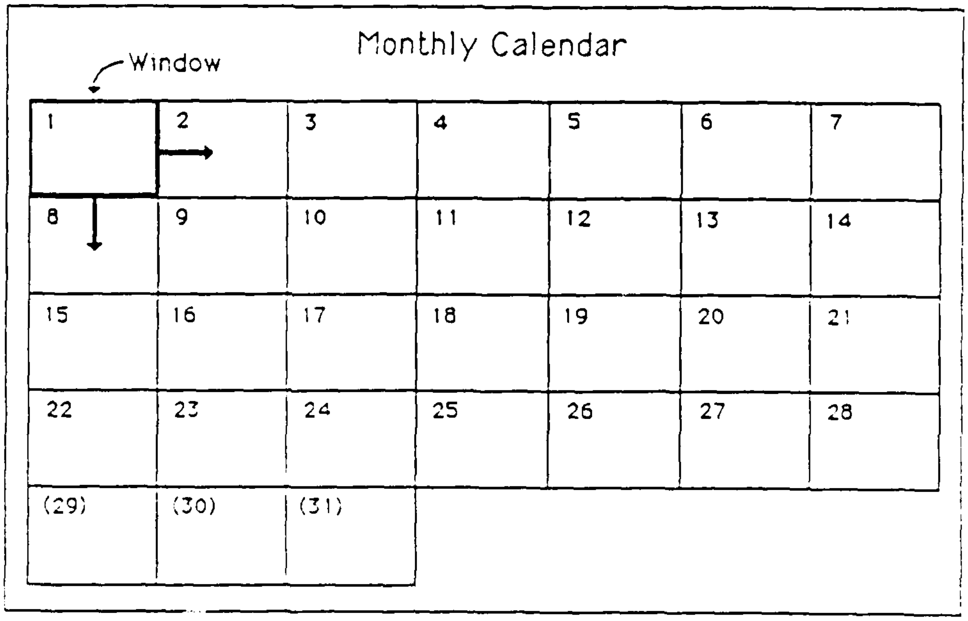

Figure A-11. The Monthly Calendar and Window at a time; **PgUp** moves you up one week. The arrow keys will also move you around, but take longer. The **Home** key will always return you to the 1st of the month. Refer to the end of this manual for some more hints on how to work in the calendar.

Don't stray outside of the boundaries of the month. There are formulas and commands that are stored to the right and below the month that should not be altered. If you think you have inadvertently changed a formula or deleted something important, STOP and QUIT WITHOUT SAVING. This may mean that you will lose some of your work, but will also keep you from mucking up the system.

The Summary Window

The right hand window contains the summary information for the month in the left hand window. In rows 4 to 10 the number of each type of sortie scheduled in the month is listed. These will initially be empty spaces, but when you 'Recalc' they will be counted and entered automatically. Each type of sortie has an ASD (average sortie duration) that is used to calculate the total hours flown for each type of sortie listed. The sortie durations can be changed by moving the cursor to the number you want to alter, entering the new number (look at the top left of the screen), and hitting **Enter**.

To the right of the AA and AG rows there will initially be an ERR%. After you 'Recalc' the first time, the ERR will be replaced by a number. This number represents the percentage of sorties that are AA vs. AG. This helps the scheduler monitor the percentage of sorties devoted to each major training activity.

In row 12 the system will show how many sorties have been actually scheduled for the month (use 'Recalc' to update this figure). This will be the sum of the figures in column D from row 4 to 10. Row 13 has a similar total for the hours in column F.

Row 14 will show the number of hours you entered in the menu labeled Programmed Hours. Row 15 shows the attrition rate you entered at the Attrition Rate menu. In row 16 the number of Hours to Schedule is shown. This figure takes the attrition into consideration so that after attrition the net hours should be the same as the programmed hours. In row 17 the Hours Remaining is displayed. This figure tells you how many hours you have left still to be scheduled in order to meet the goal shown in row 16. It is updated every time you Recalc.

The two numbers at the right of rows 17 and 18 are an estimate of how many air-to-air sorties (A) and air-to-ground sorties (G) should be scheduled to make up the deficit shown in the Hours Remaining block. In Figure A-10, with 30.9 hours remaining, the scheduler should try to schedule 23 AA sorties and 11 AG sorties to reduce Hours Remaining to zero. The number of hours of the AA and AG sorties follows a 60% and 40% split.

The last row in the summary window is for the scheduler to show when the last time he updated the information. Enter a date with the month first, then the day (e.g. Apr 5).

Options While Building a Month

The default menu is shown in Figure A-12 and is called up from the display shown in Figure A-9 by using the **Shift/F10** keys simultaneously. To remove this menu from the screen, hit the **Esc** key.

Standards

The option marked Standards will remove the left window and replace it with another that contains a calendar display for one week. This option

```
V3:                                                                    READY

     V  W  X    Y     Z  AA     AB            A    B   C   D  E    F      G  H
  1                                          1Month: MAY
  2        Go   Type  #  T/O    Remarks      2
  3     Thu      AA   6  OPEN   MIRAMAR      3 Type  ASD     #     Hours
  4       4 1    AA   3   955                4 AA    0.9 * 140 = 126.0  60 %
  5         1    AG   4  1010                5 AG    1.1 *  94 = 103.4  40 %
  6         2    AA   2  1425    TRIBE       6 XC    3.0 *   0 =   0.0
  7         2  AAR/AA 2  1455    SABRE       7 OB    6.0 *   1 =   6.0
  8         2    AG   2  1510                8 RR    2.0 *   0 =   0.0
  9         3    AA   4  1825                9 LL    1.1 *   0 =   0.0
 10                                         10 AAR   1.5 *   8 =  12.0
 11                                         11

                 OPTIONS AVAILABLE WHILE BUILDING A MONTH

     Standards        Update          (1=)Recalc        (2=)Save then Quit

     Left Window      Right Window       Quit          Remember: Esc
                                                                 Shift/F10
          Select to review/update/transfer Rhino Standard Sorties.
 Use 'Shift/F10' to call up Menu, 'Esc' to remove it.        Cap
```

Figure A-12. Monthly Options

was developed in recognition that some events are flown every week at the same time. You can fill this week with any standard events that will hold for the month, then Transfer it to the month you are working. This prevents the repetitive typing of the same events for each week.

After the Standards window is up, use **Shift/F10** to bring up the default menu for the standards option. The screen will look like Figure A-13. If you want to modify the standard events, do so and then **Save** them. If you don't want to modify or transfer the standards, just **Return** to the previous screen display. If you select **Transfer**, it will take about 1 minute to transfer the standards to the working month and then copy them into every week.

```
A2:  MON                                                        READY


     A    B    C     D  E      F            A    B  C  D  E  F    G  H
 1        Go   Type  #  T/O    Remarks     1Month: MAY
 2  MON                                     2
 3                                          3 Type  ASD    #    Hours
 4                                          4  AA   0.9 * 140 = 126.0  60 %
 5                                          5  AG   1.1 *  94 = 103.4  40 %
 6                                          6  XC   3.0 *   0 =   0.0
 7                                          7  OB   6.0 *   1 =   6.0
 8                                          8  RR   2.0 *   0 =   0.0
 9                                          9  LL   1.1 *   0 =   0.0
10                                         10  AAR  1.5 *   8 =  12.0
11                                         11

 The Standard schedule should be reviewed for accuracy prior to selecting
 Transfer.  If necessary, update the standard events for one week and Save.
 When the Standard schedule is accurate, Transfer it to your working
 schedule.  Select Return to return to working month WITHOUT the Standards.
                                                          Remember: Esc
          Save       Transfer       Return                          Shift/F10
   Save will store in memory the Rhino standard events now on the screen.
 Use 'Shift/F10' to call up Menu, 'Esc' to remove it        Cap
```

Figure A-13. Standards and Default Menu

At this step, the system will occasionally lock up and cease all operation. If nothing happens after 1 minute, and a little 'mac' appears in the information line, then the system has locked up. To correct this, you will have to exit RSS and start over. Do this by hitting the **Ctrl-Alt-Del** keys simultaneously, and then typing **RSS** at the C: prompt again to restart. If this doesn't work, turn off the power to the system, then power up and restart RSS.

Update

The Update option leads to another menu which appears in Figure A-14. In this menu, you can update either the Programmed Hours or Planned Attrition as shown in the Summary Window.

```
V3:                                                                  READY

    V  W   X     Y      Z  AA        AB          A    B  C  D  E   F     G  H
  1                                            1Month: MAY
  2        Go   Type    #  T/O     Remarks      2
  3    Thu       AA     6  OPEN    MIRAMAR      3 Type  ASD    #      Hours
  4       4  1   AA     3   955                 4  AA   0.9 * 140 = 126.0  60 %
  5          1   AG     4  1010                 5  AG   1.1 *  94 = 103.4  40 %
  6          2   AA     2  1425     TRIBE       6  XC   3.0 *   0 =   0.0
  7          2 AAR/AA   2  1455     SABRE       7  OB   6.0 *   1 =   6.0
  8          2   AG     2  1510                 8  RR   2.0 *   0 =   0.0
  9          3   AA     4  1825                 9  LL   1.1 *   0 =   0.0
 10                                            10  AAR  1.5 *   8 =  12.0
 11                                            11
 12                                            12Scheduled: Sorties   243

                             UPDATE OPTIONS
                                                  Remember: Esc
                                                            Shift/F10
          Update Hours to Fly          Update Attrition Rate

          View Flying Hour Program and position cursor for update.
#2   C:\ENSCHED\MAYCAL.SSF                                 Cap      BM107
```

Figure A-14. Update Options

Selecting Update Hours to Fly calls up the Annual Flying Hour Program display and places it in a small window at the top of the screen. Use the **PgDn** and **PgUp** keys to move up and down in the window to find the month of interest. Use **Shift/F10** to call up the default menu for this screen. The screen will then look like Figure A-15. Select "OK" to remove the FHP from the screen and return to the working month display. The cursor will be placed in the Summary window in the cell for Programmed Hours so that you can enter a new figure if desired. If you don't want to enter a new Hours figure, call up the default menu with **Shift/F10** for other options.

Selecting Update Attrition Rate will call up the Historical Attrition Rate display. As with the Hours to Fly, use **PgUp** and **PgDn** to move around, and **Shift/F10** to call up the default menu. Figure A-16 shows the screen

```
D6:  (F,1)  +B3/3                                                    READY


      A        B        C        D       E      A    B  C  D  E  F    G  H
 1  89th TFS Annual Flying Hour Program        Month: MAY
 2                                Hours
 31st Quarter   778                    FY1989   Type  ASD   #    Hours
 4                      October   259.3          AA   0.9 * 140 = 126.0  60 %
 5                     November   259.3          AG   1.1 *  94 = 103.4  40 %
 6                     December   259.3          XC   3.0 *   0 =   0.0
                                                 OB   6.0 *   1 =   6.0
 This window is presented for info            8  RR   2.0 *   0 =   0.0
 only.  It is to be used as a                 9  LL   1.1 *   0 =   0.0
 reference for making minor changes          10  AAR  1.5 *   8 =  12.0
 to the hours figure for one month.          11
 If major changes to the Annual              12Scheduled: Sorties    243
 Flying Hour Program are required,           13           Hours    247.4
 do so via the Main Menu.                    14Programmed Hours:   265.0
               Remember: Esc                 15Planned Attrition:  0.05
        OK                Shift/F10          16Hours to Schedule: 278.3
                                             17Hours Remaining:     30.9  21 A
 Returns & places cursor on hours.           18                           11 G
                                             19Date Last Updated: APR 4
 Hit 'Shift/F10' to call up Menu, 'Esc' to remove it.        Cap
```

Figure A-15. Updating Hours

display. Selecting OK will remove the Attrition Rate display and return the cursor to the Planned Attrition cell in the Summary window. Again, enter a new attrition rate and/or use **Shift/F10** to call up the options menu.

The intent of the Update option is to give the user the ability to juggle the programmed hours and attrition rate in the working month (not in the Annual FHP or Historical Attrition Rate displays). In this way, you can experiment and see the effects of different hours and attrition rates on the month's summary sheet. Fine tuning of a monthly schedule within the context of the entire year is possible.

<u>Recalc</u>

The option shown as (1=)Recalc will count the number of sorties of each type in the calendar window (left side), and show the results in the summary

K5: DEC 88 READY

	A	B	C	D	E	F	G	H	I	J	K
1		906th TFG Monthly Attrition Information									
2			Sorties		Weather		Total		Attrition		
3	Month/#Yrs		Scheduled		Cancels		Cancels		Rate		
4	JAN/4		989		140		219		0.14		As of:
5	FEB/4		957		215		273		0.22		DEC 88
6	MAR/4		1373		80		160		0.06		

This window is presented for info only. It is to be used as a reference for making minor changes in one months attrition rate. If you want to make a major change in the Historical Attrition Table, do so via the Main Menu. Remember: Esc Shift/F10

OK

Returns and places cursor on attrition

8	RR	2.0 *	0 =	0.0	
9	LL	1.1 *	0 =	0.0	
10	AAR	1.5 *	8 =	12.0	
11					
12	Scheduled:	Sorties		243	
13		Hours		247.4	
14	Programmed Hours:			265.0	
15	Planned Attrition:			0.05	
16	Hours to Schedule:			278.3	
17	Hours Remaining:			30.9	21 A
18					11 G
19	Date Last Updated:			APR 4	

Hit 'Shift/F10' to call up Menu, 'Esc' to remove it. Cap

Figure A-16. Updating Attrition

window (right side). You should periodically Recalc while working on a monthly calendar to see the effect of the sorties scheduled. Use the cursor keys and **Enter**, or simply type the number 1 to select this option.

Save Then Quit

This option will save all the inputs and changes made to both windows, and then remove these windows and return you to the monthly options window shown in Figure A-2. Use the cursor keys and **Enter**, or simply type the number 2 to select this option.

Left Window

This option moves the cursor into the left window so that you can enter sorties into the monthly calendar.

Right Window

This option moves the cursor into the right window so that you can change sortie durations, hours to fly, the attrition rate, or enter a date in the Date Last Updated cell.

Quit

Selecting Quit does NOT save any changes; it removes the calendar and the summary windows, and returns you to the monthly options menu. This gives you the option of experimenting or "what if'ing" a schedule without altering the work already accomplished. You should also use Quit if you think you have deleted or altered a formula stored in a cell that performs a calculation or function.

Modify a Month

The second option in the Monthly Scheduling Options menu is Modify a Month. This option brings up the same two-window display as Build a New Month, but does not erase the schedule contained in these windows. Modify a Month should be used whenever you want to complete a month that you may not have finished, or when you need to change or adjust a month that was completed earlier. When you select Modify a Month, the menu shown in Figure A-17 comes up, giving you the opportunity to make a hard copy of the month before you change it.

Select Month

Selecting Continue brings up a data entry menu in which you specify the month and year of interest. You also have the option of returning to the Monthly Options menu.

MODIFYING A MONTH

In modifying a month, it is assumed that the month has previously been completed, but now must be updated due to a change in manning, hours, weather, etc. When the month requested comes up on the screen, use the menu (Shift/F10) to update the flying hours or attrition as needed. Then adjust the schedule as necessary. You may want to print a copy of the old month before saving any changes so that you have an archive copy.

Select Continue below to bring up the month to be modified. Select Make Copy to go back to the MONTHLY OPTIONS menu to print a copy before making changes.

Continue

Make Copy

Select to call up the desired month for editing.

Cap

Figure A-17. Modifying a Month

Local/Zulu Time

Once you enter the month and year, the system calls up a menu similar to that shown in Figure A-7, and you have the option of changing the date for the change to/from Daylight Savings Time. Selecting No activates the macro command that calls up the same two-window screen that was used to Build a New Month. This macro takes about 30 seconds to execute. Selecting Yes brings up the screen that allows you to specify the time-change date (similar to Figure A-8). Once you have entered the appropriate date, the macro is executed and the two-window screen is called.

Working a Month

After the two-window screen is called and the working month calendar and summary sheet are presented, the default menu and all further operations are exactly the same as when you Build a New Month. The only difference is that instead of starting with a blank schedule as with a New Month, you begin wherever you left off previously.

View a Month

This option is provided so that you can look at the monthly calendar using the full screen of the computer monitor. The window with the monthly sortie summary is not called, so you will be able to see two days at a time as well as a portion of the second week. Use this option only to view a month, not to modify it.

When you select View a Month, the next menu to appear will be the Select a Month menu, much like the one shown in Figure A-4. Enter the month and year and a macro command will call up a full screen display. Use the same keys to move around this calendar as before.

When finished viewing the month, call up the default menu with **Shift/F10**. This menu appears at the bottom of the full screen display, as shown in Figure A-18, and gives you the option of returning to the Monthly Options menu with Return, or specifying a new month to view with Another. Selecting Another will call the Select a Month menu As with the other default menus, hitting **Esc** will remove the menu from the screen.

Output Products

Selecting Output Products from the Monthly Options menu will call a menu similar to that in Figure A-4. Use this menu to specify the month and year of the schedule you want to print. You also have the option of returning to the Monthly Options menu.

Once you have specified the month and date, the next menu called is labeled Output Products and is shown in Figure A-19. The first option

C68: READY

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	MAY													
2			Go	Type	#	T/O	Remarks			Go	Type	#	T/O	Remarks
51		Mon	1	OB	1	1000	ARGON		Tue	1	AA	2	955	SABRE
52		29							30	1	AA	2	1025	SLUFF
53										2	AA	2	1355	STEEL
54										2	AA	2	1425	SLUFF
55										3	AA	4	1825	
56														
57														
58														
59														
60														
61														
62														
63														

If you are finished viewing this month, select an option below.
Remember: Esc
Return Another Shift/F10
Select Return to go back to the Monthly Options Menu.
Use 'Shift/F10' to bring up the Menu. Cap

Figure A-18. Viewing a Month and Default Menu

listed will return you to the Monthly Options menu. The other options are covered below.

Monthly Schedule

This selection produces a copy of the monthly schedule on standard 8.5" by 11.0" paper. It uses compressed print, yet still requires three pieces of paper to complete and takes about 90 seconds. You will have to cut and paste and make a reduced copy on a copy machine to get it all on one sheet of paper. After printing, you will be returned back to the Output Products menu.

Requests for Bomb Ranges/Airspace

This selection will produce two documents: a bomb range request and a airspace request. The source for each is the monthly schedule. The

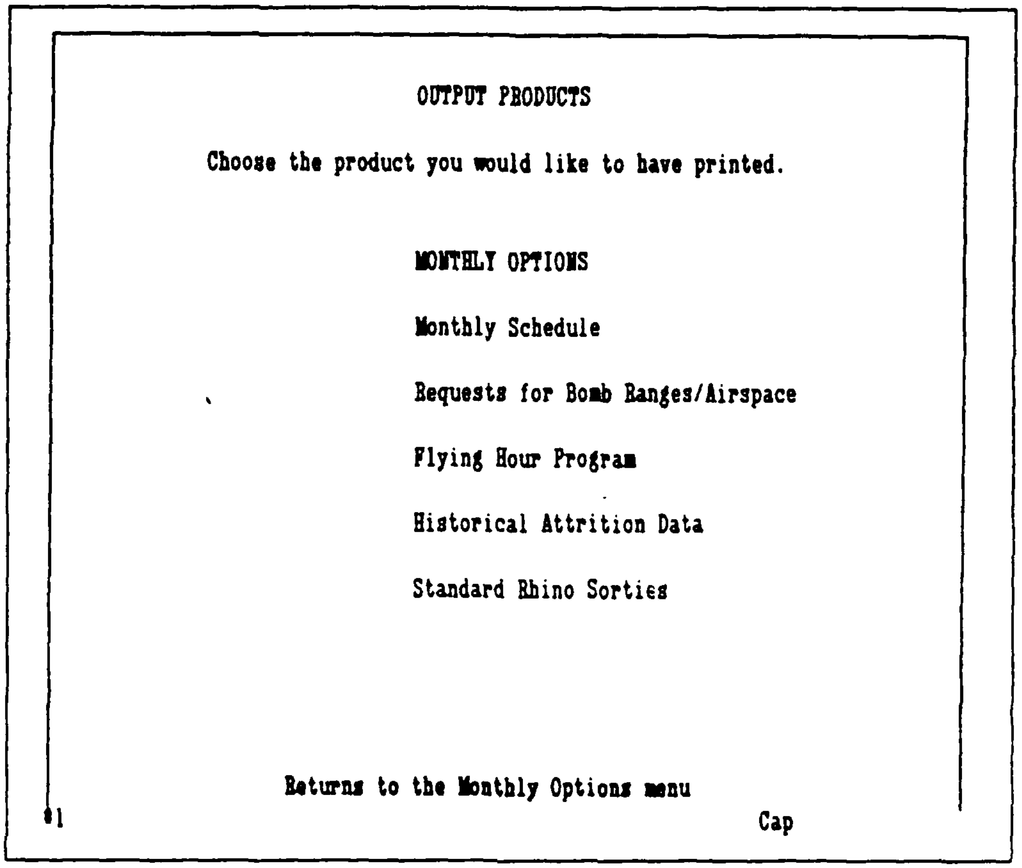

Figure A-19. Output Products Menu

requested range times will be in Zulu time and on the airspace request the flight callsign and remarks section will be printed also. Producing these documents is an involved process and takes about 3.5 minutes. They are printed on standard size paper in compressed print. After printing, you will be returned back to the Output Products menu.

Flying Hour Program

This option will produce a printed copy of the Annual Flying Hour Program as stored in the computer. Standard size paper is used, with standard print, and it takes less than 30 seconds.

Historical Attrition Data

This option prints a copy of the attrition data as stored in memory. It uses standard size paper, standard print, and takes less than 30 seconds.

OPTIONS FOR UPDATING MEMORY

Select one of the following options for Updating:

Standard Rhino Events

Flying Hour Program

Attrition Data

MONTHLY OPTIONS

Select to Update the sorties that are semi-standard each week.

#1 Cap

Figure A-20. Updating Memory Options

Standard Rhino Sorties

Selecting this option produces a copy of the standard sorties currently held in memory. It uses standard size paper, compressed print, and takes less than 30 seconds.

These last three options all return you to the Output Products menu. To quit printing output, select Monthly Options to return to that menu.

Update Memory

Selecting Update Memory from the Monthly Options menu will call up the menu shown in Figure A-20. This option should be used to maintain the currency of the three memory items shown in the menu. The Annual Flying Hour Program and the Historical Attrition Data should be updated whenever new information is received from higher headquarters or maintenance. The

A2: MON READY

	A	B	C	D	E	F	H	I	J	K	L	M
1		Go	Type	#	T/O	Remarks		Go	Type	#	T/O	Remarks
2	MON						TUE					
3								1	AA	2	1025	SLUFF
4								2	AA	2	1355	STEEL
5												
6								3	AA	2	1825	
7												
8												
9												
10												
11												
12												
13												

When you have updated the 'Standard' Rhino schedule, select Save to store the changes in memory, and then return to the Update Memory menu. If you don't want to save any changes, select Return to go back to the Update Memory menu. Remember: Esc

Save Return Shift/F10

Select to Save the updates made to the Rhino Standards, then return.

Use 'Shift/F10' to call up Menu. Cap

Figure A-21. Updating Standard Events

Standard Rhino Events can be updated prior to working on a monthly schedule, or while building a schedule (via the Standards menu option).

Standard Rhino Events

This option calls up a screen display of the standard events. Use the **Shift/F10** keys to call up the default menu as shown in Figure A-21. Remember that the **Esc** key removes the default menu so you can continue to update the events. Update the standards and Save or simply Return to the Update Memory without saving.

Flying Hour Program

Select this option to bring up a display of the Annual Flying Hour Program. The **Shift/F10** and **Esc** keys will call up and remove the default menu, which has the same options as above. This is shown is Figure A-22.

```
D20:  Jan 89
                                                    When you have updated the
                                                    Flying Hour Program and are
       A        B        C         D        E       sure that it is accurate,
 1   89th TFS Annual Flying Hour Program            select Save.  Select Return to
 2                               Hours              go back to the previous menu
 31st Quarter   778                        FY1989   WITHOUT saving.
 4                       October   259.3                      Remember: Esc
 5                       November  259.3                                Shift/F10
 6                       December  259.3                 Save
 72nd Quarter   878
 8                       January   292.7                 Return
 9                       February  292.7
 10                      March     292.7
 113rd Quarter  1079
 12                      April     359.7
 13                      May       359.7
 14                      June      359.7
 154th Quarter  1175                                 Select to Save the changes
 16                      July      391.7             you have made in the FHP and
 17                      August    391.7             return to the Update Memory
 18                      September 391.7             menu.
 19 Year Total  3910
 20                      Updated:  Jan 89
 Use 'Shift/F10' to call up Menu.                                Cap
```

Figure A-22. Updating Flying Hour Program

<u>Attrition Data</u>

This option allows you to update the Historical Attrition Data. The default menu performs just as above and is shown in Figure A-23.

<u>Monthly Options</u>

This menu selection will return you to the Monthly Options menu.

<u>Main Menu</u>

This is the last option on the Monthly Options menu and will return you to the Main Menu that is displayed when the system first starts. From the Main Menu you can proceed into the Weekly Shells phase of scheduling.

K5: DEC 88 READY

	A	B	C	D	E	F	G	H	I	J	K
1		906th TFG Monthly Attrition Information									
2			Sorties		Weather		Total		Attrition		
3	Month/#Yrs		Scheduled		Cancels		Cancels		Rate		
4	JAN/4		989		140		219		0.14		As of:
5	FEB/4		957		215		273		0.22		DEC 88
6	MAR/4		1373		80		160		0.06		
7	APR/4		1162		73		160		0.06		
8	MAY/4		1158		58		185		0.05		
9	JUN/4		1210		62		193		0.05		
10	JUL/3		831		37		103		0.04		
11	AUG/3		774		11		65		0.01		
12	SEP/3		665		47		66		0.07		

When you have updated the attrition data and are sure that the data as shown is accurate, select Save. If you want to return to the last menu without saving, select Return.

Remember: Esc

Save Return Shift/F10

Select to Save any changes to Attrition Data, and return to Update Menu.

Use 'Shift/F10' to call up Menu. Cap

Figure A-23. Updating Historical Attrition

Weekly Shells

The weekly shells option starts the process that takes information stored in a monthly calendar and produces a shell for each day that is specified by the user. When you select this option at the Main Menu the screen shown in Figure A-24 is called. You are given the option of proceeding with shell production, going to the Monthly Options menu to review or update the relevant monthly schedule(s), or returning to the Main Menu. If you are confident that the monthly schedule is accurate, select Continue to proceed to the display shown in Figure A-25.

WEEKLY SHELL PRODUCTION

The weekly shells are produced from the information that is currently stored in the monthly schedule. Therefore, to ensure that the most current information is used, make sure that the schedule for the source month is accurate and up-to-date. If the week overlaps into a second month, review BOTH months for accuracy.

If the monthly schedules are accurate, select Continue to proceed with shell production. If you want to want to review the month(s), select MONTHLY OPTIONS.

Continue

MONTHLY OPTIONS

MAIN MENU

Select to continue the shell building process.

$! Cap

Figure A-24. Weekly Shell Production

DEFAULTS

Defaults exist in the system for fuel and
weapons configurations, and for sortie durations.

Do you want to Change these default values, or
would you like to Accept them?

Accept Defaults

Change Defaults

Select to accept system defaults for fuel, wpns, ETE.

#1 Cap

Figure A-25. Default Option Menu

Defaults

When RSS takes the information stored in the monthly schedule and develops a daily shell, much more information is added. Most of this information is a result of defaults that you can selectively accept or change. These defaults determine brief times, land times, fuel loads weapon configurations, ranges, and range times. To view and/or change these default settings select Change Defaults. If the default settings are satisfactory, select Accept Defaults.

Default Selection

If you select Change Defaults the screen in Figure A-26 will appear. This display instructs you to enter a value for all options displayed, even if you want to change only one. Select Yes to proceed, No to skip the

DEFAULT SELECTION

Numerous default values are used to produce the daily shells. You will be presented with the category of values, the default values built into the system, and then asked if you want to change these values. If you elect to change any of the defaults in one category, you must enter all the values in that category, even if only one has changed.

Once you have changed the desired default values, it would be best to View a shell (rather than Print it) to see if you got the desired results.

Do you wish to proceed to change defaults?

Yes

No

Select to change the defaults for the daily shell production.

#1 Cap

Figure A-26. Default Selection Menu

default display (and by implication, accept the system defaults) and proceed to set the date of interest. If you select Yes, the next menu will be called.

Fuel Load Defaults

This menu, Figure A-27, presents you with the system default for each sortie type and gives you the opportunity to change one or more. If all of the default fuel codes are satisfactory and no change is required, select No to proceed to the next menu. If you wish to change any of the codes, select Yes and then enter the desired fuel configuration code for each sortie type. Remember that you must enter all the codes, even if only one is to change. When you enter the last code, you will proceed to the next menu, shown in Figure A-28.

DEFAULT CATEGORY: FUEL CONFIGURATIONS

Given the default fuel configuration codes below, do you wish to change them?

No Yes

SORTIE TYPE	DEFAULT CODE	YOUR CODE
AA	F	
AG	F	
XC	F	
OB	F	
RR	F	
LL	F	
AAR	F	

Remember to enter ALL the codes, even if changing only one.

#1 Cap

Figure A-27. Fuel Configuration Defaults

Weapon Load Defaults

This menu, shown in Figure A-28, is similar to the previous one: select No if the presented codes are satisfactory, Yes to change one or more. Enter the codes as before, and you will proceed.

Sortie Duration Defaults

This menu is as shown in Figure A-29. Select your desired option and proceed.

Briefing Time Default

This screen is presented in Figure A-30. If you want to change from the default 1+45 prior to takeoff, select Yes and enter the new figure. Make sure that you enter the value in minutes. For example, to brief 2 hours prior to takeoff, set 120 minutes in the data entry space. This is the

DEFAULT CATEGORY: WEAPONS CONFIGURATION

Given the default weapons configuration codes below, do you wish to change them?

No Yes

SORTIE TYPE	DEFAULT CODE	YOUR CODE
BFM	1PY	
ACT	1PY	
DACT	1MY	
AG	3	
XC	98	
RR	1	

Remember to enter ALL the codes even if changing only one.

#1 Cap

Figure A-28. Weapons Configuration Defaults

last of the default review/change menus. The next screen to appear requests the date for which shells are to be produced.

Month and Date

This menu is called after the defaults are Accepted in Figure A-25, if you decline to proceed in Figure A-26, or when you have accepted or changed the brief time default in Figure A-30. The screen display will be that shown in Figure A-31. You should enter the month and date for the day for which shells are to be produced. You also have the option to Quit this phase and return to the Main Menu.

DEFAULT CATEGORY: SORTIE DURATION TIMES

Given the default sortie duration times below, do you wish to change them?

No Yes

SORTIE TYPE	DEFAULT DURATION	YOUR DURATION
AA	0.9	
AG	1.1	
XC	3.0	
OB	6.0	
RR	2.0	
LL	1.1	
AAR	1.5	

Remember to enter ALL the durations even if changing only one.

#I Cap

Figure A-29. Sortie Durations Defaults

Updating the Database

This menu, displayed in Figure A-32, will start the appropriate macro command which will manipulate the data and produce a set of shells for the date specified above. Each option on this menu is explained.

First

The menu option First should be selected if you are producing shells from the specified month for the first time in this session at the computer. The macro command executed by First is a long one, taking almost 3 minutes to finish. The reason for this long delay is due to the time required to update the database for that month. The most recent information is desired, so RSS goes back to the monthly calendar and brings that entire month into a database accessed by the daily shell production routine. The macro does

```
DEFAULT CATEGORY:  FLIGHT BRIEFING TIME

Given the standard time allowed for flight briefings, do
you wish to change it?
                              No       Yes


DEFAULT: Flight Briefing begins:

                         105 Minutes (1+45) before T/O

              Change to:

                              Minutes before T/O.


1                                          Cap
```

Figure A-30. Briefing Time Default

this, then samples this data for the date you specified, incorporates the defaults into a daily shell, then presents you with the menu shown in Figure A-33.

Subsequent

After you have produced one day's shell from a month, it is not necessary to go through the lengthy process of updating the database with that month again. So for subsequent days from the same month, choose the option labeled Subsequent. This menu option uses the month's information brought into the database by the above macro, incorporates the defaults, produces a shell, and calls the menu in Figure A-33. It takes less time to execute, less than 1 minute.

MONTH AND DATE

Specify the month and the date for the shell you want to produce.

Month:

Date:

Quit

Enter the month in a 3-letter abbreviation (e.g. OCT).

Figure A-31. Setting Month and Date

New Month

If you are working on a week that overlaps into a new month, you will need to update the database with that month before you produce a shell. In the same way as First, New Month will access the month, bring it into the database, incorporate the defaults, and constructs a shell. It also requires about 3 minutes. Once you have done this, you can use Subsequent the next time around.

Options with Shells

This menu will appear when a daily shell has been prepared and is ready to be viewed, edited, or printed. Figure A-33 shows the menu; its options are explained below.

UPDATING THE DATABASE

If this is the first day's shells to be printed during this session select First to update the database that is the source for the shells.

If this is not the first day's shells, and you are still working the same month, select Subsequent to save time. Do NOT select Subsequent if you had to change months (e.g. the next day is May 1).

If you had to change months, select New Month.

First

Subsequent

New Month

Select to update database for first set/month of shells.
Cap

Figure A-32. Updating the Database

View

You should use this option to view the shell on the screen before it is printed. Selecting View will bring up the display shown in Figure A-34. Due to screen size you can only see a portion of what will be printed. By using the cursor control keys you can move right, left, and down. It is not possible to move up, so you may have to View the same shell a couple of times to see everything you want to. The shell displayed will be exactly like the one that will be printed, so review it carefully. When you are done viewing the shell, hit Esc to get back to the Options menu shown in Figure A-33.

```
                    OPTIONS WITH SHELLS

You have four options at this point.
1) View on the screen the shell for the date selected;
2) Edit the shell for this date;
3) Print a hard copy of the shell;
   Note: Ensure that the printer has large (14") paper loaded.
4) Quit back to the Home Menu.

Options 1 and 2 let you View or Edit the selected day's
shell as many times as you like.  Option 3 will Print a
copy and then return you to the point where you can
specify a new Date and Month to produce a shell.

                View   (Hit 'Esc' to Return)

                Edit   (Use 'F4' key to edit, 'Esc' to return.)

                Print

                Quit
   Select to View the shell for the date specified.
                                                  Cap
```

Figure A-33. Options with Shells

Edit

If you want to change something on the shell, select Edit. This will call up the screen shown in Figure A-35. The top line shows abbreviations for the columns below them and are, in order:

NU - number in flight,
BR - brief time,
TO - takeoff time,
LD - land time,
FL - fuel configuration,
WPN - weapon configuration,
MSN - mission,
ETE - enroute time,
CS - callsign (Rhino XX),
AR - ARCT time (if any; 0 if none),
RNG - range,
RST - range start time,
RND - range end time, and
RMKS - remarks (20 characters now available).

Each line shown is for the flight lead only. To change something, you must move through the display with the cursor keys until the value you want to change is located at the top left position on the screen. Once you are positioned correctly, hit the **F4** key to bring up a small window at the bottom of the screen. Type in the new value and hit **Enter**. You can change as many values as you want to: when you are finished, hit **Esc**. You will be returned to the Shell Options menu.

After you have edited the daily shell, it is a good idea to View it again just to be sure you have it the way you want it before it's printed. The final step is printing the shell.

WEEKLY/DAILY FLYING SCHEDULE (OPERATIONS)-FIGHTER/ATTACK SOF:

LINE NO.	ACFT NO.	SCHED BRF	SCHED T/O	SCHED LND	FUEL CONFIG	WPNS CONFIG	MISSION	ETE	ATD	ATA	AT
1		825	1010	1120	F	3	AG	1.1			
2		.	.	.	.	.	.	.			
3		.	.	.	.	.	.	.			
4		.	.	.	.	.	.	.			
5		1240	1425	1515	F	1MY	DACT	0.9			

Arrow keys to scroll sideways, PGDN or RETURN to continue, or ESC to cancel.

Hit Esc to Return to OPTIONS Menu. Cap

Figure A-34. The View Option Screen Display

NU	..BR	..TO	..LD	FL	WPN...	MSN......	ETE	CS	..AR	RNG...	.RST	.RND	RMKS......
4	825	1010	1120	F	3	AG	1.1	11	0	ATT	1440	1500	
2	1240	1425	1515	F	1MY	DACT	0.9	21	0	R5503	1830	1900	TRIBE
2	1310	1455	1625	F		AAR	1.5	31	0	R5503	1900	2000	SABRE
4	1640	1825	1915	F	1PY	AA	0.9	41	0	R5503	2230	2300	

Use cursor keys to page through display. Press ESC when done.
Use cursor keys, then F4 key to EDIT, then hit Enter; Esc forCap

Figure A-35. The Edit Option Screen Display

Print

When you are satisfied that the daily shell reflects the correct events, select Print to get a hard copy. The printer should have 11" by 14" paper installed to print the shells. After the shell is printed, you will be returned to the menu depicted in Figure A-31 so that you can specify another date and print another day's shell.

If You're Having Problems

There are several possible ways for problems to occur with RSS. One has already been mentioned: the system locks up on you. The best way to handle a locked keyboard is to hit **Ctrl-Alt-Del** simultaneously. This will reboot the entire system, and you can restart RSS as usual. However, sometimes even this won't work and you may have to turn off the computer and restart.

Both of these fixes may result in some of your work being lost, but there isn't much you can do about that. The system doesn't lock up often; usually when you attempt to Transfer the Rhino Standard Events to the working month.

Another possible problem could be getting "lost" in the system. If a menu choice or some sequence of keyboard entries takes you to an unfamiliar screen display, try hitting **Esc** until you get back to something familiar. It may be that you get back to the screen display shown in Figure A-36. This is Enable's main menu. Select Return to DOS, then restart with **RSS**.

A third problem is a little more serious. If you happen to delete or modify formulas or commands embedded in a spreadsheet, you may begin to get erroneous or nonsensical results. This is most likely to happen when you

```
                         ENABLE (tm)

          Select an option with the cursor and [↵]

   Press [Esc] if you change your mind and [F1] if you need help.

     Use System        Help          MCM        Return to DOS

 Word Processing   Spreadsheet/Graphics   Telecom   DBMS/Graphics




 1                                                 Cap
```

Figure A-36. Enable's Main Menu

are working on a monthly schedule, since each monthly calendar has numerous commands and formulas stored in various spreadsheet cells. The best way to prevent this problem is to stay within the monthly calendar borders. However, if you do accidentally alter something, the easiest fix is to reload the file for that month from the master disks. Refer to the RSS Maintenance Manual for information on how to do this. This chore should be attempted only by someone familiar with computers, and if possible, familiar with Enable.

Hints on Enable

If you can learn some basic Enable features and commands, you can work a little faster and easier with RSS, especially in the monthly calendar spreadsheet environment. The best place to start would be with Enable commands to move and copy cells in spreadsheets. Start by reviewing the Enable Spreadsheet User Manual and then experiment by copying and moving sortie information. Remember that you can do just about anything you want as long as you don't Save anything. Just Quit, and there is no harm done.

The Enable default menu is called by hitting **F10**, vice **Shift/F10** for RSS. Getting to know some of the commands presented when you call Enable's default menu would also be helpful. Enable will present you with Help anytime you hit **F1**, so if the Enable manual isn't handy, try that.

Good luck with RSS.

RSS Maintenance Manual

The purpose of this manual is to assist in the installation and maintenance of the Rhino Scheduling System. It is intended to be used by a person familiar with computers, and if possible, familiar with Enable.

Enable is a product of The Software Group, and consists of spreadsheet, database, word processing, graphics, and telecommunication modules. RSS makes use of the spreadsheet and database components; the word processor was used to design the report forms used to produce printed output.

This guide will describe the hardware and software requirements, how to install RSS, and some features of its spreadsheets, databases, menus, macros, and data files.

Hardware and Software Requirements

RSS was designed on a Zenith Z-248 computer with a 20 Megabyte hard disk and 640K of RAM. The printer should be an Alps P2000G and requires 8.5" by 11.0" paper as well as 11.0" by 14.0" paper. RSS does not require any extra RAM or a math co-processor to run.

Enable 2.15 was used to develop RSS. The system will not work correctly with version 1.15, but should execute with the newer Enable OA. Enable itself uses a great deal of RAM, so to ensure that enough is left over for RSS to work properly, you may want to minimize the number of terminate and stay resident (TSR) programs that are called from the autoexec.bat file. A disk manager, automenu, or pc tools type of utility may require just enough RAM to limit RSS's operation at certain points. This may occur when two or more windows are open with large spreadsheets active.

Installing RSS

The first step is to ensure that Enable is correctly installed. Unless you changed the default, Enable created its own directory called ENSKB. The batch file listed below assumes that Enable is in ENSKB; if it is named differently, just substitute the correct name in place of ENSKB.

Create a directory named ENSCHED. Then copy all the files from the program disks into this directory. You will need about 1.6 megabytes of disk space available for all the data and program files. Most of this space is required by the large spreadsheet files used for each month. Because the spreadsheet files are so large, a generic month named "MON" is used to store a copy of these spreadsheets for just one month. These generic files should be reproduced and modified for each month; the directions are given below. It means a little more work to install, but fewer master disks are needed to store RSS.

The RSS Batch File

A batch file is used to start the Rhino Scheduling System with the simple command "RSS". The batch file listed below is labeled RSS.BAT and can be kept in your BIN directory or anywhere that is included in a PATH command.

```
cd enskb
enable (,,,,c:\ensched),@9,n
cd\
```

This file starts enable, specifies the directory containing the data and program files (ensched), and executes an initial macro command ($(9).MCM) that brings up the opening menu.

The Program Files

Program files are those that work with Enable to present specific applications of Enable components and features. These applications are the menus, macros, windows, report forms, and spreadsheet and database files that make up RSS. Program files can be categorized by function and differentiated by their file extensions.

Menu Files

Menu files contain the specifications for each menu that is presented to the RSS user. There are 36 menu files and each has the extension ".MNU". The menu file names were chosen to suggest menu function; related menu files have a numeric suffix. For example, the Main menu has the file name MAIN.MNU. The file names of the menus for building a month are labeled BLDMON1.MNU, BLDMON2.MNU, BLDMON3.MNU, and so on. Figure B-1 shows a complete listing of the menu files used in RSS.

MAIN.MNU	VUMON1.MNU	UPMEM4.MNU
MODMON.MNU	VUMON2.MNU	SHELL1.MNU
BLDMON1.MNU	OUTPUT1.MNU	SHELL2.MNU
BLDMON2.MNU	OUTPUT2.MNU	SHELL3.MNU
BLDMON3.MNU	WORKMON.MNU	SHELL4.MNU
BLDMON4.MNU	WRKUPDT.MNU	DFLTS1.MNU
BLDMON5.MNU	UPHRS.MNU	DFLTS2.MNU
BLDMON6.MNU	UPATRT.MNU	DFLTS3.MNU
MODMON1.MNU	STNDRDS.MNU	DFLTS4.MNU
MODMON2.MNU	UPMEM1.MNU	DFLTS5.MNU
MODMON3.MNU	UPMEM2.MHU	DFLTS6.MNU
MODMON4.MNU	UPMEM3.MNU	VUSHELL1.MNU

Figure B-1. RSS Menus

Macro Files

Enable uses macro commands in three forms. The longest macros are stored in their own files, and these files have the extension ".MCM", or in one case, ".DBM". The name for each macro file is only one letter long, so it is difficult to match name with function. Figure B-2 lists these files and their related functions.

```
${9}.MCM - Brings up Main.MNU at initial start-up
${A}.MCM - Produces daily shell for FIRST time
${B}.MCM - Calls FHP, then Returns to BLDMON3.MNU
${C}.MCM - Calculates sortie summary from calendar
${J}.MCM - Calls calendar and summary displays to MOD
           no Local/Zulu time change
${K}.MCM - Calls calendar and summary displays to MOD
           with Local/Zulu time change
${O}.MCM - Calls and clears calendar and summary for BLD
           with Local/Zulu time change
${P}.MCM - Calls and clears calendar and summary for BLD
           no Local/Zulu time change
${R}.MCM - Transfers MONCAL.SSF to MONDB.SSF to MONDAT.DBF
           and produces AA & AG range requests
${S}.MCM - Produces daily shell for SUBSEQUENT time
${V}.MCM - Calls attrition data, then Returns to BLDMON4.MNU
${N}.DBM - Prints a daily shell
```

Figure B-2. RSS Macro files

A second form of macro command can be stored within the menus from which they are executed. The storage space available is limited, so these macros are generally short in length. Macros stored within menus do not have names and can only be accessed from the menu. Figure B-3 lists those menus that have embedded macros and describes the function of each macro.

The third form of macro is designed to work within a spreadsheet, and is used in RSS to help copy standard events throughout a month. Each monthly calendar has one of these macros stored in cells C90 through C107.

Report Forms

Three report forms are used in RSS and have the extension '.RPT'. Each report form was constructed with the word processing module of Enable, and is used by the database to output information in a useable form. The report forms use the 'dot' commands of the database to manipulate the data into the correct form.

The first form is named AGREQST.RPT and contains the form which is used by the database to print the bomb range request. Another form is the AAREQST.RPT form. This is used to print the airspace requests for

MAIN.MNU	- 'Quit the Program': Quits RSS and returns to DOS
WORKMON.MNU	- 'Standards': Calls and sizes STNDRDS.SSF 'Save then Quit': Saves MONCAL.SSF and MON.SSF, then Quits to MONOPT.MNU 'Left Window': Moves cursor to left window 'Right Window': Moves cursor to right window 'Quit': Closes MONCAL.SSF and MON.SSF and goes to MONOPT.MNU
WRKUPDT.MNU	- 'Update Hours': Calls & sizes FHP.SSF 'Update Attrition': Calls & sizes ATTRIT.SSF
UPATRT.MNU	- 'OK': Returns to MON.SSF
UPHRS.MNU	- 'OK': Returns to MON.SSF
STNDRDS.MNU	- 'Save': Saves any changes to STNDRDS.SSF 'Transfer': Transfers Standard events to MONCAL.SSF 'Return': Return to MONCAL.SSF without transferring standards
VUMON.MNU	- 'Year': after entering year, calls MONCAL.SSF to view
VUMON2.MNU	- 'Return': Quits viewing, calls MONOPT.MNU 'Another': Quits viewing month, calls VUMON1.MNU to select another month

Figure B-3. Menus Containing Macros

OUTPUT2.MNU	-	'Monthly Schedule': Prints MONCAL.SSF and MON.SSF 'Flying Hour Program': Prints FHP.SSF 'Historical Attrition Data': Prints ATTRIT.SSF 'Standard Rhino Sorties': Prints STNDRDS.SSF
UPMEM1.MNU	-	'Standard...': Calls STNDRDS.SSF for updating 'Flying...': Calls FHP.SSF for updating 'Attrition...': Calls ATTRIT.SSF for updating
UPMEM2.MNU	-	'Save': Saves changes to STNDRDS.SSF & returns 'Return': Quits without saving
UPMEM3.MNU	-	same as UPMEM2.MNU for FHP.SSF
UPMEM4.MNU	-	same as UPMEM2.MNU for ATTRIT.SSF
SHELL2.MNU	-	'Accept Defaults': Inputs all defaults via DFLTS6.MNU hidden menu
DFLTS1.MNU	-	'No': Inputs all defaults via DFLTS6.MNU
DFLTS2.MNU	-	'No': Inputs all fuel configuration defaults
DFLTS3.MNU	-	'No': Inputs all weapon load defaults
DFLTS4.MNU	-	'No': Inputs all sortie duration defaults
DFLTS5.MNU	-	'No': Inputs flight briefing default

Figure B-3. Menus Containing Macros (Cont.)

air-to-air training. The third form is used to print the daily shell and is named SHELL.RPT.

Data Transfer Form

The form named WINDOW2.CTF is used to transfer sortie data from the spreadsheet module to the database module. The database module is used to manipulate and format sortie data into printed range requests, airspace requests and daily shells. To make the transfer from spreadsheet to database, a transfer form is needed. The form WINDOW2.CTF aids the transfer of data from the spreadsheet [MON]DB.SSF to [MON].DAT, where MON is the three-letter abbreviation for the particular month in question.

The Data Files

Database Files

RSS has two basic classes of database file. The first is named SHELL1.*BF and holds the data that is used to construct daily shells. (The * is either a D or $ as explained below.) Only one SHELL1 exists; its contents are deleted and then reloaded every time a new day's shell is produced.

The second class of database file is [MON]DAT.*BF (where MON is the month). There are thirteen of these files; one for each month and one generic month file. These files contain the basic sortie data that is transferred from the spreadsheet data file for each month. They are used to produce the range and airspace requests and as a source for the SHELL1 file.

Database Definition Files

A data definition file is one of two files that are associated with a particular database and has the extension "$BF". The second is the database data file itself and is discussed below.

Each database file must be defined before any data can be stored in it. A database definition file has the extension ".$BF" and contains the parameters for every field that makes up the database.

Database Data Files

The actual data is stored in files with the extension ".DBF". As mentioned previously, there are thirteen of these files for [MON]DAT.DBF and one for SHELL1.DBF.

Spreadsheet Files

Each month in the year is supported by three spreadsheet files. One is labeled [MON]CAL.SSF, another [MON].SSF, and the third [MON]DB.SSF. The first is a calendar and has numerous functions and formulas embedded within it, and as a result can require up to 70 kilobytes of memory. The second file is used to summarize the information contained in the first. The last file is used to transform the calendar file's information into the proper format for transfer to the database.

Replicating the Spreadsheet Files

When you initially install RSS you will have to copy the MON files twelve times to set up an entire year of monthly calendar shells, summary windows, and transfer spreadsheets. The best way to do this is through Enable commands. If you start RSS by using the batch command, wait until the Main Menu is displayed and then hit Esc to bring up Enable's main menu. Select Use System, Spreadsheet/Graphics, and Revise. At the filename prompt, type a "?" to get a listing of all files with a .SSF extension. Move the cursor to the MONCAL.SSF file, hit "C" for copy, then Enter twice, then change the MON to JAN (or whichever month you want to create) and hit Enter again. A new file named JANCAL.SSF should appear in the file listing. In the same way, make a copy for each month with each generic MONCAL.SSF, each MON.SSF, and each MONDB.SSF file. When you are done you should have 36 more files in the listing than before.

[MON]CAL.SSF Files

Now that you have a CAL file for each month, you will need to call each one up and edit it slightly to customize it for the particular month it represents. Get back to the Enable main menu, and select Use System, Spreadsheet/Graphics, Revise, and then enter JANCAL (for example).

When the file comes up you will want to replace MONTH NAME in cell A1 with January. To do so, select F10, Worksheet, Titles, then Clear. Use the cursor keys to get to cell A1, hit the F4 key, look to the upper left corner of the screen, change the name to January, and hit Enter. Cursor down two rows, then select F10, Worksheet, Titles, and Horizontal.

Now, cursor right to cell AY6. This cell contains the figure /01/01. The first 01 represents the month. If you are working on January, then this is correct. If you were working on February, then you would want to change this to /02/01. For March it would be /03/01, and so on for the rest of the year. To change it, put the cursor on AY6, select the F4 key, change the cell as required, and hit Enter. Now hit the Home key, and save the file by selecting F10, Save, Accept, hit Enter, and Replace. Quit by selecting F10, Quit, and Yes. This will return you to Enable's main menu.

This procedure should be repeated for every month of the year.

[MON]DB.SSF Files

Each [MON]DB.SSF file will also have to be customized. Call up the first one, say FEBDB.SSF. Move the cursor to cell J4 and enter (in CAPS) the three-letter abbreviation for the month you are working on. Again, for January there is already a JAN in J4. But for February, enter a FEB. Now, select F10, Worksheet, Copy, and hit Enter. Then type exactly "J5..J375" and hit Enter. The FEB should be copied down the J column all the way to row 375. Now move the cursor to cell K7 and change the /01/01 just as you did above for MONCAL.SSF. Again, hit the Home key, and then select F10, Save, Accept, hit Enter, and Replace. Then you can quit with F10, Quit, and Yes and repeat for every month.

[MON].SSF Files

These files do not need to be modified after they have been replicated with the MON.SSF file.

Maintaining RSS

A detailed description of the content and logic of the menus and macros used in RSS is beyond the scope of this appendix. However, much could be learned by browsing through the system by a person who knows computers and knows Enable. Modification of the system should not be attempted unless that expertise is available.

Correcting System Faults

The easiest way to fix faults in the system is to reload the appropriate program files and see if that corrects the problem. If possible, try to narrow down the problem to a particular menu, macro, spreadsheet, or database. Then copy that particular file into the data directory (ENSCHED) and see if the problem is cleared up. If you can't determine where the problem is, copy all the program files into the data directory from the master disks.

If you want to get into the inner workings of the menus and macros, you should be familiar with the Enable manuals on these subjects. To review the contents of menus, and the commands behind the menu choices call up Enable's main menu. Select MCM, Tools, Menus, and Revise. To get a list of all *.MNU files, enter a "?". When you select a file, an explanatory screen is called. Strike any key to continue and the menu's top level will appear. This is what you see when the menu is actually called. To see what each menu choice or data entry has behind it, place the cursor on the gray rectangular space prior to each menu choice or data entry spot, then hit Shift/F9. You can move through the menu form by hitting Enter to move forward or Esc to move backwards. If you move forward to a message line, hit Shift/F9 to continue moving forwards. When you are finished reviewing the menu's interior and are back at the top level of the menu, select F10, hit Enter, then Q for Quit.

If you want to take a look at a macro with the .MCM or .DBM extension, start at the main menu of Enable. Select MCM, Macros, List, then All. Move the cursor to the macro you want to review and hit Enter. The file that comes up is basically a special word processing file with numerous Enable commands in it. The Enable System Overview Guide has macro codes summarized in an Appendix and is an excellent reference. After you are done looking at the file, select F10, Quit, and Yes.

The macros embedded in the menus can be reviewed by diving down into the interior of the menus as described above. If the macro is fairly short in length, it may be displayed within the menu. Otherwise, the menu will refer to the external files accessed via the MCM.

The macro embedded in the calendar spreadsheet can be reviewed by calling any of the [MON]CAL.SSF files and moving down to cell C90. This macro takes the Rhino Standard Events that were copied to the spreadsheet by another macro, tests for which day of the week the 1st of the month falls on, then copies the standard week into the calendar in the correct order.

The RSS F-16 Conversion Guide is located in Appendix C. This manual details the parts of RSS that need to be updated for the conversion of the 89th TFS from the F-4D to the F-16.

RSS F-16 Conversion Guide

The purpose of this manual is to guide the modification of the Rhino Scheduling System when the 89th TFS undergoes conversion from the F-4D to the F-16. Alteration of RSS should only be attempted by a person familiar with computers, and if possible, familiar with Enable.

Monthly Scheduling

Changing Mission Types

In its F-4 version, RSS does monthly scheduling in the following missions: AA - air-to-air, AG - air-to-ground, XC - cross-country, OB - out & back, RR - round robin, LL - low level, and AAR (or AARN) - air refueling (or night AR). If these need to be changed, use the following procedure.

Starting with the generic month MON, call up the MON.SSF file (the monthly summary file). Simply go down column A and enter whatever sortie codes are required (maximum of 7). When finished, save the file. Repeat this procedure for every month in the year.

Now call up the MONCAL.SSF file. Use the tab key to move right to column AZ. The next seven columns (from AZ to BF) are used to tally the number of each sortie type for each day of the month. The formulas stored in each cell use the @IF function to test for the presence of a particular sortie type (AA, AG, etc.) in the "Type" column of each day. If that sortie type is present, then the number in the flight (#) will be added to the total for that type of sortie. Each formula has seven @IF functions; one for each day of the week.

For example, cell AZ3 has the following formula stored in it:

```
@IF($D3='AA',$E3,0) + @IF($K3='AA',$L3,0)+ @IF($R3='AA',$S3,0) + ...
```

To change the sortie types, go to the column that applies. Say for example that you want to change from LL to INS for Instrument sorties. Go to cell BE3, hit F4, and change every LL (total of seven) to a INS, then hit Enter. Now, to copy this change throughout the month, ensure that the cursor is still on BE3, then hit F10, Worksheet, Copy, then hit Enter, type exactly BE4..BE62 and hit enter. The changes should now have been made all the way down through row 62 (the end of the calendar).

Make all the required changes to sortie types, then save the file. As with the MON.SSF file, you will have to change each month in the same way. You will also have to change sortie types in two macros, as described below.

Changing Sortie Durations

Changing the sortie durations in the MON.SSF file (the summary sheet) is not complicated. Call up the MON.SSF file, go to column B and make the appropriate entries for each sortie type. When the sortie durations are as desired, save the file. Make the same changes for each month in the year.

Adjusting Sortie Percentages

When RSS was built, the 89th was flying approximately 60% of its sorties as AA, and 40% as AG. The summary sheet (MON.SSF) uses this to guide the scheduler's adjustment of the calendar to meet the hours goal. The formulas in cells G17 and G18 took the Hours Remaining figure from F17 and converted it into the number of AA and AG sorties needed to be added/cut from the schedule to reduce F17 to zero.

If you want to change the 60/40 proportion, go to cell G17; the formula will be (F17*.6)/B4. Change the .6 to whatever percentage of sorties that you want to be AA. Do the same for cell G18 (.4 in this case). Save the file when finished. Repeat for all months.

Weekly Shells

Modifying Menus

Modifications to update the system to F-16 requirements will involve changing the defaults to reflect F16 fuel loads, weapons loads, sortie durations and sortie types. The first step is to review the menus for defaults to see what needs to be changed. Look at Figures A-27, 28, & 29 in Appendix A and note the changes required to sortie types, fuel and weapons codes and sortie durations.

The next step is to modify the menus. From the Enable main menu hit MCM, Tools, Menus, Revise and enter DFLTS2.MNU to modify the Fuel Configuration codes. Read the intro screen and hit any key to get the menu up. The first two columns show the menu types and the default codes. Modify each as required. Then go up to the menu selection 'No' and put the cursor on the gray rectangle and hit Shift/F9. Hit Enter until you reach the macro stored in the menu. The string of seven 'F~' 's are where the old defaults are entered, and are arranged from top to bottom. Change the appropriate 'F' to whatever you changed on the menu screen. For example, if you changed the XC fuel code to D, change the third F to a D, because XC is the third code from the top. After you have made the changes to the macro, continue to hit Enter until the menu comes up. Save the menu with Alt/F10 and Save; then quit by hitting F10, Enter, Quit. In a similar fashion, modify DFLTS3.MNU and DFLTS4.MNU.

Now you must modify the menus that allow you to accept all the defaults. To do so, call up SHELL2.MNU first. Put the cursor on the gray area before the Accept menu choice and hit Shift/F9. Hit Enter until arriving at the macro inside the menu. Notice the strings of F's, then the weapons codes, then the sortie durations. Change the appropriate codes, making sure to leave the ~ after each and the 2, 4, or 6 at the beginning of each string. After the changes have been made, hit Enter (or Shift/F9 for message line) until the menu comes up, then save with Alt/F10 and quit as above. Follow the same procedure with the menu labeled DFLTS1.MNU and the menu selection "No".

Modifying Macros

Two macros control the use of the defaults in the production of the shells. They are $(A).MCM and $(S).MCM and correspond to the "First" and "Subsequent" selections respectively on the menu depicted in Figure A-32 in Appendix A. Each menu treats the defaults the same, so changes made to one macro should be made in exactly the same way to the other. It would be better to start with $(S).MCM since it takes less time to execute, and will be easier to ops check after any changes.

To call up the macro for modification, start at the Enable main menu and hit MCM, Macros, Revise, MCM, and then S for the "Subsequent" macro. The best thing to do is make a printed copy of the macro, make your proposed changes in pencil, then enter them into the computer. To make a printed copy, just hit Alt/F2.

Modifying Sortie Types

If you have changed any of the sortie types that appear in the monthly schedule, or that are in any of the menus that specify defaults, then the macros must be modified as well. Again, using the macro $(S).MCM as the first to be modified, go to line 34 of the macro. (Note that the macro starts its first line at line 2.) Lines 34, 39, 42, 45, 48, 51, and 54 each have the sortie type enclosed in quotes. Change the sortie types as required, making sure that they are still enclosed in quotes. You probably won't want to change from AA, AG or AAR, since these are basic to any squadron. Make the same changes at lines 72 and 75, and also at lines 111, 114, 117, and 120. Save the changes by hitting F10, Save, Accept. Quit with F10, Quit, Yes.

Once you have made the changes and saved the file, restart RSS and check to see if the macro performs as required. Make up a day with an example of each sortie type so that all defaults are tested. Correct any problems and retest. When the macro performs as desired, make the exact same changes to $(A).MCM. Note that the line numbers will be different for the two macros.

Modifying Default Ranges

RSS uses R5503 as the default for AA airspace and Atterbury (ATT) as the bomb range default. If a change is desired to the R5503 default, go to line 85 in the $(S).MCM macro and replace R5503 with whatever is desired. Do the same at line 102. If a change to the ATT default is desired, go to line 94 and make the required change. Be sure to use quotes and save the changes. Test the results as above, then duplicate the changes in $(A).MCM.

Bibliography

Andriole, Stephen J. and others. Storyboarding for C^2 Systems Design: A Combat Support System Case Study. Paper presented at the Fifth Annual Workshop on C^2 Decision Aiding, Kansas City, October, 1987.

Davis, Michael W. Applied Decision Support. Englewood Cliffs, NJ: Prentice-Hall Inc, 1988.

Kopf, Thomas J. Design of an Aircrew Scheduling Decision Aid for the 6916th Electronic Security Squadron. MS Thesis, AFIT/GST/ENS/87M-9. School of Engineering, Air Force Institute of Technology (AU), WrightPatterson AFB OH, June 1987 (AD-A185 362).

Novak, Joseph D. and D. Bob Gowin. Learning How To Learn. Cambridge: Cambridge University Press, 1984.

Schneider, Maj William L. Adaptive Design in Building a Decision Support System for AWACS Programmed Flying Training. MS Thesis, AFIT/GST/ENS/87M-14. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 1987.

Simon, Herbert A. The New Science of Management Decision. New York: Harper and Row, 1960.

Sprague, Ralph H. and Eric D. Carlson. Building Effective Decision Support Systems. Englewood Cliffs, NJ: Prentice-Hall, 1982.

Trapp, Capt Paul E. and Capt Jeffrey W. Grechanik. Design Evolution of a Fighter Training Scheduling Decision Support System. MS Thesis, AFIT/GST/ENS/87M-6. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 1987 (AD-A180 265).

Valusek, LtCol John R. "Adaptive Design of DSSs: A User Perspective," DSS-88 Transactions, Eighth International Conference on Decision Support Systems, Boston MA, June 6-9, 1988. Providence, RI: The Institute of Management Sciences.

Valusek, Maj John R. and Dennis G. Fryback. "Information Requirements Determination: Obstacles Within, Among, Between Participants," Proceedings of the Twenty-First Annual Computer Personnel Research Conference, May 2-3, 1985. Association for Computing Machinery (ISBN# 0897911563).

Young, Lawrence F. Decision Support and Idea Processing Systems. Dubuque, Iowa: Wm. C Brown Publishers, 1989.

Vita

James Howard Jeter [redacted] He attended the University of Florida in Gainesville, Florida from 1974 to 1976, and received the Associate of Arts degree. He studied at The Florida State University, in Tallahassee, Florida from 1977 until he graduated in August 1978 with a Bachelor of Science degree in Meteorology. He entered USAF Officer Training School and was commissioned in February 1979. His first assignment was as an Air Weather Service forecaster at Ft. Rucker, Alabama. In May 1981 he began Undergraduate Navigator Training at Mather AFB, California, followed by Fighter Lead-In Training at Holloman AFB, New Mexico in early 1982. In May 1982 he entered the F-4 Replacement Training Unit (RTU) at Homestead AFB, FL. In December 1982 he reported to the 39th TFS at George AFB, California and flew as Weapons Systems Officer in the F-4E. He was reassigned in June 1984 to the 512th TFS at Ramstein AB, Germany. His next assignment came in January 1986 to the 81st TFS, Spangdahlem AB, Germany. In the 81st he was a daily scheduler and then served as Chief of Squadron Scheduling. In August of 1987 he entered the School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, Ohio. In March 1989 he was selected for membership in Omega Rho, the operations research honor society.

[redacted]

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

2b. DECLASSIFICATION / DOWNGRADING SCHEDULE

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

4. PERFORMING ORGANIZATION REPORT NUMBER(S): AFIT/GST/ENS/89J-3

5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION: School of Engineering

6b. OFFICE SYMBOL *(If applicable)*: AFIT/ENS

7a. NAME OF MONITORING ORGANIZATION

6c. ADDRESS *(City, State, and ZIP Code)*: Air Force Institute of Technology
Wright-Patterson AFB, Ohio 45433

7b. ADDRESS *(City, State, and ZIP Code)*

8a. NAME OF FUNDING / SPONSORING ORGANIZATION

8b. OFFICE SYMBOL *(If applicable)*

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c. ADDRESS *(City, State, and ZIP Code)*

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO	WORK UNIT ACCESSION NO.

11. TITLE *(Include Security Classification)*: ON-SITE ADAPTIVE DESIGN OF A DECISION SUPPORT SYSTEM FOR FIGHTER SQUADRON SCHEDULING (U)

12. PERSONAL AUTHOR(S): James H. Jeter, B.S., Capt, USAF

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT *(Year, Month, Day)*	15. PAGE COUNT
MS Thesis	FROM ______ TO ______	1989 May	115

16. SUPPLEMENTARY NOTATION

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*: Decision Support System, Adaptive Design, Scheduling, Fighter Squadron Scheduling

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

Thesis Advisor: John R. Valusek, PhD, Lt Col, USAF
Assistant Professor of Operations Research

(See reverse for Abstract)

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

21. ABSTRACT SECURITY CLASSIFICATION: UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE *(Include Area Code)*	22c. OFFICE SYMBOL
John R. Valusek, Lt Col, USAF	(513) 255-2549	AFIT/ENS

DD Form 1473, JUN 86 *Previous editions are obsolete.*

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

Abstract

The objective of this research project was to develop a decision support system (DSS) for the scheduler in a USAF tactical fighter squadron. A DSS supports the cognitive or mental processes of judgement and choice. The decision process supported in this project is the planning and programming of annual flying hours into monthly, weekly, and daily schedules. A kernel (or base) system was established, then extended and expanded with multiple iterations of the adaptive design approach. The adaptive design approach allowed the user to set system requirements in a gradual and interactive manner.

This thesis used as its subject the 89th Tactical Fighter Squadron located on Wright-Patterson AFB, OH. The close proximity of the 89th TFS and the Air Force Institute of Technology removed a common barrier to effective adaptive design: separation of the user and the builder. Through direct and frequent interaction, the 89th scheduler and the builder were able to develop and evolve a system that met the requirements of the user.

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/GST/ENS/89J-3	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION School of Engineering 6b. OFFICE SYMBOL (If applicable) AFIT/ENS	7a. NAME OF MONITORING ORGANIZATION
6c. ADDRESS (City, State, and ZIP Code) Air Force Institute of Technology Wright-Patterson AFB, Ohio 45433	7b. ADDRESS (City, State, and ZIP Code)
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification) ON-SITE ADAPTIVE DESIGN OF A DECISION SUPPORT SYSTEM FOR FIGHTER SQUADRON SCHEDULING (U)

12. PERSONAL AUTHOR(S)
James H. Jeter, B.S., Capt, USAF

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
MS Thesis	FROM ____ TO ____	1989 May	115

16. SUPPLEMENTARY NOTATION

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
Decision Support System, Adaptive Design, Scheduling, Fighter Squadron Scheduling

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

Thesis Advisor: John R. Valusek, PhD, Lt Col, USAF
Assistant Professor of Operations Research

(See reverse for Abstract)

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL John R. Valusek, Lt Col, USAF	22b. TELEPHONE (Include Area Code) (513) 255-2549 22c. OFFICE SYMBOL AFIT/ENS