ACKNOWLEDGEMENTS

Several people deserve thanks for their help during the production of this project. Software and hardware support was provided by VT-27. Several members of the Operations Department of VT-27, in particular, Jim Litz and Cory Allsop, helped with the proofreading of the project. Dr. Herbert Haynes, the project committee chairman, provided valuable assistance in the organization of the project report.

My greatest thanks go to my family. My wife, Sandy, and my three children, Elizabeth, Lyle II, and Matthew, were extremely patient and understanding during the time spent preparing this project.

Lyle J. Burnham
TABLE OF CONTENTS

ABSTRACT ............................................. 2
INTRODUCTION ........................................... 3
APPROACH TO THE SOLUTION ......................... 5
  DIVISION OF TASKS ................................. 5
  DATA ENTRY ....................................... 7
  THE DATABASE-RESIDENT SYLLABUS ................. 9
SCHEDULING ........................................... 11
ENVIRONMENT ......................................... 12
SUMMARY AND CONCLUSIONS ......................... 13
BIBLIOGRAPHY ......................................... 14
APPENDICES ......................................... 15
ABSTRACT

The primary objective of this project is to automate the scheduling process of VT-27, a Navy Primary Flight Training Squadron. The Scheduling Program uses a database-resident syllabus to allow for easy syllabus changes and a longer life cycle of the system. The secondary objective of this project was to integrate the Scheduling Program into the VT-27 LAN System. As of this date, the LAN system has not been installed. The Scheduling Program contains an installation portion which enables the user to enter all necessary data.
INTRODUCTION

The Commander of Naval Air Training (CNATRA) is responsible for providing designated naval aviators to the U.S. Navy, Marine Corps and Coast Guard. In order to become designated naval aviators, students must complete three phases of flight training - Primary, Intermediate, and Advanced. All students complete Primary flight training in the T-34C Turbomentor, a single-propeller, fixed-wing training aircraft. At the completion of Primary flight training, students are then designated to continue in the Jet, Propeller, or Helicopter syllabus. Students who are designated to continue in the Propeller or Helicopter syllabus complete their intermediate flight training with their primary flight training squadron.

VT 27, located at Naval Air Station Corpus Christi, is one of four Primary flight training squadrons in the U.S. Navy. The squadron completes the Primary flight training of approximately 400 students each year, of whom about half also complete Intermediate flight training. To accomplish this mission, the squadron consists of approximately seventy-five instructor pilots and sixty T-34C aircraft.

The squadron is divided into three flights to assist in the management of flight training. Each flight consists of approximately twenty-five instructors, sixty-five students in Primary flight training, and fifteen students in Intermediate flight training. Each flight has a Flight Leader who supervises three Scheduling Officers. These Scheduling Officers create the daily flight schedule on rotating weeks.

The Scheduling Officer ensures that each student completes the fifty-three flight/simulator events required of Primary flight training, and nineteen flight/simulator events of Intermediate flight training. Each day he must determine when instructors and students are available to be scheduled, what events a student is ready for, and what type of events an instructor is qualified to teach. The Scheduling Officer for each flight spends about six hours each day creating the flight schedule. Most of this time is spent performing administrative duties that must be completed prior to the actual scheduling of the flight/simulator events.

Once the three Scheduling Officers have completed their individual schedules, the squadron Flight Officer, with the assistance of the Scheduling Officers, merges the schedules together. He must check to see that conflicts do not exist in the schedule, that assets such as aircraft and simulators are not overtasked, and most important, that the maximum possible number of flight/simulator events are scheduled. This process takes an additional two to three hours every day.

The daily scheduling of instructors and students is a labor-intensive process. The Scheduling Officer spends a large
part of his day determining much of the data necessary to produce
the schedule, such as what events students are ready for and what
events instructors are qualified to teach. The Scheduling
Officer must also be thoroughly familiar with all the rules and
regulations set forth by all levels of the command that govern
scheduling. He has to be extremely knowledgeable about the
sequence of events that must be strictly followed in the Primary
and Intermediate syllabuses. A listing of the various rules that
that govern creating the flight schedule and the daily tasks of
the Scheduling Officer is enclosed in the appendices. Also
enclosed in the appendices are copies of the Primary and
Intermediate Master Curriculum Guides.

Since the introduction of computers into primary training
squadrons, many attempts have been made at automating the
scheduling process. Most of the software developed has had very
short lifespans. The main limitation of these programs was that
the flight syllabus would be transformed into code, and when the
syllabus changed, these programs became obsolete.
APPROACH TO THE SOLUTION

DIVISION OF TASKS

The first major step taken in designing the Scheduling Program was to determine what tasks the Scheduling Officer would be responsible for and what tasks the computer would perform. The tasks were divided the following way:

SCHEDULER OFFICER'S TASKS:

- Entering and maintaining information about events such as event name, event type, etc.

- Developing a syllabus chain for students and entering the appropriate information into the database.

- Entering information about instructors such as name, flight assignment, what events they can perform, etc.

- Entering information about students such as name, flight assignment, on-wing instructor, etc.

- Entering information about instructors who must fly with a student (on-wing instructor) and instructors who cannot fly with a student (instructor who awarded that student a down).

- Entering the availability of flight and simulator slots.

- Entering any miscellaneous information required.

- Determining and entering of snivels (crew rest, instructor or student non-availability).

- Entering of events completed by students.

- Scheduling students who need solo flights

PROGRAM'S TASKS:

- Checking the syllabus chain for mistakes (prior events that do not exist) after a chain is entered.

- Deleting related records of instructors and students from all databases when instructors or students are deleted.

- Adding related records to all databases when instructors or students are added.
- Entering information about flights that must be flown with on-wing instructors when students are added to the database.

- Determining students who have the time available for an event.

- Ensuring that instructors do not perform more than the maximum number of events per day and not more than the maximum number of CAT II events per day.

- Ensuring that day- or night-only events are scheduled at the appropriate time.

- Attempting to schedule students at night who still need night time.

- Producing a flight schedule as output.

- Producing a listing of students needing solo flights.

- Producing a listing of flight/simulator slots available after completion of scheduling.
DATA ENTRY

Initially, data necessary for the creation of the daily flight schedule needs to be entered and updated. When the VT-27 LAN System is installed (proposed date is January 1990), most of this time-consuming task will be performed by persons other than the Scheduling Officer through terminals connected to the LAN. For example, the Flight Duty Officer will update completed SNA events, and other departments will be responsible for updating information about instructors and students.

Since the LAN has not been installed, it was determined that the Scheduling Officer would be responsible for the initial entry and updating of the data necessary for creation of the daily flight schedule. The Scheduling Program handles some necessary maintaining of the databases (such as removal of all records belonging to a deleted instructor). The program also determines, through the database resident syllabus, the next events for a student, and attempts to schedule that student for one of those events.

Since there is a large amount of data that must be maintained in order to produce the schedule, the program is divided into two major portions. The installation portion includes the modules required to allow the Scheduling Officer to initially input and maintain the databases. Most of this data requires maintenance on a weekly or monthly basis.

The Daily Flight Schedule portion of the program enables the Scheduling Officer to make daily changes to the data, such as completed student events and snivels. Once this data is changed, the program then determines which students have the time available for an event and what the next events are. The program then attempts to schedule one of those events.

Since the accuracy of the daily flight schedule depends on the accuracy of the databases used, one of the major objectives in designing the program was to ensure that data was accurate before being stored in the database. This was accomplished through the following methods:

1. Most input screens have an option of either adding, changing, or deleting data. In some cases where the amount of data was small, it was determined to include only addition and deletion options.

2. "Yes or No" prompts were extensively used. In most cases these prompts were to determine if data was entered accurately. Usually if a negative response was returned, the subprogram aborted and returned to its parent screen.

3. When possible, cross-checking of data was performed. For example, after entering a student number, the student's name
was displayed on the screen and the user was asked to verify that the correct student was entered. If the student number entered could not be found, an appropriate message is displayed.

4. dBASE III PLUS's "picture" and "range" template functions were used to check the accuracy of input data when applicable. This assisted greatly in providing automatic data type and range checking.

5. In some cases data had to be entered as one type, but checked for accuracy as another type. For example, dBASE III PLUS does not include a "time" type. In order to check that data entered as times did not exceed maximum values (e.g. 1078), the time was entered as two numeric values (hours and minutes), checked with the "range" template function, and then, through string manipulation, converted to a four-place string value to be stored in the database.

6. When character values required certain values, data was checked to determine if it matched those certain values when entered. If not, the data was not accepted.

7. When instructors or students were added or deleted from the database, additions or deletions of records were made to appropriate permanent databases.
THE DATABASE-RESIDENT SYLLABUS

The major reason most other scheduling programs have been short-lived was the fact that the syllabus was represented in the actual logic of the program. When the syllabus changed, the program was no longer applicable unless the program was rewritten.

In order to alleviate this shortcoming in the Scheduler Program, the approach of producing a database-resident syllabus was used. Even though the Scheduling Officer actually develops the data for the syllabus, the Scheduler Program determines the next events that a student is ready for.

Since students must follow logical progressions through the flight syllabus, the syllabus can be represented as a graph structure. The syllabuses are usually divided into stages. For example, in the Primary Flight syllabus, the stages are Familiarization (FAM), Basic Instruments (BI), Precision Aerobatics (PA), Night Familiarization (NF), Formation (FORM), and Radio Instruments (RI). All primary students start with flying FAM events. After the completion of FAM-8, SNAs can then complete either the rest of the FAM events, or BI events, or a mixture of both. However, prior to completing FAM-14 (the student’s first solo flight), BI-3s must be completed. After the completion of FAM-14, students can begin flying PA events, but can also fly BI events. After the completion of all PA and BI events, students then begin either flying RI or FORM events, but not both. Anytime during the flying of RI or FORM events, NF events can be flown.

The data necessary for each event in the syllabus is the number of the event, the number of any next events, and the number of any events that must have been previously completed. This "chaining" of events produces a graph structure for the syllabus. Examples of the Primary and Intermediate syllabus chains are enclosed in the appendices.

When determining the next events that a student is ready for, the Scheduler Program conducts a breadth-first search through the syllabus graph. It is assumed that a syllabus has one starting point. Each event (starting at event number 1) is checked to see if it is completed. If an event is complete, the number of branches from this event is determined, and children nodes are placed in a queue. Each event in the queue is popped and checked for completeness.

If the event is incomplete, all required previous events are checked to see if they are complete. If a previous event is incomplete, the program moves up the chain until it finds an event that is complete. The event just after the completed event (an incomplete event) is then listed as an event that needs to be scheduled. If all required previous events are completed, then
the original incompleted event is listed to be scheduled.

The program uses the total number of branches to determine when to stop the breadth-first search. If new branches have a completed event, then branches are added. When a branch terminates (no next events) or an incompleted event is reached, branches are removed. When the number of branches reaches zero, the search terminates.

One of the major problems with developing the breadth-first search was the fact that at some points in the graph a student could branch to one path or another, but not both (e.g. RI or FORM). This was handled by providing a boolean value for each node in the graph. If the value was false, the branch was considered as a single branch. However, if the value was true, all true sibling nodes were checked for completeness. If any node with a true value was completed, that was the branch taken, and the other nodes were ignored. If none of the nodes with true values were complete, the last node checked was picked to be scheduled.

Another problem with representing the syllabus as a graph was the possibility of cycles. For example, a student can either complete all the FORM events and then start RI events or can complete all RI events and then start FORM events. Since there is a link between the last FORM event and the first RI event, and the last RI event and the first FORM event, a cycle exists. In order to stop the cycle, all events that are checked and are complete are placed in a list. If later in the chain, the event is checked and complete and it is in the list, the number of branches is decremented by one, thereby, removing the branch with the cycle.

Using the database-resident syllabus along with the search program enables the Scheduling Officer to change the syllabus, and still continue to use the Scheduler Program. However, the one big drawback is the time it takes to conduct the search. Using students who had completed about half of the events in the syllabus, the search averaged about three minutes for each student using a floppy disk drive machine, and just less than a minute for hard drive machines. In order to decrease the amount of time in creating the flight schedule, students are first checked to see if they are available to be scheduled, and only if a student is available is a search conducted to determine the next events the student is ready for.
SCHEDULING

Once all appropriate databases are prepared, the actual scheduling of students commences. The objective is to attempt to schedule all available students for one of their next incompleted events.

As was done manually, student and instructor time matrix databases are created and initialized. An example of the time matrix used by the Scheduling Officer is enclosed in the appendices. Snivels for both are then entered on the matrix to mark times of non-availability.

In order to save time from doing unnecessary searches through the syllabus chain, a student is checked to see if there is a four-hour period in which to schedule an event. If not, that student is skipped.

If a student has an appropriate time slot available, then a search is conducted through the syllabus to determine the next incompleted events that student has. If there are any (a student could be complete), then the first event is checked to see if it can be scheduled.

If the event is a simulator, the student's availability is checked to see if there is a matching simulator time slot. If so, the simulator is scheduled.

If the event is a flight, the event is checked to see if it is a day- or night-only event. If so, the earliest day or night time that the student is available is used. If the event is optional (day or night), and the student does not have enough night time, the earliest night time available is used. At this time, the database is checked to see if there is a specific instructor that the student must fly this event with. If so, that instructor is checked to see if he and the student are available at the same time. If so, the event is scheduled.

If there is no specific instructor that the student must fly that event with, then each instructor is checked to see if the instructor is qualified to fly that event and if the instructor has a compatible time with the student. If so, the database is again checked to ensure that there is no restriction not allowing that instructor to fly with the student. If not, the event is scheduled.

If the event cannot be scheduled, then any other events that the student is ready for are checked. If none of the other events can be scheduled, or once the student is scheduled, the program moves on to the next student.
ENVIROMENT

The overall plan of the project was to develop a scheduling program to interact with the VT-27 LAN System in order to develop a daily flight schedule. Most data input and updating would be accomplished by persons other than the scheduler (Duty Officer, other departments, etc.). Presently, the installation portion of the Scheduler Program is used to update the databases on a single Zenith 248 personal computer.

dBASE III PLUS was used as the language for the Scheduler Program because it was the only language processor owned by VT-27. There are many advantages of using dBASE III PLUS over other languages. Storage and retrieval of data are handled entirely by the language. It has several features that assist in setting up screens and the entering of data.

There were several disadvantages encountered that were specific to the dBASE III PLUS language. Even though data retrieval is handled by the language, at times it can be extremely slow. There is a maximum limit on the number of fields (128) that can exist in a database. Databases with more fields have to be separated into several databases. Another major limitation is that the fields in a database must be defined prior to executing the program. At several points during the development of the program, it would have saved storage to create fields as needed during execution. Since this feature was not available, fields must be created in case they may be needed. This causes wasted storage if the fields are rarely or never used. During the development of the program, it was convenient to use the event numbers as field names in several of the databases. However, all dBASE III PLUS field names must begin with a letter. Because of this it is necessary for several string manipulation operations to be performed when using numbers to represent fields names.
SUMMARY AND CONCLUSIONS

The database-resident syllabus in the Scheduler Program enables the Scheduling Officer to still use the program despite changes in the syllabus. Because of this feature, the program should have a longer lifespan than most other flight scheduling programs.

The amount of time spent entering data into the databases is a major liability of the program. This is true for the initial setup and daily execution. Once the VT-27 LAN System is installed, much of the database maintenance will be accomplished either automatically or by persons other than the Scheduling Officer, thereby reducing the man-hours spent creating the daily flight schedule.

Even though dBASE III PLUS seems to be extremely slow at times when processing data (such as determining the next events that a student is ready for), the time factor is acceptable. Presently about twenty-five to thirty man-hours each day are spent creating the schedule. It is estimated that it will take about eight man-hours to create the flight schedule using the Scheduling Program, and some of this time involves processing by the computer only, while the Scheduling Officer performs other duties.

Presently the scheduling portion of the program schedules each student for only one event. It also starts at the "top" of the list of students and instructors and moves down to schedule events. The program also does not handle some of the more complicated scheduling items such as scheduling hotseat and quickturn flights. Future enhancements will be made in order to lengthen the lifespan of the Scheduler Program.
BIBLIOGRAPHY

