NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

DESIGN AND IMPLEMENTATION OF AN OPERATIONS MODULE FOR THE ARGOS PAPERLESS SHIP SYSTEM

by

William R. Ault

June 1989

Thesis Advisor C. T. Wu

Approved for public release; distribution is unlimited.
**Title**: DESIGN AND IMPLEMENTATION OF AN OPERATIONS MODULE FOR THE ARGOS PAPERLESS SHIP SYSTEM

**Author**: William R. Ault

**Type of Report**: Master's Thesis

**Date**: June 1989

**Pages**: 132

**Abstract**: The "paperless" ship is an idea which has been advocated at the highest levels in the Navy. The goal is to eliminate the enormous amount of paper required in the normal operation of a modern naval warship. The ARGOS system under development at the Naval Postgraduate school is a prototype solution which uses HyperCard HyperTalk for prototype development. The operations functional area, including sections for training, scheduling, message generation, and publication management is an important part of this development.

**Subject Terms**: Software engineering, Database design.

**Abstract Security Classification**: Unclassified

**Telephone**: (415) 646-3391

**Office Symbol**: 52WQ
Design and Implementation of an Operations Module for the ARGOS Paperless Ship System

by

William R. Ault
Lieutenant, United States Navy
B.S., United States Naval Academy, 1982

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
June 1989

Author: William R. Ault

Approved by:

C. T. Wu, Thesis Advisor

D. K. Hsiao, Second Reader

Robert B. McGhee, Chairman, Department of Computer Science

Kneale T. Marshall, Dean of Information and Policy Sciences
ABSTRACT

The "paperless" ship is an idea which has been advocated at the highest levels in the Navy. The goal is to eliminate the enormous amount of paper required in the normal operation of a modern naval warship. The ARGOS system under development at the Naval Postgraduate school is a prototype solution which uses HyperCard/HyperTalk for prototype development. The operations functional area, including sections for training, scheduling, message generation, and publication management is an important part of this development.
# TABLE OF CONTENTS

I. INTRODUCTION ............................................................................. 1

II. PROBLEM STATEMENT ............................................................... 4

III. THE PROGRAMMING ENVIRONMENT ........................................ 7

IV. IMPLEMENTATION .................................................................... 10
   A. REPORTS .................................................................................. 10
   B. TRAINING ............................................................................... 14
      1. Ship Training ...................................................................... 14
      2. Personnel Training ............................................................. 16
   C. PUBLICATIONS ................................................................. 18
   D. SCHEDULES ......................................................................... 19

V. CONCLUSIONS ............................................................................ 21

LIST OF REFERENCES ...................................................................... 23

APPENDIX A. OPERATIONS STACK SCRIPTS ................................. 24

APPENDIX B. REPORTS STACK SCRIPTS ..................................... 25

APPENDIX C. SET LIBRARY STACK SCRIPTS ............................... 43

APPENDIX D. TRAINING STACK SCRIPTS .................................. 90

APPENDIX E. SCHEDULES STACK SCRIPTS ................................. 140

APPENDIX F. PUBLICATIONS STACK SCRIPTS ............................. 173

BIBLIOGRAPHY ............................................................................. 185

INITIAL DISTRIBUTION LIST ...................................................... 186
I. INTRODUCTION

The Navy of today capitalizes on virtually all aspects of modern technology. Nuclear power, cruise missiles, and satellite communications are but a few of the many examples of quantum leaps in technology which have been taken full advantage of by the U.S. Navy. The ability of shipbuilders and designers to quickly understand the benefits of a new technology, and implement that technology in U.S. Warships has been a hallmark of the U.S. Navy for over a century.

In the latter half of the twentieth century, the dominant new technology has been the computer. Computer technology has been implemented in fire control, guidance, and navigation systems, greatly enhancing capabilities in those areas. As far as non-tactical uses of computers are concerned, the primary implementation has been the shipboard non-tactical ADP Program (SNAP). The goals of the SNAP program are: “To collect information only once; to provide maximum automated information systems (either on line or off line); to require minimal supply, maintenance and training support; and to require no additional shipboard personnel.”

SNAP does not fully capitalize on existing technology in handling non-tactical data and information onboard a ship. Today, it is estimated that a small combatant (DD, FF) carries onboard upwards of twenty tons of paper [Ref. 1: p. 157]. Much of this weight is in the form of technical, training and maintenance manuals, personnel administration, and training records, and various other instructions and publications. Keeping this myriad of publications updated and accessible, and simply storing such a volume of

---

1 As stated in the SNAP II Organizational Maintenance Management Subsystem desk top guide.
paper quickly becomes a problem on a warship that has been optimized for space. On any modern combatant, space used to house paper is at a compromise of the mobility, habitability and warfighting capability of the ship. It is for these reasons that the "paperless ship" concept was introduced.

The goal of the paperless ship is to remove as much paper, if not all paper from a ships in order to reclaim the space and weight taken up by the paper for more mission critical uses. The ARGOS project is an effort to satisfy the requirements of the paperless ship, while maintaining the stated goals of the SNAP program.

The ARGOS project allows for computerized access, manipulation and creation of data normally stored on paper. ARGOS additionally allows commanding officers and battle group commanders to instantly access the material, personnel and training readiness of their ship or battle group, by accessing the information available through ARGOS.

ARGOS is a multi-media, object oriented, event driven data base system which combines textual and graphical data in allowing any use of that data that would be available if the data were stored on paper. The ARGOS prototype has been implemented at the Naval Postgraduate School using a Macintosh 2 computer, the HyperCard programming environment, and the HyperTalk programming language. At present, ARGOS is divided into six different modules, or functional areas: maintenance, operations, supply, administration, medical and personnel. This division is not static, but is merely a method of dividing the ship's administrative workload for the purpose of prototype development.

The purpose of this thesis is to demonstrate a design and implementation of an operations module for the ARGOS system, and to integrate it with the other modules in the ARGOS system.

2 Macintosh, HyperCard, and HyperTalk are registered trademarks of Apple Computer Inc.
The thesis is organized as follows:

- The statement of the problem. Identifying data and operations unique to the module.
- Programming environment. A brief discussion of HyperCard and HyperTalk.
- Implementation. A description of how the module was created.
- Conclusions. Lessons learned from the research, as well as recommended areas for further study/review.

The script listings for prototype stacks developed in conjunction with this thesis are included as Appendicies A through F.
II. PROBLEM STATEMENT

The operations officer on a FFG-7 class ship in the U.S. Navy has been provided with state-of-the-art detection, surveillance, communications, and weapons systems for his use as a watchstander. When he is through with his watch, however, he conducts his business for the most part without the help from modern technology. The responsibilities of the operations officer usually include the following:

- Preparing the ship's employment schedule.
- Maintaining tactical publications and instructions, operations order, and Naval warfare publications.
- Serving as the ship's Training Officer.
- Communications.

In preparing the ship's schedule (sometimes one year or more in advance), a method is needed to make the initial schedule, revise it, and present alternative schedules for final approval. When done manually, this process involves many hours of additions, deletions, and rewrites, and becomes very labor intensive. The schedule approved is only a proposal, and must be forwarded to the fleet and type commanders for approval. The official ship's schedule which results is invariably different, which requires generation of the new schedule sheets.

The maintenance of the tactical publications and instructions library appears to be straightforward. This includes entering promulgated changes, and keeping custody records for each publication. When a publication is needed, its location and change status (that is, what changes have been entered) must be determined. This is usually accomplished by locating the publication librarian or searching the ship's office in the hope of finding what
is needed. Depending on the availability of personnel, this can be time consuming, and frustrating exercise.

The duties of the ship's training officer include:

- Scheduling standard training requirement (STR) accomplishment.
- Overall responsibility for divisional and departmental training, and General Military Training (GMT).
- Surface Warfare Officer (SWO) and Enlisted Surface Warfare Specialist (ESWS) training.
- Management of off-ship schools.

STR scheduling and record involves ship's exercises as specified in Fleet Exercise Publications (FXP). The ship generates and submits to the type commander a training report periodically. This computer formatted message is entered into the type commander's database and the ship receives a report of its training status for the type commander. The problems that arise are ensuring that all exercises conducted since the last training report are included in the message, and properly updating the database (manually).

The oversight of divisional and departmental training consists mainly of ensuring compliance with Navy training requirements and regulations. These requirements include maintaining training records (schedules, attendance records, accomplishment records), lesson plans, and qualified instructors lists. On even a small ships, this involves a massive amount of paper. Additionally, the formats for attendance records and lesson plans vary widely from division to division. SWO and ESWS training require the same records that divisional training does, but these records are maintained by the Training Officer personally.

Managing the ships utilization of off ships schools involves knowing the ships requirements for graduates on board, and scheduling school quotas to maintain the proper number of graduates onboard. The required graduates information is promulgated by the type commander as part of a TYCOM instruction. The scheduling information for convening
dates and available quotas is located in the Catalog of Navy Training Courses (CANTRAC). Searching for a particular course in the CANTRAC, which is distributed on microfiche, can be very tedious and time consuming.

The final area of responsibility is communications. Messages transmitted by a ship are either free format text messages (similar to a telegram), or formatted messages to be entered into the World Wide Military Command and Control System (WWMCCS) database. Formatted messages follow a strict format which enables the data contained to be scanned into the WMMCS database by computer. It follows that the best way to draft such messages is via computer. A software system is presently in the fleet to do this, but it is a stand alone system used only in generating formatted messages.

With the exception of formatted message generation and maintenance of individual training via service record entries, the functions described above are done manually, often at the expense of efficiency. The operations module for ARGOS should address these areas and make the performance of these tasks much more effective and efficient.

The major problem in developing such a system at the Naval Postgraduate School is ensuring that the information used in development is current and correct. We are "out of the loop" in the promulgation of new directives, and changes in existing instructions. While we may draw on personal fleet experience as a guide, ensuring 100% correctness of the information used to develop the system is impossible at NPS.

Another consideration is the security classification of the information. A complete system could not be implemented without addressing this issue, especially in the area of message generation. Since the goal is a working prototype that will demonstrate the capabilities of the system, these problems have minimal impact on system development. They must be addressed however, before a complete system can be implemented.
III. THE PROGRAMMING ENVIRONMENT

This chapter discusses the HyperCard environment and its programming language, HyperTalk. HyperCard was developed by Apple Computer for use in the Macintosh family of computers. HyperCard version 1.2.1 was used in developing this thesis.

HyperCard is an event-driven, object oriented programming environment. All HyperCard actions are initiated by messages sent to objects [Ref. 2: p.12]. The basic structure of HyperCard is the stack. The term "stack" should not be confused with the standard last-in first-out data structure normally associated with the word stack in most computer applications. In HyperCard, a stack is analogous to a 3x5 card file. Each stack is a HyperCard object containing one or more HyperCard cards. Each card consists of pictures (graphics), fields, and buttons. Fields, and buttons are also HyperCard objects, while pictures are not. Fields are areas where text is read or entered by the user, and where text is stored for access and manipulation by HyperCard. A field may be locked to prevent modification by the user, or unlocked to allow text to be added or deleted. Buttons are primarily designed to perform some action on mouse events (e.g., mouseUp, mouseDown). The final HyperCard object is the background. A background has the same structure as a card, it may contain graphics, fields, and buttons and is associated with a particular stack. A background is shared by cards in a stack, each card is associated with a background. If a stack contains only one background, it is said to be homogeneous. If there is more than one background in a stack, the stack is heterogeneous.

Each card, in effect, consists of several layers. At the bottom is the background graphics layer. Any graphics common to several cards are placed here. Background buttons and fields come next, each occupying its own layer. The background is visible in all
cards associated with that particular background. At the card level, the same structure as in the background is followed. The graphics is the lowest layer (furthest from view), with buttons and fields layered above.

Events in HyperCard cause it to send messages, which in turn may cause some action to occur based on the contents of the script of the object receiving the message. Messages travel through HyperCard along a message hierarchy, which is a one way path from buttons and fields, to the card, the background, the stack, the home stack, and finally to HyperCard itself. The location in the hierarchy at which HyperCard sends a message is called the entry point [Ref. 3: p. 376]. When a message is sent HyperCard searches the script of the object at the entry point for a message handler for the current message. If a message handler is found, it is executed and the message is "trapped" (i.e., it stops its journey up the message hierarchy). If no message handler is found, the message continues up the hierarchy to the next level. If the message gets to the HyperCard level without encountering a message handler, it is lost and no action results. When a handler executes, it sends its statements as messages, first to its own object, and then up the message hierarchy. A message handler acts on the object in which it is contained, regardless of where the message originated.

Statements in a HyperTalk script may be HyperTalk commands, HyperTalk functions, user-defined functions, external commands (XCMDs), or external functions (XFNCs). XCMDs and XFNCs are written in Pascal, C, or 68000 assembly language and compiled separately. They are then added as resources to the stack and may be called by HyperCard. XCMDs and XFNCs allow the programmer to add features to a HyperCard stack that are not supported by HyperCard itself. HyperTalk offers a fairly complete set of commands and functions, and the ability of the programmer to define his own functions, extends the applications of HyperCard.
There are several advantages in using the HyperCard environment and HyperTalk for the development of the ARGOS operations module. The interface is very intuitive, which allows the user to quickly learn how to use it. Stacks can be created using HyperCard alone without any knowledge or use of HyperTalk code. Users can rapidly discover the power of HyperCard before even familiarizing themselves with the HyperTalk language.

HyperTalk is very close to the natural language. Commands closely resemble imperative English (e.g., “put card field 1 into field answer”), and the syntax is relatively forgiving. This allows the new programmer to quickly become comfortable in writing HyperTalk scripts, which, in turn allows for faster prototype development.

HyperTalk is an imperative language, which eliminates the need for compilation and “make” commands associated with most high level languages. This greatly speeds up stack development and debugging. HyperTalk’s imperative nature can have a negative impact on execution speed, but execution is still fast enough in most instances. Since HyperTalk is object-oriented, scripts tend to be small, easy to understand and fully transportable to other objects scripts. The benefits of using HyperCard/HyperTalk as a prototyping tool can be summarized in the fact that the time (cost) of development is drastically reduced.
IV. IMPLEMENTATION

A. REPORTS

The reports subarea is designed to aid in the composition of formatted messages for transmission from the ship to higher authority. These messages follow a strict format to allow the information contained in the message to be added to the WWMCCS data base. The prototype message generator modeled the Oprep-3 message reporting system. Oprep-3 messages are transmitted to report incidents of high national or Navy interest. Guidelines for submission of Oprep-3 reports are contained in OPNAVINST 3100.6D, which was used in the development of this system. Due to their nature, Oprep-3 reports are very time sensitive, with submission of the initial report required within 20 minutes of the incident. Additionally, since the reports are formatted, the format must be strictly followed. With these requirements in mind, the system was designed to enable the user to quickly draft an Oprep-3 message, while at the same time ensuring that format and content requirements of OPNAVINST 3100.6D are adhered to.

When the user selects “Reports” from the operations menu, he then is sent to the reports stack. The first card in the stack contains a button labeled “Reports”. This button uses the HPopUpMenu XCMD to implement a hierarchical pop up menu of report types. The menu appears when the mouse is down and dragged down towards the bottom of the screen from within the button. When the mouse crosses the bottom edge of the button, the first menu appears. This menu lists the different report types that may be drafted. The menu was taken directly from the formatted message origination system (FMOS), which is a stand-alone system currently available in the fleet. The only menu choice implemented

3 The HPopUpMenu was written by Guy de Picciotto and is available as "freeware" through International Datawares Inc., San Jose, CA.
is "Oprep-3". When the mouse is dragged into this choice, another menu pops up listing the different types of Oprep-3 messages. "Navy Blue" is the only choice implemented, and is selected by releasing the mouse over the selection.

When "Navy Blue" is selected, the user is sent to the Navy Blue Card. This card lists all the required and conditional data elements used in drafting an Oprep-3 Navy Blue message. An open card message handler is used to ask the user through the use of dialog boxes whether the report is an initial report, an amplification of an earlier report, or the final report on the incident being reported. The response is recorded in a global variable for later use. The user is also asked to choose the Classification for the message (secret, confidential, or unclassified) and this response is also stored in a global variable. The classification can be subsequently changed by choosing the "Classification" button on the Navy Blue card.

Once the open card handler has executed, the user can then either draft the message by choosing the appropriate data set, or enter the addressees for the message. The data sets are listed on the Navy Blue card, and are selected by clicking the mouse on the data set name. When the data set is chosen, the user is then sent to the "Set Library" stack, and the card containing information on the data set chosen. Each data set card has a field defined for each message data set. Individual data elements can be either required or optional. Data element fields can be of a fixed length, variable (with a maximum length), or free text (unlimited length). The system must ensure that entered data elements are within the format size limits, and that they do not contain illegal characters. This was accomplished by the use of the HyperCard idle message.

The idle message is sent by HyperCard to the current card when no other system operation is taking place. Each data set card keeps track of the cursor location by putting the field name of the active field into a hidden card field. The card sends the idle message to that field, and each field has an on idle message handler. The on idle handler in the field
checks for illegal characters - namely the slash (/), which is used in the Oprep-3 reporting system as a data field delimiter. This checking is done quickly through the use of the offset function. The offset function returns the location of an indicated chunk of text in a specified container. If the chunk is not in the container, zero is returned. If the offset of the single slash in the present field is not zero, then the slash is removed, a beep sounds, and the cursor is placed after the location of the deleted slash.

The length of the active field is also checked in the field’s idle handler, if necessary, through the use of the “length of” function. If the user enters an extra character in the field, the extra character is deleted, a beep sounds, and the cursor is placed at the end of the field. The user can progress through the fields by using the mouse, the tab key, or the return key. Each field has a mouse within handler to select after the last character of that field when the mouse is within the field.

Where possible, required data known by the message generating system is automatically entered into the appropriate data field. The originator of the message (USS Jarrett in the prototype), the message type, the report type (initial, amplification, or final) and the report serial number are all entered in the appropriate fields by the system. The serial number of the message is automatically incremented when the user enters the message card (Navy Blue in the prototype) if the message is cancelled, the proper serial number is restored. This automatic data entry feature decreases the time required to draft the message.

Each data set card contains an “Enter”, “Return”, and “Delete” button. The enter button verifies that all required data fields are present. If required data is missing, an error message is displayed, and the user is returned to the appropriate data field. If all required data is present, the data fields, the single slash delimiters, and the double slash end of data set marker are entered into the message, adhering to the 69 character per line requirement of OPNAVINST 3100.6D, and the user is returned to the Navy Blue card. The delete button deletes the data set from the message currently being drafted, and returns the user
to the data set card. The user is then returned to the message card to choose the next operation. The cancel button simply empties all data fields on the data set card, and returns the user to the message card.

All information on data set content contained in OPNAVINST 3100.6D appears in the graphics of the appropriate card. The field delimiters and end of set marker are also on the card in the appropriate places. The data entry fields use the courier 12 font, which is proportional. If the field can contain a maximum of 20 characters, then any 20 characters of courier 12 will occupy the same amount of space. This was important in tailoring the physical size of the data fields, allowing only the proper number of characters to be displayed.

The addressees of the message are entered by clicking the "addressees" button on the message card. The user is then sent to the addressee card. This card automatically contains the action and information addressees required by OPNAVINST 3100.6D for the message. Additional addressees may be required, depending on the incident being reported. These addresses may be entered by clicking on the appropriate address. When all addresses have been entered, they may be entered by clicking the enter button, or canceled by clicking the cancel button. In either case, the user is returned to the message card.

When the message drafting is complete, the user may review the message to ensure that the information is correct, and to check spelling etc. Manual changes can be made to the message at this stage. The message can then be printed, or canceled. If it is printed, it is also saved for future reference on its own card. Printing of the field is done by using the PrintField XCMD. The saved messages can be viewed and deleted through the message file. The message file is accessed via the "Msg File" button located on the top card of the reports stack.

4 Portions of the PrintField XCMD are copyrighted by Think Technologies. It was written by Mark Scherfling.
B. TRAINING

1. Ship Training

Ship training is conducted according to standard training requirements (STRs), which describe training evolutions conducted by the ship in various mission areas (ASW, AAW, etc.). Mission area STRs are divided into five categories, core, basic, intermediate, advanced, and repetitive. Each mission area has an “M-rating” which describes the status of training readiness in that mission area. M-ratings are from M-1 to M-4, and are determined by the status of the mission area STRs. When an STR is conducted, a rating of M-1 is assigned. For repetitive STRs, the M-rating degrades to M-2, M-3, and finally M-4 at intervals defined in fleet exercise publications. For all other STR categories the M-1 rating is current for 21 months, at which time it degrades directly to M-4. Due to classification considerations, the Mobility (NIOB) mission area was the only mission area modeled by the prototype. The MOB mission area is further divided into four sub-areas: Engineering (MOB-E), Damage Control (MOB-D), Seamanship (MOB-S), and Navigation (MOB-N).

Each STR is uniquely identified by a unique six digit exercise code. The first two digits identify the primary mission area, the third digit identifies the training category, and the fifth and sixth digits identify the STR within the particular category. Data for each STR are stored using the “item” facility of HyperCard. An item is a chunk description for a string in a container delimited by a comma. For example, if “Tom,Dick,Harry” were in a container, “Tom” would be item 1, “Dick” item 2, and “Harry” item 3. The item facility makes access to data elements a quick, easy and accurate process, especially when certain data elements may be of varying length or omitted entirely. For STR data storage, the data items are, in order, exercise code, STR number, Exercise Name, M-rating, completion date, expiration date, score, evaluation method, and reporting source. Additionally, repetitive STRs have the added items of M-2 degradation period, M-3 degradation, and
M-4 degradation. These item lists are stored in background fields, with a dedicated card for each primary mission sub-area (MOB-E, MOB-D, MOB-S, MOB-N). Background fields must be used, as the HyperCard find command will not search card fields. For each of these fields containing the STR item lists is a corresponding background field which contains the same information formatted for viewing by the user. This apparent redundancy is necessary since data stored as items are in an inappropriate format for viewing by the user.

To view the STR data base, the user selects the STR option in the training pop up menu on the first card of the stack. When this item is selected, the second pop up menu offers the option of viewing the data base or drafting a training report (TRAREP). When the “View” option is chosen the user is sent to the view card, which has an open card handler which asks the user to input the mission area to view. Any of the subareas, or the entire MOB data base may be viewed. The user can update the database by clicking the “Update” button. The updating process involves checking the expiration date of each STR in the database currently in view. If the grade has expired, the M-rating is degraded, and the expiration date is erased. The date of the most recent update is displayed on the screen, and is stored in a field on the appropriate data storage card. the M-rating for the mission area in view can be determined by clicking the “M-Rating” button. The new M-rating is calculated, displayed, and stored in the same manner as the update information. The M-rating operation does not update the expiration dates, and the M-rating computed is only as good as the data in storage.

When an STR is accomplished, the fact is recorded by clicking the “Enter Data” button. This button asks the user to enter the exercise code, completion date, score, and evaluation method. The data is entered in the appropriate data item line, a new expiration date is computed, a M-rating of M-1 is assigned, and the string “PDG” is placed in reporting source field. This marks the data as unreported, ensuring that it will be included...
in the next TRAREP. The corresponding data viewing field is also updated, and the result is sent to the viewing field on the card the user is currently viewing.

When the user desires to draft a TRAREP, this option is selected from the pop up menu on the training card. The user is then sent to the “draft TRAREP” card, which has three buttons; “Draft”, “Cancel”, and “Print”. The card has an open card handler which automatically increments the message serial number and displays it on the screen, and records and displays the date-time-group of the message. When the “Draft” button is clicked, the entire STR database is searched for the “PDG” flag in the reporting source item. When a PDG flag is found, the appropriate data is entered in the TRAREP, and the new serial number is placed in the reporting source item. If no STR accomplishments are found, the user is notified, and the message drafting process is aborted. The user is then asked if there is any air controller data to report through a series of HyperCard “Ask” and “Answer” functions. When the message is completed, it is displayed for review, and may be either saved/printed by clicking “Print”, or canceled by clicking “Cancel”. If cancel is selected, the database is returned to its previous state (the “PDG” string replaces the new serial number), and the user is returned to the training card. The Print operation is performed by using the PrintField XCMD, since HyperCard does not directly support the printing of fields only.

2. Personnel Training

Personnel training is conducted in accordance with the Personnel Qualification System (PQS), leading to qualification in various PQS watch stations. The prototype provides a mechanism for scheduling PQS training on a quarterly basis. The options available to the user are:

- Schedule creation
- Schedule modification
- Recording training accomplishment
- Schedule deletion
- Drawing and printing a training schedule chart.

Each option is accessed using the pop up menu described previously, and each has its own card.

The schedule creation card has the open card handler which asks for the title of the schedule and the calendar quarter of the schedule. When the quarter is entered, a calendar for the three months of the quarter is drawn on the card for the user to refer to. The user then enters the lesson name and scheduled date of training for up to 15 lessons.

Data is stored using the item feature, with item 1 being the lesson name, item 2 the date, and item 3 either an “S” or a “C” to indicate scheduled or completed training. When the schedule is saved, a new card is created, and the title, quarter, and schedule data are stored in card fields. The name of the card is the title of the schedule and the quarter scheduled. This allows for the use of the same schedule title in many different quarters. A listing of schedule card names is maintained on the “Schedule File” card. When a lesson is completed, it is annotated in the “Record Accomplishment” card. The user enters the date the lesson was conducted, and the schedule date is replaced by this date, and a “C” is placed in item 3 for the specific lesson to signify completion. The schedule is then returned to storage with the changes added.

The user can draw a schedule any time after it has been created by selecting the “Draw Schedule” option from the pop up menu. The 15 lesson limit on schedules is due to the space limitation of the draw function. The drawing process uses the line and text painting tools available in HyperCard. Two problems were encountered in implementing this function. First, when changing fonts while printing text, all text entered since the last mouse click is changed to the new font. Therefore, before changing fonts, the mouse must be clicked to ensure all text remains in the desired font. Second, when entering text near
other graphics with white space surrounding the text character. To avoid this, text is entered first, and graphics afterward. When the schedule chart is completed, it may be printed using the HyperCard print card function.

A major problem with the HyperCard date functions was also encountered in implementing this area. All dates entered must be in the HyperCard “Short date” format (e.g., 6/7/89 for 7 June 1989). The problem is the fact that HyperCard will accept an invalid date (e.g., 2/31/89, 13/13/89) and transform it into some date which is valid format, but incorrect in content. This problem was overcome by creating a user defined function **validDate** which ensures that only valid dates are entered into the training data base in both the ship training and personnel training areas.

C. PUBLICATIONS

The publications subarea was created in the prototype to serve mainly as a stub for future development. The facilities provided are:

- Finding a publication by title.
- Listing publications by originator.
- Listing publications by classification.
- Listing publications by location/custodian.
- Entering/Deleting titles.
- Updating custodian or change number information.

Data is stored in a background field using the item facility, with a corresponding field of formatted data for the user to view. The data items are: Title, annex number, appendix number, tab number, effective date of the publication, classification, latest change number and custodian. The change number and custodian information can be changed by the user, new titles can be entered, and titles can be deleted. Data storage is similar to the STR data base. All data items are stored in a background field, with corresponding formatted data for viewing stored in another background field on the same card. When changes are made
to the database, both fields are updated accordingly. The prototype is only an inventory system, since mass storage for the large volume needed to store an entire publication library, and security issues are still under research.

D. SCHEDULES

The schedule subarea is designed to facilitate the creation of the ship's employment schedule, and also to record the ship's actual employment for historical purposes. Information on employment scheduling was taken from Naval Warfare Publication 10-1-10 Chapter 8. The methods used were very similar to the personnel training scheduling functions described in section B. Schedules are created, modified, deleted, and drawn in much the same manner. There are some significant differences, however.

A schedule event must have a start date and an end date, they are the same for a one day event. Also, events are defined as either occurring in port or underway, and any event may be either a major (primary) or concurrent employment. The ship must have one major employment schedule for every day of the quarter. Again, the item facility was used, with data items as follows: employment abbreviation, start date, end date, major or concurrent employment, and inport or underway.

Employment abbreviations are stored in a background field, and a corresponding field of formatted data for viewing. These fields do not change, with all data being a reproduction of table 8-1 of NWP 10-1-10. When a schedule abbreviation is entered during creation or modification, a check is made using the find function to ensure its validity. If it is not a valid entry, the user is notified. On a valid entry, the inport/underway status of the event is retrieved and the user is asked to select either major or concurrent employment. A check is made to ensure that major employments do not overlap and valid data is entered into the schedule. When a major employment is scheduled, an annotation is made in a card field signifying that a major employment is scheduled for the appropriate days.
When the schedule is saved, this field is checked to ensure major employment for every day of the quarter. This field must also be saved with the schedule, and retrieved for use when modifying the schedule.

Information on the underway and inport dates is stored in the “official” schedule, since this information is required by other functional areas. The term “official” is used because the user may want to store several alternative schedules for the same time period during the schedule planning process. Only the information designated official will be available for use by other modules.
V. CONCLUSIONS

The design and implementation of the ARGOS operations module has demonstrated not only the feasibility of such a system, but also the strengths of HyperCard/HyperTalk as a system prototyping tool. The four subareas modeled are representative of basic areas of responsibility of the fleet operations officer. These areas are by no means all inclusive or complete.

Throughout the design process, an effort was made to keep the system as user-friendly and simple to operate as possible. The ARGOS operations module makes information available where and when it is needed. Thus, the user will be more productive, efficient, and make fewer errors. He can spend more time producing (i.e., writing a training schedule or operational report) and less time investigating. Formatted messages can be drafted quickly and virtually error free. Employment schedules and training schedules can be created and maintained quickly, and information can be extracted easily. The end result of the development of the ARGOS operations subarea is that the job efficiency of the people using the system will increase substantially.

Before the operations functional area can be fully implemented, there are several problem areas that must be addressed. First and foremost is security. the majority of publications, messages, and information the operations officer deals with on a daily basis carry at least a confidential classification. This fact limited the scope of development of the operations functional area, since it was desired to keep the ARGOS prototype unclassified. The solution to the security issue will undoubtedly involve a combination of the use of HyperCard’s password capabilities and additional physical security measures.

Another need for full implementation is the requirement to integrate some type of ROEM (removable optical erasable media) and/or CD-ROM (compact disk - read only
memory) mass storage device. This capability is especially needed as a storage device for publications and instructions. Having reference publications on some computer accessible medium would allow rapid searching for subject matter by keywords. This capability, like all others in the ARGOS system, would greatly increase efficiency.

There are several areas of development in the operations functional area that are worthy of consideration for future research and development:

- Addressing the problems of computer security associated with the ARGOS system.
- Complete implementation of the message generation subarea, including computer interfaces with the ship’s communication system.
- Development of the training subarea as its own functional area, or the development of a transportable training module for use as an add-on to the administrative functional area.
- Development of a mass storage (ROEM/CD-ROM) capability for ARGOS.
- Design and Implementation of a navigation subarea or functional area.

Obviously, the above list is not a complete representation of the possible areas of improvement and expansion of the ARGOS system. A major portion of the acceptance of a new system such as ARGOS involves salesmanship. Demonstration of other applications of Macintosh and HyperCard/HyperTalk capabilities would only serve to increase the attractiveness of ARGOS as a system. For example, developing a personnel training module, while not directly associated with ARGOS, would make ARGOS more attractive, since the personnel training would use the same hardware. Alternatively, making ARGOS transportable to the MS-DOS environment would make the system very attractive, since that capability already exists in the fleet.

The primary goal in the development of the ARGOS system is to substantially reduce or eliminate the need for paper aboard ship. It is clear that this goal is met in the operations functional area. Additionally, significant increase in efficiency will be realized when using a fully implemented ARGOS system, since data is more accessible to the user than it is when stored on paper.
LIST OF REFERENCES


APPENDIX A. OPERATIONS STACK SCRIPTS

SCRIPTS FOR STACK: operations

** BACKGROUND #1: Operations **************************************
on openStack
  hide message box
  show menuBar
  pass openStack
end openStack

** CARD #1, BUTTON #1: Up ********************************************
on mouseUp
  visual effect zoom out
  go to card id 10931 of stack argos
end mouseUp

** CARD #1, BUTTON #2: Reports ***************************************
on mouseUp
  go to reports
end mouseUp

** CARD #1, BUTTON #3: Training *************************************
on mouseUp
  go to training
end mouseUp

** CARD #1, BUTTON #4: Publications **********************************
on mouseUp
  go to pubs
end mouseUp

** CARD #1, BUTTON #5: Schedules ***********************************
on mouseUp
  go to schedules
end mouseUp

** CARD #1, BUTTON #6: EXIT ***************************************
on mouseUp
  go argos
end mouseUp
APPENDIX B. REPORTS STACK SCRIPTS

SCRIPTS FOR STACK: Reports

** BACKGROUND #1: Operations ****************************
on openStack
  hide message box
  show menuBar
  pass openStack
end openStack

** CARD #1: reports ****************************
on openCard
  global draftflag
  put "false" into draftflag
end openCard

** CARD #1, BUTTON #1: exit ****************************
on mouseUp
  go argos
end mouseUp

** CARD #1, BUTTON #2: reports ****************************
on mouseDown
  put "Service" into menu1
  put return & "Maritime" after menu1
  put return & "General Purpose" after menu1
  put return & "Air Defense\nControl" after menu1
  put return & "FlagOTC" after menu1
  put return & "NGFS" after menu1
  put return & "Joint Mgs" after menu1
  put return & "OpRep 3,Pinnacle NucFlash,Pinnacle Front Burner,Pinnacle Emergency Destruction-Disability,Pinnacle Emergency Evacuation,Pinnacle Broken Arrow,Pinnacle,Navy Blue Faded Giant,Navy Blue Bent Spear,Navy Blue Dull Sword,Navy Blue,Unit SitRep" after menu1
  get HPopupMenu(menu1,0,74,67)
  if it is not zero then
    Put Item 1 of it into TheLine
    put Item 2 of it into TheItem
    If TheLine = 1 and TheItem = 2 then
      push card
      go to card id 4268
    end if
    If TheLine=8 and TheItem=2 then
      push card
      go to card "pinnacle"
end if
If TheLine = 8 and Theltem = 11 then
push card
go to card navy_blue
end if
If TheLine=8 and Theltem=4 then
go to card id
end if
If TheLine=1 and Theltem=3 then
go to card id
end if
end if
end mouseDown

** CARD #1, BUTTON #3: Msg File
on mouseUp
push card
go to card msg_file
end mouseUp

** CARD #1, BUTTON #4: Msg Settings
on mouseUp
push card
go to card settings
end mouseUp

** CARD #1, BUTTON #5: return
on mouseUp
go to operations
end mouseUp

** CARD #2: navy_blue
on openCard
   global msgtype, drafter, msgflag, msgflag2, draftflag, oldnum, serno, status, class
   put empty into msgtype
   put "OPREP-3" into msgtype
   put empty into msgflag
   put "NAVYBLUE" into msgflag
   put empty into msgflag2
   put ".-" into msgflag2
if draftflag <> "true" then
   put "true" into draftflag
   set lockscreen to true
   put empty into drafter
   put card field orig of card settings into drafter
   put card field orig of card settings into card field orig of card addees
   put card field action of card settings & return into addees
   card field action of card addees
   put card field info of card settings & return into card field info
of card addressees
set lockscreen to false
answer "Initial, Amplification, or Final report" with "Initial" or "Amplification" or "Final"
put it into response
answer "Classification of message ?" with "Secret" or "Confidential" or "Unclas"
if it = "Secret" then
  put "SECRET" into class
else
  if it = "Confidential" then
    put "CONFIDENTIAL" into class
  else
    put "UNCLAS" into class
  end if
end if
if response = "initial" then
  put "INIT" into status
  if card field ser_no of card settings is empty then
    put empty into oldnum
    put "001" into serno
    put serno into card field ser_no of card settings
  else
    put card field ser_no of card settings into oldnum
    put char 1 to 3 of oldnum into temp
    put temp + 1 into newnum
    if newnum < 10 then
      put "00" & newnum into serno
      put serno into card field ser_no of card settings
    else
      if newnum < 100 then
        put "0" & newnum into serno
        put serno into card field ser_no of card settings
      else
        put newnum into serno
        put serno into card field ser_no of card settings
      end if
    end if
  end if
end if
exit openCard
end if
if response = "amplification" then
  put "FOLUP" into status
else
  put "FINAL" into status
end if
put card field ser_no of card settings into oldnum
put the length of oldnum into len
if len = 3 then
  put oldnum & "A" into serno
put serno into card field ser_no of card settings
exit openCard
end if
if len = 4 then
    if char 4 of oldnum = "Z" then
        put char 1 to 3 of oldnum & "AA" into serno
        put serno into card field ser_no of card settings
    else
        put char 1 to 3 of oldnum & ~
        numToChar(CharToNum(char 4 of oldnum) + 1) into serno
        put serno into card field ser_no of card settings
    end if
    exit openCard
end if
if len = 5 then
    if char 5 of oldnum = "Z" then
        if char 4 of oldnum = "Z" then
            put char 1 to 3 of oldnum & "AAA" into serno
            put serno into card field ser_no of card settings
        else
            put char 1 to 3 of oldnum
            & numToChar(CharToNum(char 4 of oldnum) + 1) & "A" into serno
            put serno into card field ser_no of card settings
        end if
    else
        put char 1 to 4 of oldnum & ~
        numToChar(CharToNum(char 5 of oldnum) + 1) into serno
        put serno into card field ser_no of card settings
    end if
end if
end openCard

** CARD #2, BUTTON #1: Return **************************************************
on mouseUp
answer "Message will be deleted" with "OK" or "Return"
if it is "ok" then
    global msgtype,drafter,serno,msgflag,msgflag2,status,draftflag,oldnum
    put empty into msgtype
    put empty into drafter
    put empty into serno
    put empty into msgflag
    put empty into msgflag2
    put empty into status
    put empty into draftflag
    put oldnum into card field ser_no of card settings
    put empty into oldnum
    set lockScreen to true
    go to card scratch of stack set library
    put empty into card field test
    go to card "navy blue" of stack reports
    pop card
end if
end mouseUp
** CARD #2, BUTTON #2: exer *****************************************************
on mouseUp
  push card
  set lockScreen to true
  go to card scratch of stack set library
  if "OPER/" is in card field test then
    answer "OPER' Field used" with "return"
    pop card
    set lockScreen to false
    exit mouseUp
  end if
  go to card exer of stack set library
end mouseUp

** CARD #2, BUTTON #3: oper *****************************************************
on mouseUp
  push card
  set lockScreen to true
  go to card scratch of stack set library
  if "EXER/" is in card field test then
    answer "EXER' Field used" with "return"
    pop card
    set lockScreen to false
    exit mouseUp
  end if
  go to card oper of stack set library
end mouseUp

** CARD #2, BUTTON #4 *****************************************************
on mouseUp
  push card
  go to card msgid of stack set library
end mouseUp

** CARD #2, BUTTON #5: ref *****************************************************
on mouseUp
  push card
  go to card ref of stack set library
end mouseUp

** CARD #2, BUTTON #6: ampn *****************************************************
on mouseUp
  push card
  go to card ampn of stack set library
end mouseUp

** CARD #2, BUTTON #7: narr *****************************************************
on mouseUp
  push card
  go to card narr of stack set library
end mouseUp
** CARD #2, BUTTON #8: flagword

```
on mouseUp
    push card
    go to card flagword of stack set library
end mouseUp
```

** CARD #2, BUTTON #9: timeloc

```
on mouseUp
    push card
    go to card timeloc of stack set library
end mouseUp
```

** CARD #2, BUTTON #10: gentext

```
on mouseUp
    push card
    go to card gentext of stack set library
end mouseUp
```

** CARD #2, BUTTON #11: rmks

```
on mouseUp
    push card
    go to card rmks of stack set library
end mouseUp
```

** CARD #2, BUTTON #12: closetext

```
on mouseUp
    push card
    go to card closetext of stack set library
end mouseUp
```

** CARD #2, BUTTON #13: decl

```
on mouseUp
    push card
    go to card decl of stack set library
end mouseUp
```

** CARD #2, BUTTON #14: Classification

```
on mouseUp
    global class
    answer "What is the classification ?" with "Secret" or "Confidential" or "Unclas"
    if it = "secret" then
        put "S E C R E T" into class
    end if
    if it = "confidential" then
        put "C O N F I D E N T I A L" into class
    end if
    if it = "unclas" then
        put "UNCLAS" into class
    end if
end mouseUp
```
** CARD #2, BUTTON #15: Addressees ********************************************
on mouseUp
  push card
  go to card addreses
end mouseUp

** CARD #2, BUTTON #16: Standard settings ****************************************
on mouseUp
  push card
  go to card settings
end mouseUp

** CARD #2, BUTTON #17: Print **********************************************
on mouseUp
  set lockscreen to true
  global msgtype,drafter,serno,msgflag,msgflag2,status,draftflag,oldnum
  put msgflag & "_" & serno into filename
  put empty into msgtype
  put empty into drafter
  put empty into msgflag
  put empty into msgflag2
  put empty into status
  put serno into card field ser_no of card settings
  put empty into serno
  put empty into oldnum
  go to card scratch of stack set library
  put "BT" after last char of card field test
  put card field test into tempmsg
  prinField(card field test)
  go to card navy_blue of stack reports
  if tempmsg <> "BT" then
    put filename & return after last char of field listing of card-
    msg file
    set lockscreen to true
    doMenu "new card"
    set the name of this card to filename
    go to card filename
    doMenu "new field"
    set style of card field 1 to opaque
    set rect of card field 1 to 0,0,512,342
    set textfont of card field 1 to courier
    set textsize of card field 1 to 12
    set lockText of card field 1 to true
    doMenu "new field"
    set style of card field 2 to scrolling
    set rect of card field 2 to 1,26,510,280
    set textfont of card field 2 to courier
    set textsize of card field 2 to 12
    doMenu new button
    set icon of card button 1 to 14953
    set rect of card button 1 to 0,303,48,342
set showName of card button 1 to false
set autoHilite of card button 1 to false
set style of card button 1 to transparent
put "on mouseUp" & return & "pop card" & return & "end mouseUp" into tempscript
set script of card button 1 to tempscript
put tempmsg into card field 2
go to card navy_blue
set lockscreen to false
put empty into draftflag
choose browse tool
repeat with j = 1 to 12
  show card field j
end repeat
go to card reports
else
  answer "Message is empty" with "return"
end if
end mouseUp

** CARD #2, BUTTON #18: Cut Tape **********************
on mouseUp
  push card
  go to card cut_tape
end mouseUp

** CARD #2, BUTTON #19: Review **********************
on mouseUp
  push card
  go to card scratch of stack set library
end mouseUp

** CARD #2, BUTTON #20: Cancel **********************
on mouseUp
  global msgtype,drafter,seno,msgflag,msgflag2,status,draftflag,oldnum
  put empty into msgtype
  put empty into drafter
  put empty into seno
  put empty into msgflag
  put empty into msgflag2
  put empty into status
  put empty into draftflag
  put oldnum into card field ser_no of card settings
  put empty into oldnum
  repeat with j = 1 to 12
    show card field j
  end repeat
  set lockScreen to true
  go to card scratch of stack set library
  put empty into card field test
  go to card "navy blue" of stack reports
  pop card
** CARD #2, BUTTON #21: EXIT *********************************************
on mouseUp
  go argos
end mouseUp

** CARD #3, BUTTON #1: New Button *********************************************
on mouseUp
  pop card
end mouseUp
** CARD #4, BUTTON #1: New Button *********************************************
on mouseUp
  pop card
end mouseUp
** CARD #5, BUTTON #1: New Button *********************************************
on mouseUp
  pop card
end mouseUp
** CARD #6: addees *********************************************
on openCard
  select after last char of card field orig
end openCard

** CARD #6, FIELD #1: ORIG *********************************************
on tabKey
  select after last char of card field action
end tabKey
** CARD #6, FIELD #2: ACTION *********************************************
on tabKey
  select after last char of card field "info"
end tabKey
** CARD #6, FIELD #3: info *********************************************
on tabKey
  select after last char of card field "orig"
end tabKey
** CARD #6, BUTTON #1: Return *********************************************
on mouseUp
  play "RETURN"
  pop card
end mouseUp

** CARD #6, BUTTON #2: Additional Addee Info *********************************************
on mouseUp
  push card
  go to card "addee info"
end mouseUp

** CARD #6, BUTTON #3: Enter *********************************************
on mouseUp
  global class

33
set lockScreen to true
put "FROM" & return & card field orig & return & return & "TO" & return & card field action & return & "INFO" & return & card field info & return & "BT" & return & class & return into temp
push card
go to card scratch of stack "set library"
put temp before line 1 of card field test
pop card
pop card
end mouseUp

** CARD #7: add dec info ****************************
on openCard
if "COMNAVAIRSYSCOM WASHINGTON DC" is in card field "info" of card addees then
  hide card field one
else
  show card field one
end if
if "CMC WASHINGTON DC" is in card field "info" of card addees then
  hide card field two
else
  show card field two
end if
end openCard

** CARD #7, BUTTON #1: Return *************************
on mouseUp
  play "RETURN"
  pop card
end mouseUp

** CARD #7, BUTTON #2: airsyscom ****************************
on mouseUp
if "COMNAVAIRSYSCOM WASHINGTON DC" is in card field "info" of card addees then
  answer "That address has already been entered" with "return"
else
  put "COMNAVAIRSYSCOM WASHINGTON DC" & return after last char of card field "info" of card addees
  hide card field one
end if
end mouseUp

** CARD #7, BUTTON #3:cmc ****************************
on mouseUp
if "CMC WASHINGTON DC" is in card field "info" of card addees then
  answer "That address has already been entered" with "return"
else
  put "CMC WASHINGTON DC" & return after last char of card field "info" of card addees
end if

** CARD #7, BUTTON #4: NEXT PAGE ********************************************

```
on mouseUp
  go to card "addee info2"
end mouseUp
```

** CARD #8: ADDEE INFO2 ********************************************

```
on openCard
  if "COMNAVSECINVCOM WASHINGTON DC/22D/" is in card field "info" of card addees then
    hide card field one
  else
    show card field one
  end if
  if "COMSC WASHINGTON DC" is in card field "info" of card addees then
    hide card field two
  else
    show card field two
  end if
  if "NAVXDIVINGSU PANAMA CITY FL" is in card field "info" of card addees then
    hide card field three
  else
    show card field three
  end if
  if "CNO OP ZERO ONE WASHINGTON DC" is in card field "info" of card addees then
    hide card field four
  else
    show card field four
  end if
  if "NAVSAFECEN NORFOLK VA" is in card field "info" of card addees then
    hide card field five
  else
    show card field five
  end if
end openCard
```

** CARD #8, BUTTON #1: Return ********************************************

```
on mouseUp
  play "RETURN"
  pop card
end mouseUp
```

** CARD #8, BUTTON #2: PREV PAGE ********************************************

```
on mouseUp
  go to card "addee info"
end mouseUp
```

35
** CARD #8, BUTTON #3: nis ****************************
on mouseUp
  if "COMNAVSECVCOM WASHINGTON DC//22D//" is in card field "info" of ¬
  card addees then
    answer "That address has already been entered" with "return"
  else
    put "COMNAVSECVCOM WASHINGTON DC//22D//" & return after last char—
    of card field "info" of card addees
    hide card field one
  end if
- - - mouseUp

** CARD #8, BUTTON #4: msc ****************************
on mouseUp
  if "COMSC WASHINGTON DC" is in card field "info" of ¬
  card addees then
    answer "That address has already been entered" with "return"
  else
    put "COMSC WASHINGTON DC" & return after last char—
    of card field "info" of card addees
    hide card field two
  end if
end mouseUp

** CARD #8, BUTTON #5: dive ****************************
on mouseUp
  if "COMNAVSEASYSCOM WASHINGTON DC" is in card field "info" of ¬
  card addees and "NAVXDIVINGSU PANAMA CITY FL" is in card field "info" of ¬
  card addees then
    answer "That address has already been entered" with "return"
    exit mouseUp
  end if
  if "COMNAVSEASYSCOM WASHINGTON DC" is not in card field "info" of ¬
  card addees and "NAVXDIVINGSU PANAMA CITY FL" is not in card field ¬
  "info" of card addees then
    put "COMNAVSEASYSCOM WASHINGTON DC" & return & ¬
    "NAVXDIVINGSU PANAMA CITY FL" & return after last char—
    of card field "info" of card addees
    hide card field three
    exit mouseUp
  end if
  if "COMNAVSEASYSCOM WASHINGTON DC" is in card field "info" of ¬
  card addees and "NAVXDIVINGSU PANAMA CITY FL" is not in card field ¬
  "info" of card addees then
    put "NAVXDIVINGSU PANAMA CITY FL" & return after last char—
    of card field "info" of card addees
    hide card field three
    exit mouseUp
  end if
  if "COMNAVSEASYSCOM WASHINGTON DC" is not in card field "info" of ¬
  card addees and "NAVXDIVINGSU PANAMA CITY FL" is in card field ¬
  "info" of card addees then

36
** CARD #8, BUTTON #6: milpers *************************************
on mouseUp
  if "COMNAVMILPERSCOM WASHINGTON DC" is in card field "info" of card addees
    and "CNO OP ZERO ONE WASHINGTON DC" is in card field "info" of card addees
    then
      answer "That address has already been entered" with "return"
    exit mouseUp
  end if
  if "COMNAVMILPERSCOM WASHINGTON DC" is not in card field "info" of card addees
    and "CNO OP ZERO ONE WASHINGTON DC" is not in card field "info" of card addees
    then
      put "COMNAVMILPERSCOM WASHINGTON DC" & return after last char of card field "info" of card addees
      hide card field four
      exit mouseUp
  end if
  if "COMNAVMILPERSCOM WASHINGTON DC" is in card field "info" of card addees
    and "CNO OP ZERO ONE WASHINGTON DC" is not in card field "info" of card addees
    then
      put "CNO OP ZERO ONE WASHINGTON DC" & return after last char of card field "info" of card addees
      hide card field four
      exit mouseUp
  end if
  if "COMNAVMILPERSCOM WASHINGTON DC" is not in card field "info" of card addees
    and "CNO OP ZERO ONE WASHINGTON DC" is in card field "info" of card addees
    then
      put "COMNAVMILPERSCOM WASHINGTON DC" & return after last char of card field "info" of card addees
      hide card field four
    end if
end mouseUp

** CARD #8, BUTTON #7: safety *************************************
on mouseUp
  if "NAVSAFECEN NORFOLK VA" is in card field "info" of card addees
    then
      answer "That address has already been entered" with "return"
    else
      put "NAVSAFECEN NORFOLK VA" & return after last char of card field "info" of card addees
      hide card field five
    end if
end mouseUp
** CARD #8, BUTTON #8: NEXT PAGE ****************************

on mouseUp
  go to card "addee info3"
end mouseUp

** CARD #9: Addee info3 ****************************

on openCard
  if "COMNAVMEDCOM WASHINGTON DC" is in card field "info" of card addees then
    hide card field one
  else
    show card field one
  end if
  if "NAVINSGEN WASHINGTON DC" is in card field "info" of card addees then
    hide card field two
  else
    show card field two
  end if
  if "COMNAVSEASYSCOM WASHINGTON DC" is in card field "info" of card addees then
    hide card field three
  else
    show card field three
  end if
end openCard

** CARD #9, BUTTON #1: Return ****************************

on mouseUp
  play "RETURN"
  pop card
end mouseUp

** CARD #9, BUTTON #2: PREV PAGE ****************************

on mouseUp
  go to card "addee info2"
end mouseUp

** CARD #9, BUTTON #3: med ****************************

on mouseUp
  if "COMNAVMEDCOM WASHINGTON DC" is in card field "info" of card addees then
    answer "That address has already been entered" with "return"
  else
    put "COMNAVMEDCOM WASHINGTON DC" & return after last char of card field "info" of card addees
    hide card field one
  end if
end mouseUp

** CARD #9, BUTTON #4: ig ****************************

on mouseUp

if "NAVINSGEN WASHINGTON DC" is in card field "info" of card addees then
   answer "That address has already been entered" with "return"
else
   put "NAVINSGEN WASHINGTON DC" & return after last char of card field "info" of card addees
   hide card field two
end if
end mouseUp

** CARD #9, BUTTON #5: navsea

on mouseUp
if "COMNAVSEASYSCOM WASHINGTON DC" is in card field "info" of card addees then
   answer "That address has already been entered" with "return"
else
   put "COMNAVSEASYSCOM WASHINGTON DC" & return after last char of card field "info" of card addees
   hide card field three
end if
end mouseUp

** CARD #9, BUTTON #6: shore com

on mouseUp
ask "Enter PLAD of Major shore commander"
if it is in card field "info" of card addees then
   answer "That address has already been entered" with "return"
else
   if it is not empty then
      put it & return after last char of card field "info" of card addees
   end if
end if
end mouseUp

** CARD #9, BUTTON #7: NEXT PAGE

on mouseUp
  go to card "addee info4"
end mouseUp

** CARD #10, BUTTON #1: Return

on mouseUp
  play "RETURN"
  pop card
end mouseUp

** CARD #10, BUTTON #2: PREV PAGE

on mouseUp
  go to card "addee info3"
end mouseUp

** CARD #10, BUTTON #3: SYSCOM

on mouseUp
  ask "Enter PLAD of SYSCOM"
if it is in card field "info" of card addees then
  answer "That address has already been entered" with "return"
else
  if it is not empty then
    put it & return after last char of card field "info" of card addees
  end if
end if
end mouseUp

** CARD #10, BUTTON #4: CGUARD ************
on mouseUp
  ask "Enter PLAD of Coast Guard District"
  if it is in card field "info" of card addees then
    answer "That addressee has already been entered" with "return"
  else
    if it is not empty then
      put it & return after last char of card field "info" of card addees
    end if
  end if
end mouseUp

** CARD #10, BUTTON #5: SOPA ************
on mouseUp
  ask "Enter PLAD of SOPA"
  if it is in card field "info" of card addees then
    answer "That addressee has already been entered" with "return"
  else
    if it is not empty then
      put it & return after last char of card field "info" of card addees
    end if
  end if
end mouseUp

** CARD #10, BUTTON #6: addee ************
on mouseUp
  ask "Enter PLAD"
  if it is in card field "info" of card addees then
    answer "That addressee has already been entered" with "return"
  else
    if it is not empty then
      put it & return after last char of card field "info" of card addees
    end if
  end if
end mouseUp

** CARD #11: settings ************
on closeCard
  global drafter, serno
  put card field ser_no into serno
  put card field orig into drafter
end closeCard
** CARD #11, BUTTON #1: Return ************
on mouseUp
  play "RETURN"
  pop card
end mouseUp

** CARD #11, BUTTON #2: EXIT ****************************
on mouseUp
  go argos
end mouseUp

** CARD #12, BUTTON #1: Return ****************************
on mouseUp
  pop card
end mouseUp

** CARD #13: msg_file ****************************
on openCard
  show field 1
end openCard
on closeCard
  hide field 1
end closeCard

** CARD #13, BUTTON #1: Print ****************************
on mouseUp
  ask "Enter message name"
  if it is empty then
    exit mouseUp
  end if
  put it into msgname
  go to card msgname
  if it is not empty then
    answer "Message not found" with "Return"
    exit mouseUp
  else
    printField(card field 2)
    go to card msg.file
  end if
end mouseUp

** CARD #13, BUTTON #2: View ****************************
on mouseUp
  ask "Enter message name"
  if it is not in field listing then
    if it is not empty then
      answer it && "is not on file" with "return"
      exit mouseUp
    else
      exit mouseUp
    end if
  else
  end if
end else
push card
go to card it
end if
end mouseUp

** CARD #13, BUTTON #3: Delete  

```plaintext
on mouseUp
ask "Enter message name"
if it is not in field listing then
  if it is not empty then
    answer it && "is not on file" with "return"
    exit mouseUp
  else
    exit mouseUp
  end if
else
  set lockscreen to true
  put it into tempname
  find tempname in field listing
  put the foundLine into temp
  delete temp
  push card
go to card it
doMenu delete card
pop card
set lockscreen to false
end if
end mouseUp
```

** CARD #13, BUTTON #4: Return

```plaintext
on mouseUp
pop card
end mouseUp
```
APPENDIX C. SET LIBRARY STACK SCRIPTS

SCRIPTS FOR STACK: Set library

** STACK SCRIPT **********************************************
on openStack
  hide field field_name
  set textArrows to true
end openStack

** BKGND #1, FIELD #2 **********************************************
on mouseUp
end mouseUp

** CARD #1: ampn **********************************************
on openCard
  show card field ampn_id
  click at 500,200
  hide card field enter
  put "AMPN/" into card field ampn_field
  select after last char of card field ampn_field
end openCard

on idle
  send idle to card field ampn_field
end idle

** CARD #1, FIELD #1: ampn_field **********************************************
on idle
  if "AMPN/" is not in line 1 of card field ampn_field then
    get the length of line 1 of card field ampn_field
    put char 5 to it of line 1 of card field ampn_field into tempstring
    put "AMPN/" & tempstring into line 1 of card field ampn_field
    select after char 5 of line 1 of card field ampn_field
  end if
  if the scroll of card field ampn_field > 0 then
    hide card field ampn_id
  else
    show card field ampn_id
  end if
end idle

on tabKey
end tabKey

** CARD #1, BUTTON #1: Enter **********************************************
on mouseUp
  put "Checking Data" into line 5 of card field enter
  show card field enter
  set cursor to 4
  repeat while the length of last line of card field ampn_field = 0
    delete last line of card field ampn_field
end repeat
end repeat
if the number of lines in card field ampn_field = 1 then
  if offset(return, card field ampn_field) = 6 then
    beep
    hide card field enter
    answer "Field is empty" with "return"
    put "AMPN/" into card field ampn_field
    select after last char of card field ampn_field
    exit mouseUp
  end if
end if
if the length of card field ampn_field < 6 then
  beep
  hide card field enter
  answer "Field is empty" with "return"
  put "AMPN/" into card field ampn_field
  select after last char of card field ampn_field
  exit mouseUp
end if
put offset("/f,card field ampn_field) into temp
if temp <> 0 then
  hide card field enter
  beep
  answer "EOSM in field" with "return"
  if temp = 5 then
    select char 6 of card field ampn_field
    exit mouseUp
  else
    select char temp to temp + 1 of card field ampn_field
    exit mouseUp
  end if
end if
put "Entering Data" into line 5 of card field enter
put word 1 of card field ampn_field into tempstring
repeat with j = 2 to the number of words in card field ampn_field
  if j = the number of words in card field ampn_field then
    put the length of word j of card field ampn_field into lastword
    if the length of tempstring + lastword < 67 then
      put " & word j of card field ampn_field after last char of tempstring
      put tempstring & "/" & return after last char of card test of card id 3502
      put empty into field fieldname
      hide card field enter
      pop card
      exit repeat
    else
      put tempstring & return after last char of card field test of card id 3502
      put word j of card field ampn_field & "/" & return after last char of card field test of card id 3502
    end if
  end if
end repeat
put empty into field field_name
hide card field enter
pop card
exit repeat
end if
end if
put the length of word j of card field ampn_field into wordlength
put the length of tempstring into linelength
if linelength + wordlength < 69 then
  if j = 1 then
    put word j of card field ampn_field after —
    last char of tempstring
    next repeat
  else
    put " " & word j of card field ampn_field after —
    last char of tempstring
    next repeat
  end if
else
  put tempstring & return after last char of card field test —
  of card id 3502
  put empty into tempstring
  put word j of card field ampn_field into tempstring
  next repeat
end if
end repeat
end mouseUp

** CARD #1, BUTTON #2: Cancel ******************************************
on mouseUp
  put empty into card field ampn_field
  put empty into field field_name
  pop card
end mouseUp

** CARD #1, BUTTON #3: Delete ******************************************
on mouseUp
  set cursor to 4
  -- set lockScreen to true
  go to card scratch
  put empty into firstline
  repeat with j = 1 to the number of lines in card field test
    if "ampn/" is in line j of card field test then
      put j into firstline
      put j into lastline
      if offset("/\r",line j of card field test) = 0 then
        repeat with k = j to the number of lines in card field test
          if offset("/\r",line k of card field test) = 0 then
            next repeat
          else
            put k into lastline
          end if
        exit repeat
      end if
    end if
  end repeat
45
end if
end repeat
end if
exit repeat
end repeat
if firuline is not empty then
delete line firsline to lastline of card field test
end if
go to card ampn
end mouseUp

** CARD #2: exer ************************************
on openCard
hide card field enter
put "exercise_name" into field field_name
select after last char of card field exercise_name
end openCard
on idle
if "exercise_name" is in field field_name then
send idle to card field exercise_name
else
send idle to card field addl_id
end if
end idle
** CARD #2, FIELD #2: exercise_name ******************
on mouseEnter
put "exercise_name" into field field_name
select after last char of card field exercise_name
end mouseEnter
on openField
put "exercise_name" into field field_name
end openField
on idle
put the number of chars in card field exercise_name into temp
if temp > 56 then
beep
put char 1 to 56 of card field exercise_name into validstring
put validstring into card field exercise_name
select after last char of card field exercise_name
end if
get offset("/",card field exercise_name)
if it <> 0 then
beep
delete char it of card field exercise_name
select after last char of card field exercise_name
end if
end idle
on tabKey
put "addl_id" into field field_name
select after last char of card field addl_id
end tabKey
** CARD #2, FIELD #3   ***********************************************
on mouseEnter
    put "addlid" into field fieldname
    select after last char of card field addlid
end mouseEnter
on openField
    put "addlid" into field fieldname
end openField
on idle
    put the number of chars in card field addlid into temp
    if temp > 16 then
        beep
        put char 1 to 16 of card field addlid into validstring
        put validstring into card field addlid
        select after last char of card field addlid
    end if
    get offset("/",card field addlid)
    if it <> 0 then
        beep
        delete char it of card field addlid
        select after last char of card field addlid
    end if
end idle
on tabKey
    put "exercise_name" into field fieldname
    select after last char of card field exercise_name
end tabKey
** CARD #2, BUTTON #1: Enter   **********************************************
on mouseUp
    put " Checking Data" into line 5 of card field enter
    show card field enter
    set cursor to 4
    if card field exercise_name is empty then
        hide card field enter
        beep
        answer "Field 1 is MANDATORY. It is empty" with "return"
        put "exercise_name" into field field_name
        select after last char of card field exercise_name
        exit mouseUp
    end if
    if card field addlid is empty then
        hide card field enter
        beep
        answer "Field 2 is MANDATORY. It is empty" with "return"
        put "addlid" into field field_name
        select after last char of card field addlid
        exit mouseUp
    end if
    put " Entering Data" into line 5 of card field enter
    put the length of card field exercise_name into temp1
    put the length of card field addlid into temp2
    if temp1 + temp2 <= 61 then
put "EXER/" & card field exercise_name & "/" & card field addl_id & "/" & return after last char of card field test of card id 3502
else
put "EXER/" & card field exercise_name & return & "/" & card field addl_id & "/" & return after last char of card field test of card id 3502
end if
put empty into field field-name
pop card
end mouseUp

** CARD #2, BUTTON #2: Cancel
**************************************************************************

on mouseUp
put empty into card field exercise_name
put empty into card field addl_id
put empty into field field-name
pop card
end mouseUp

** CARD #2, BUTTON #3: Delete
**************************************************************************

on mouseUp
set cursor to 4
set lockScreen to true
go to card scratch
repeat with j = 1 to the number of lines in card field test
ifs "exer" is in line j of card field test then
if offset("/",line j of card field test) <> 0 then
delete line j of card field test
exit repeat
else
delete line j to j + 1 of card field test
exit repeat
end if
end if
end repeat
go to card exer
end mouseUp

** CARD #3: oper
**************************************************************************

on openCard
hide card field enter
put "codeword" into field field_name
select after last char of card field codeword
end openCard

on idle
if "codeword" is in field field_name then
send idle to card field codeword
end if
if "orig._refno" is in field field_name then
send idle to card field orig._refno
end if
if "nickname" is in field field_name then
send idle to card field nickname
end if
if "nickname2" is in field field-name then
  send idle to card field nickname2
end if
end idle

** CARD #3, FIELD #2: CODEWORD **********************************************
on mouseEnter
  put "codeword" into field field-name
  select after last char of card field codeword
end mouseEnter
on openField
  put "codeword" into field field-name
end openField
on idle
  put the number of chars in card field codeword into temp
  if temp > 32 then
    beep
    put char 1 to 32 of card field codeword into validstring
    put validstring into card field codeword
    select after last char of card field codeword
  end if
  get offset("/",card field codeword)
  if it <> 0 then
    beep
    delete char it of card field codeword
    select after last char of card field codeword
  end if
end idle
on tabKey
  put "orig_refno" into field field-name
  select after last char of card field orig_refno
end tabKey

** CARD #3, FIELD #4: orig_refno **********************************************
on mouseEnter
  put "orig_refno" into field field-name
  select after last char of card field orig_refno
end mouseEnter
on openField
  put "orig_refno" into field field-name
end openField
on idle
  put the number of chars in card field orig_refno into temp
  if temp > 23 then
    beep
    put char 1 to 23 of card field orig_refno into validstring
    put validstring into card field orig_refno
    select after last char of card field orig_refno
  end if
  get offset("/",card field orig_refno)
  if it <> 0 then
    beep
    beep
delete char it of card field orig_refno
select after last char of card field orig_refno
end if
end idle
on tabKey
put "nickname" into field fieldname
select after last char of card field nickname
end tabKey

** CARD #3, FIELD #6: nickname ********************************

on mouseEnter
put "nickname" into field fieldname
select after last char of card field nickname
end mouseEnter
on openField
put "nickname" into field fieldname
end openField
on idle
put the number of chars in card field nickname into temp
if temp > 23 then
    beep
    put char 1 to 23 of card field nickname into validstring
    put validstring into card field nickname
    select after last char of card field nickname
end if
get offset(\",card field nickname)
if it <> 0 then
    beep
    delete char it of card field nickname
    select after last char of card field nickname
end if
end idle
on tabKey
put "nickname2" into field field_name
select after last char of card field nickname2
end tabKey

** CARD #3, FIELD #8: nickname2 ********************************

on mouseEnter
put "nickname2" into field field_name
select after last char of card field nickname2
end mouseEnter
on openField
put "nickname2" into field field_name
end openField
on idle
put the number of chars in card field nickname2 into temp
if temp > 23 then
    beep
    put char 1 to 23 of card field nickname2 into validstring
    put validstring into card field nickname2
    select after last char of card field nickname2
end if
get offset(\",card field nickname2)
if it <> 0 then
beep
delete char it of card field nickname2
select after last char of card field nickname2
end if
end idle
on tabKey
put "codeword" into field fieldname
select after last char of card field codeword
end tabKey
** CARD #3, BUTTON #1: Enter
************
on mouseUp
put " Checking Data" into line 5 of card field enter
show card field enter
set cursor to 4
if card field codeword is empty then
hide card field enter
beep
answer "Field 1 is MANDATORY. It is empty" with "return"
put "codeword" into field fieldname
select after last char of card field codeword
exit mouseUp
end if
if card field orig_refno is empty then
hide card field enter
beep
answer "Field 2 is MANDATORY. It is empty" with "return"
put "orig_refno" into field fieldname
select after last char of card field orig_refno
exit mouseUp
end if
put " Entering Data" into line 5 of card field enter
if card field nickname is empty and card field nickname2 is empty then
put "OPER! & card field codeword & "/" & card field orig_refno & "/" & return after last char of card field test of card id 3502
end if
if card field nickname is not empty and card field nickname2 is empty then
put the length of card field codeword into temp1
put the length of card field orig_refno into temp2
put the length of card field nickname into temp3
if temp1 + temp2 + temp3 <= 60 then
put "OPER/" & card field codeword & "/" & card field orig_refno & "/" & return after last char of card field test of card id 3502
else
put "OPER/" & card field codeword & "/" & card field orig_refno & return after last char of card field test of card id 3502
put "/" & card field nickname & "/" & return after last char of card field test of card id 3502

51
if card field nickname is empty and card field nickname2 is not empty then
  put the length of card field codeword into temp1
  put the length of card field orig_refno into temp2
  put the length of card field nickname2 into temp3
  if temp1 + temp2 + temp3 <= 58 then
    put "OPER/" & card field codeword & "/" & card field orig_refno & "/" & card field nickname2 & "/" & return after last char of card field test of card id 3502
  else
    put "OPER/" & card field codeword & "/" & card field orig_refno & "/" & card field nickname2 & "/" & return after last char of card field test of card id 3502
  end if
end if
if card field nickname is not empty and card field nickname2 is not empty then
  put the length of card field codeword into temp1
  put the length of card field orig_refno into temp2
  put the length of card field nickname into temp3
  put the length of card field nickname2 into temp4
  if temp1 + temp2 + temp3 + temp4 <= 59 then
    put "OPER/" & card field codeword & "/" & card field orig_refno & "/" & card field nickname2 & "/" & card field nickname & "/" & return after last char of card field test of card id 3502
  else
    if temp1 + temp2 + temp3 <= 62 then
      put "OPER/" & card field codeword & "/" & card field orig_refno & "/" & card field nickname & "/" & return after last char of card field test of card id 3502
    else
      put "OPER/" & card field codeword & "/" & card field orig_refno & return after last char of card field test of card id 3502
    end if
  end if
end if
put empty into field field_name
hide card field enter
pop card
end mouseUp
** CARD #3, BUTTON #2: Cancel ****************************
on mouseUp
  put empty into card field codeword
  put empty into card field orig_refno
  put empty into card field nickname
  put empty into card field nickname2
  put empty into field field_name
  pop card
end mouseUp

** CARD #3, BUTTON #3: Delete ****************************
on mouseUp
  set cursor to 4
  set lockScreen to true
  go to card scratch
  repeat with j = 1 to the number of lines in card field test
    if "oper/" is in line j of card field test then
      if offset("/",line j of card field test) <> 0 then
        delete line j of card field test
        exit repeat
      else
        delete line j to j + 1 of card field test
        exit repeat
      end if
    end if
  end repeat
  go to card oper
end mouseUp

** CARD #4 ****************************
on openCard
  hide card field enter
  global msgtype, drafter, serno
  put msgtype into card field msg_type
  put drafter into card field originator
  put serno into card field ser_no
  get the long date
  put char 1 to 3 of word 2 of it into tempmonth
  repeat with j = 2 to 3
    put numToChar(charToNum(char j of tempmonth) - 32) into char j of tempmonth
  end repeat
  put tempmonth into card field month
  put "msg_type" into field field_name
  select after last char of card field msg_type
end openCard
on idle
  if "msg_type" is in field field_name then
    send idle to card field msg_type
  end if
  if "originator" is in field field_name then
    send idle to card field originator
end if
if "ser_no" is in field field_name then
  send idle to card field ser_no
end if
if "month" is in field field_name then
  send idle to card field month
end if
end idle
** CARD #4, FIELD #2: msg_type ******************************************************************************
on mouseEnter
  put "msg_type" into field field_name
  select after last char of card field msg_type
end mouseEnter
on openField
  put "msg_type" into field field_name
  select after last char of card field msg_type
end openField
on idle
  put the number of chars in card field msg_type into temp
  if temp > 20 then
    beep
    put char 1 to 20 of card field msg_type into validstring
    put validstring into card field msg_type
    select after last char of card field msg_type
  end if
  get offset("/",card field msg_type)
  if it <> 0 then
    beep
    delete char it of card field msg_type
    select after last char of card field msg_type
  end if
end idle
on tabKey
  put "originator" into field fieldname
  select after last char of card field originator
end tabKey
** CARD #4, FIELD #4: ORIGINATOR ******************************************************************************
on mouseEnter
  put "originator" into field field_name
  select after last char of card field originator
end mouseEnter
on openField
  put "originator" into field field_name
  select after last char of card field originator
end openField
on idle
  put the number of chars in card field originator into temp
  if temp > 20 then
    beep
    put char 1 to 20 of card field originator into validstring
    put validstring into card field originator
    select after last char of card field originator
  end if
  54
end if
get offset("/",card field originator)
if it <> 0 then
    beep
    delete char it of card field originator
    select after last char of card field originator
end if
end idle
on tabKey
    put "ser_no" into field field_name
    select after last char of card field ser_no
end tabKey
** CARD #4, FIELD #6: ser_no *****************************************************
on mouseEnter
    put "ser_no" into field field_name
    select after last char of card field ser_no
end mouseEnter
on openField
    put "ser_no" into field field_name
    select after last char of card field ser_no
end openField
on idle
    put the number of chars in card field ser_no into temp
    if temp > 7 then
        beep
        put char 1 to 7 of card field ser_no into validstring
        put validstring into card field ser_no
        select after last char of card field ser_no
    end if
get offset("/",card field ser_no)
if it <> 0 then
    beep
    delete char it of card field ser_no
    select after last char of card field ser_no
end if
end idle
on tabKey
    put "month" into field field_name
    select after last char of card field month
end tabKey
** CARD #4, FIELD #8: MONTH *****************************************************
on mouseEnter
    put "month" into field field_name
    select after last char of card field month
end mouseEnter
on openField
    put "month" into field field_name
    select after last char of card field month
end openField
on idle
    put the number of chars in card field month into temp
    if temp > 3 then

beep
put char 1 to 3 of card field month into validstring
put validstring into card field month
select after last char of card field month
end if
get offset("/",card field month)
if it <> 0 then
  beep
  delete char it of card field month
  select after last char of card field month
end if
end idle
on tabKey
put "msg_type" into field field_name
select after last char of card field msg_type
end tabKey
** CARD #4, BUTTON #1: Enter **********************************************/
on mouseUp
  put "Checking Data" into line 5 of card field enter
  show card field enter
  set cursor to 4
  if card field msg_type is empty then
    hide card field enter
    beep
    answer "Field 1 is MANDATORY. It is empty" with "return"
    put "msg_type" into field field_name
    select after last char of card field msg_type
    exit mouseUp
  end if
  if card field originator is empty then
    hide card field enter
    beep
    answer "Field 2 is MANDATORY. It is empty" with "return"
    put "originator" into field field_name
    select after last char of card field originator
    exit mouseUp
  end if
  if card field ser_no is empty then
    hide card field enter
    beep
    answer "Field 3 is MANDATORY. It is empty" with "return"
    put "ser_no" into field field_name
    select after last char of card field ser_no
    exit mouseUp
  end if
  if card field month is empty then
    hide card field enter
    beep
    answer "Field 4 is MANDATORY. It is empty" with "return"
    put "month" into field field_name
    select after last char of card field month
    exit mouseUp
end if
put " Entering Data" into line 5 of card field enter
put "MSGID/" & card field msg_type & "/" & card field originator & -
/ & card field ser_no & "/" & card field month & "/" & return -
after last char of card field test of card id 3502
put empty into field field_name
pop card
end mouseUp

** CARD #4, BUTTON #2: Cancel *****************************************
on mouseUp
  put empty into card field msg_type
  put empty into card field originator
  put empty into card field ser_no
  put empty into card field month
  put empty into field fieldname
  pop card
end mouseUp

** CARD #4, BUTTON #3: Delete ********************************************
on mouseUp
  set lockScreen to true
  go to card scratch
  repeat with j = 1 to the number of lines in card field test
    if "msgid/" is in line j of card field test then
      delete line j of card field test
      exit repeat
    end if
  end repeat
  go to card msgid
end mouseUp

** CARD #5: ref ******************************************
on openCard
  hide card field enter
  if field fieldname is empty then
    put "serial_ltr" into field fieldname
    select after last char of card field serial_ltr
  else
    put field fieldname into temp
    select after last char of card field temp
  end if
end openCard

on idle
  if "serial_ltr" is in field fieldname then
    send idle to card field serial_ltr
  end if
  if "msg_type" is in field fieldname then
    send idle to card field msg_type
  end if
  if "originator" is in field fieldname then
    send idle to card field originator
end idle
end if
if "day_month" is in field field_name then
send idle to card field day_month
end if
if "ser_no" is in field field_name then
send idle to card field ser_no
end if
if "spec_notation" is in field field_name then
send idle to card field spec_notation
end if
end idle
** CARD #5, FIELD #2: serial_ltr *************************************
on mouseEnter
   put "serial_ltr" into field field_name
   select after last char of card field serial_ltr
end mouseEnter
on openField
   put "serial_ltr" into field field_name
end openField
on idle
   if the number of chars in card field serial_ltr > 1 then
      beep
      put char 1 of card field serial_ltr into validstring
      put validstring into card field serial_ltr
      select after last char of card field serial_ltr
   end if
   if card field serial_ltr is not empty then
      get card field serial_ltr
      if it is not in "ABCDEFGHIJKLMNOPQRSTUVWXYZ" then
         beep
         put empty into card field serial_ltr
         select after last char of card field serial_ltr
      end if
   end if
end idle
on tabKey
   put "msg_type" into field field_name
   select after last char of card field msg_type
end tabKey
** CARD #5, FIELD #4: msg_type *************************************
on mouseEnter
   put "msg_type" into field field_name
   select after last char of card field msg_type
end mouseEnter
on openField
   put "msg_type" into field field_name
end openField
on idle
   put the number of chars in card field msg_type into temp
   if temp > 20 then
      beep
      put char 1 to 20 of card field msg_type into validstring
put validstring into card field msg_type
select after last char of card field msg_type
end if
get offset(",", card field msg_type)
if it <> 0 then
    beep
    delete char it of card field msg_type
    select after last char of card field msg_type
end if
end idle
on tabKey
    put "originator" into field fieldname
    select after last char of card field originator
end tabKey
** CARD #5, FIELD #6: originator *********************************************
on mouseEnter
    put "originator" into field fieldname
    select after last char of card field originator
end mouseEnter
on openField
    put "originator" into field fieldname
end openField
on idle
    put the number of chars in card field originator into temp
    if temp > 20 then
        beep
        put char 1 to 20 of card field originator into validstring
        put validstring into card field originator
        select after last char of card field originator
    end if
    get offset(",", card field originator)
    if it <> 0 then
        beep
        delete char it of card field originator
        select after last char of card field originator
    end if
end idle
on tabKey
    put "day_month" into field fieldname
    select after last char of card field day_month
end tabKey
** CARD #5, FIELD #8: day_month ***********************************************
on mouseEnter
    put "day_month" into field fieldname
    select after last char of card field day_month
end mouseEnter
on openField
    put "day_month" into field fieldname
end openField
on idle
    put the number of chars in card field day_month into temp
    if temp > 12 then
beep
put char 1 to 12 of card field day_month into validstring
put validstring into card field day_month
select after last char of card field day_month
end if
get offset("/",card field day_month)
if it <> 0 then
    beep
    delete char it of card field day_month
    select after last char of card field day_month
end if
end idle
on tabKey
    put "ser_no" into field field_name
    select after last char of card field ser_no
end tabKey
** CARD #5, FIELD #10: ser_no *****************************
on mouseEnter
    put "ser_no" into field field_name
    select after last char of card field ser_no
end mouseEnter
on openField
    put "ser_no" into field field_name
end openField
on idle
    put the number of chars in card field ser_no into temp
    if temp > 7 then
        beep
        put char 1 to 7 of card field ser_no into validstring
        put validstring into card field ser_no
        select after last char of card field ser_no
    end if
    get offset("/",card field ser_no)
    if it <> 0 then
        beep
        delete char it of card field ser_no
        select after last char of card field ser_no
    end if
end idle
on tabKey
    put "spec_notation" into field field_name
    select after last char of card field spec_notation
end tabKey
** CARD #5, FIELD #12: spec_notation *****************************
on mouseEnter
    put "spec_notation" into field field_name
    select after last char of card field spec_notation
end mouseEnter
on openField
    put "spec_notation" into field field_name
end openField
on idle
put the number of chars in card field spec_notation into temp
if temp > 5 then
  beep
  put char 1 to 5 of card field spec_notation into validstring
  put validstring into card field spec_notation
  select after last char of card field spec_notation
end if
get offset("/",card field spec_notation)
if it <> 0 then
  beep
  delete char it of card field spec_notation
  select after last char of card field spec_notation
end if
end idle
on tabKey
  put "serial_ltr" into field field_name
  select after last char of card field serial_ltr
end tabKey
** CARD #5, BUTTON #1: Continue ********************************************
on mouseUp
  put card field serial_ltr into card field serial_ltr of card ref2
  put card field msg_type into card field msg_type of card ref2
  put card field originator into card field originator of card ref2
  put card field day_month into card field day_month of card ref2
  put card field ser_no into card field ser_no of card ref2
  put card field spec_notation into card field spec_notation of card ref2
  put field field_name into field field_name of card ref2
  go to card ref2
end mouseUp

** CARD #5, BUTTON #2: Enter ***********************************************
on mouseUp
  put "Checking Data" into line 5 of card field enter
  show card field enter
  set cursor to 4
  if card field serial_ltr is empty then
    hide card field enter
    beep
    answer "Field 1 is MANDATORY. It is empty" with "return"
  put "serial_ltr" into field field_name
  select after last char of card field serial_ltr
  exit mouseUp
end if
  if card field msg_type is empty then
    hide card field enter
    beep
    answer "Field 2 is MANDATORY. It is empty" with "return"
  put "msg_type" into field field_name
  select after last char of card field msg_type
  exit mouseUp
end if
if card field originator is empty then
    hide card field enter
    beep
    answer "Field 3 is MANDATORY. It is empty" with "return"
    put "originator" into field fieldname
    select after last char of card field originator
    exit mouseUp
end if
if card field day_month is empty then
    hide card field enter
    beep
    answer "Field 4 is MANDATORY. It is empty" with "return"
    put "day-month" into field fieldname
    select after last char of card field day_month
    exit mouseUp
end if
put "Entering Data" into line 5 of card field enter
if card field ser_no is empty and card field spec_notation is empty then
    put "REF/" & card field serial_ltr & "/" & card field msg_type & "/
    " & card field originator & "/" & card field day_month & "/" & 
    return after last char of card field test of card id 3502
    put empty into field field name
    hide card field enter
    pop card
    exit mouseUp
end if
if card field ser_no is not empty and card field spec_notation is empty then
    put the length of card field msg_type into temp1
    put the length of card field originator into temp2
    put the length of card field day_month into temp3
    put the length of card field ser_no into temp4
    if temp1 + temp2 + temp3 + temp4 <= 58 then
        put "REF/" & card field serial_ltr & "/" & card field msg_type & "/
        " & card field originator & "/" & card field day_month & "/" & 
        card field ser_no & "/" & return after last char of card field test of card id 3502
        put empty into field field name
        hide card field enter
        pop card
        exit mouseUp
    else
        put "REF/" & card field serial_ltr & "/" & card field msg_type & "/
        " & card field originator & "/" & card field day_month & return
        after last char of card field test of card id 3502
        put "/" & card field ser_no & "/" & return after last char 
        of card field test of card id 3502
        put empty into field field name
        hide card field enter
        pop card
        exit mouseUp
    end if
else
    put "REF/" & card field serial_ltr & "/" & card field msg_type & "/
    " & card field originator & "/" & card field day_month & return
    after last char of card field test of card id 3502
    put "/" & card field ser_no & "/" & return after last char 
    of card field test of card id 3502
    put empty into field field name
    hide card field enter
    pop card
    exit mouseUp
end if
end if
end if
if card field ser_no is empty and card field spec_notation is not empty then
  put "REF/" & card field serial_ltr & "/" & card field 
  msg_type & "/" & card field originator & "/" & card field 
  day_month & "/" & "/" & "/" & card field spec_notation & "/" & return 
  & return after last char of card field test of card id 3502
  put empty into field field_name
  hide card field enter
  pop card
  exit mouseUp
else
  put "REF/" & card field serial_ltr & "/" & card field msg_type & 
  "/" & card field originator & "/" & card field day_month & 
  "/" & card field ser_no & "/" & card field spec_notation & 
  "/" & return 
  after last char of card field test of card id 3502
  put empty into field field_name
  hide card field enter
  pop card
  exit mouseUp
end if
end mouseUp

** CARD #5, BUTTON #3: Cancel ******************************************
on mouseUp
  put empty into card field serial_ltr
  put empty into card field msg_type
  put empty into card field originator
  put empty into card field day_month
  put empty into card field ser_no
  put empty into card field spec_notation
  put empty into card field serial_ltr of card ref2
  put empty into card field msg_type of card ref2
  put empty into card field originator of card ref2
  put empty into card field day_month of card ref2
  put empty into card field ser_no of card ref2
  put empty into card field spec_notation of card ref2
  pop card
end mouseUp

** CARD #5, BUTTON #4: Delete ******************************************
on mouseUp
  set cursor to 4
  set lockScreen to true
  go to card scratch
  repeat with j = 1 to the number of lines in card field test
    if "reft" is in line j of card field test then
      if offset("/", line j of card field test) <> 0 then
        delete line j of card field test
      exit repeat
    exit repeat
63
else
    delete line j to j + 1 of card field test
end if
end if
end repeat
go to card ref
end mouseUp

** CARD #6: ref2 ******************************************************
on openCard
    hide card field enter
    if field field_name is empty then
        put "serial_ltr" into field field_name
        select after last char of card field serial_ltr
    else
        put field field_name into temp
        select after last char of card field temp
    end if
end openCard
on idle
    if "serial_ltr" is in field field_name then
        send idle to card field serial_ltr
    end if
    if "msg_type" is in field field_name then
        send idle to card field msg_type
    end if
    if "originator" is in field field_name then
        send idle to card field originator
    end if
    if "day_month" is in field field_name then
        send idle to card field day_month
    end if
    if "ser_no" is in field field_name then
        send idle to card field ser_no
    end if
    if "spec_notation" is in field field_name then
        send idle to card field spec_notation
    end if
end idle

** CARD #6, FIELD #2: serial_ltr ********************************************
on mouseEnter
    put "serial_ltr" into field field_name
    select after last char of card field serial_ltr
end mouseEnter
on openField
    put "serial_ltr" into field field_name
end openField
on idle
    if the number of chars in card field serial_ltr > 1 then
        beep
        put char 1 of card field serial_ltr into validstring
put validstring into card field serial_ltr
select after last char of card field serial_ltr
end if
if card field serial_ltr is not empty then
get card field serial_ltr
if it is not in "ABCDEFHJKLMNPRSTUWVXYZ" then
    beep
    put empty into card field serial_ltr
    select after last char of card field serial_ltr
end if
end if
end idle

on tabKey
    put "msg_type" into field field_name
    select after last char of card field msg_type
end tabKey

** CARD #6, FIELD #4: msg_type *************************************************
on mouseEnter
    put "msg_type" into field field_name
    select after last char of card field msg_type
end mouseEnter
on openField
    put "msg_type" into field field_name
end openField
on idle
    put the number of chars in card field msg_type into temp
    if temp > 20 then
        beep
        put char 1 to 20 of card field msg_type into validstring
        put validstring into card field msg_type
        select after last char of card field msg_type
    end if
    get offset(/,card field msg_type)
    if it <= 0 then
        beep
        delete char it of card field msg_type
        select after last char of card field msg_type
    end if
end idle

on tabKey
    put "originator" into field field_name
    select after last char of card field originator
end tabKey

** CARD #6, FIELD #6: originator ***********************************************
on mouseEnter
    put "originator" into field field_name
    select after last char of card field originator
end mouseEnter
on openField
    put "originator" into field field_name
end openField
on idle
put the number of chars in card field originator into temp
if temp > 20 then
    beep
    put char 1 to 20 of card field originator into validstring
    put validstring into card field originator
    select after last char of card field originator
end if
get offset("\",card field originator)
if it <> 0 then
    beep
    delete char it of card field originator
    select after last char of card field originator
end if
end idle
on tabKey
    put "day_month" into field fieldname
    select after last char of card field day_month
end tabKey
** CARD #6, FIELD #8: day_month ********************************************
on mouseEnter
    put "day_month" into field fieldname
    select after last char of card field day_month
end mouseEnter
on openField
    put "day_month" into field fieldname
end openField
on idle
    put the number of chars in card field day_month into temp
    if temp > 12 then
        beep
        put char 1 to 12 of card field day_month into validstring
        put validstring into card field day_month
        select after last char of card field day_month
    end if
    get offset("\",card field day_month)
    if it <> 0 then
        beep
        delete char it of card field day_month
        select after last char of card field day_month
    end if
end idle
on tabKey
    put "ser_no" into field fieldname
    select after last char of card field ser_no
end tabKey
** CARD #6, FIELD #10: ser_no **********************************************
on mouseEnter
    put "ser_no" into field fieldname
    select after last char of card field ser_no
end mouseEnter
on openField
    put "ser_no" into field fieldname
end openField
on idle
  put the number of chars in card field ser_no into temp
  if temp > 7 then
    beep
    put char 1 to 7 of card field ser_no into validstring
    put validstring into card field ser_no
    select after last char of card field ser_no
  end if
  get offset("/", card field ser_no)
  if it <> 0 then
    beep
    delete char it of card field ser_no
    select after last char of card field ser_no
  end if
end idle
on tabKey
  put "spec_notation" into field fieldname
  select after last char of card field spec_notation
end tabKey
** CARD #6, FIELD #12: spec_notation
--------------------------------------------------------------------------------
on mouseEnter
  put "spec_notation" into field fieldname
  select after last char of card field spec_notation
end mouseEnter
on openField
  put "spec_notation" into field fieldname
end openField
on idle
  put the number of chars in card field spec_notation into temp
  if temp > 5 then
    beep
    put char 1 to 5 of card field spec_notation into validstring
    put validstring into card field spec_notation
    select after last char of card field spec_notation
  end if
  get offset("/", card field spec_notation)
  if it <> 0 then
    beep
    delete char it of card field spec_notation
    select after last char of card field spec_notation
  end if
end idle
on tabKey
  put "serial_ltr" into field fieldname
  select after last char of card field serial_ltr
end tabKey
** CARD #6, BUTTON #1: Continue
--------------------------------------------------------------------------------
on mouseUp
  put card field serial_ltr into card field serial_ltr of card ref
  put card field msg_type into card field msg_type of card ref
  put card field originator into card field originator of card ref

67
put card field day_month into card field day_month of card ref
put card field ser_no into card field ser_no of card ref
put card field spec_notation into card field spec_notation of card ref
put field field_name into field field_name of card ref
go to card ref
eend mouseUp

** CARD #6, BUTTON #2: Cancel ****************************
on mouseUp
  put empty into card field serial_ltr
  put empty into card field msg_type
  put empty into card field originator
  put empty into card field day_month
  put empty into card field ser_no
  put empty into card field spec_notation
  put empty into card field serial_ltr of card ref2
  put empty into card field msg_type of card ref2
  put empty into card field originator of card ref2
  put empty into card field day_month of card ref2
  put empty into card field ser_no of card ref2
  put empty into card field spec_notation of card ref2
  put empty into card field_name
  pop card
end mouseUp

** CARD #6, BUTTON #3: Enter ****************************
on mouseUp
  put "Checking Data" into line 5 of card field enter
  show card field enter
  set cursor to 4
  if card field serial_ltr is empty then
    hide card field enter
    beep
    answer "Field 1 is MANDATORY. It is empty" with "return"
    put "serial_ltr" into field fieldname
    select after last char of card field serial_ltr
    exit mouseUp
  end if
  if card field msg_type is empty then
    hide card field enter
    beep
    answer "Field 2 is MANDATORY. It is empty" with "return"
    put "msg_type" into field fieldname
    select after last char of card field msg_type
    exit mouseUp
  end if
  if card field originator is empty then
    hide card field enter
    beep
    answer "Field 3 is MANDATORY. It is empty" with "return"
    put "originator" into field fieldname
    select after last char of card field originator
exit mouseUp
end if
if card field day_month is empty then
    hide card field enter
    beep
    answer "Field 4 is MANDATORY. It is empty" with "return"
    put "day_month" into field field_name
    select after last char of card field day_month
    exit mouseUp
end if
put "Entering Data" into line 5 of card field enter
if card field ser_no is empty and card field spec_notation is empty then
    put "REF/ & card field serial_ltr & "/ & card field msg_type & "/ & card field originator & "/ & card field day_month & "/ & card field ser_no & "/ & return after last char of card field test of card id 3502
    put empty into field field_name
    hide card field enter
    pop card
    exit mouseUp
end if
if card field ser_no is not empty and card field spec_notation is empty then
    put the length of card field msg_type into temp1
    put the length of card field originator into temp2
    put the length of card field day_month into temp3
    put the length of card field ser_no into temp4
    if temp1 + temp2 + temp3 + temp4 <= 58 then
        put "REF/ & card field serial_ltr & "/ & card field msg_type & "/ & card field originator & "/ & card field day_month & "/ & card field ser_no & "/ & return after last char of card field test of card id 3502
        put empty into field field_name
        hide card field enter
        pop card
        exit mouseUp
    end if
else
    put "REF/ & card field serial_ltr & "/ & card field msg_type & "/ & card field originator & "/ & card field day_month & return after last char of card field test of card id 3502
    put "/ & card field ser_no & "/ & return after last char of card field test of card id 3502
    put empty into field field_name
    hide card field enter
    pop card
    exit mouseUp
end if
end if
day_month & "/" & ":" & "/" & card field spec_notation & "/" & return after last char of card field test of card id 3502
put empty into field field_name
hide card field enter
pop card
exit mouseUp
else
put "REF" & card field serial_ltr & "/" & card field msg_type & "/" & card field originator & "/" & card field day_month & "/" & card field ser_no & "/" & card field spec_notation & "/" & return after last char of card field test of card id 3502
put empty into field field_name
hide card field enter
pop card
exit mouseUp
end if
end mouseUp

** CARD #6, BUTTON #4: Delete ************************************************************
on mouseUp
set cursor to 4
set lockScreen to true
go to card scratch
repeat with j = 1 to the number of lines in card field test
if "reff" is in line j of card field test then
if offset("/",line j of card field test) <> 0 then
delete line j of card field test
exit repeat
else
delete line j to j + 1 of card field test
exit repeat
end if
end if
end repeat
go to card ref
end mouseUp

** CARD #7: flagword *************************************************
on openCard
hide card field enter
global msgflag
put msgflag into card field flagword
global msgflag2
put msgflag2 into card field flagword2
put "flagword" into field field_name
select after last char of card field flagword
end openCard
on idle
if "flagword" is in field field_name then
send idle to card field flagword
end if
if "flagword2" is in field field_name then
send idle to card field flagword2
end if
end idle

** CARD #7, FIELD #2: FLAGWORD  *********************************************
on mouseEnter
  put "flagword" into field field_name
  select after last char of card field flagword
end mouseEnter
on openField
  put "flagword" into field field_name
end openField
on idle
  put the number of chars in card field flagword into temp
  if temp > 8 then
    beep
    put char 1 to 8 of card field flagword into validstring
    put validstring into card field flagword
  end if
  get offset("/", card field flagword)
  if it <> 0 then
    beep
    delete char it of card field flagword
  end if
end idle
on tabKey
  put "flagword2" into field field_name
  select after last char of card field flagword2
e end tabKey

** CARD #7, FIELD #4: flagword2 *********************************************
on mouseEnter
  put "flagword2" into field field_name
  select after last char of card field flagword2
end mouseEnter
on openField
  put "flagword2" into field field_name
end openField
on idle
  put the number of chars in card field flagword2 into temp
  if temp > 21 then
    beep
    put char 1 to 21 of card field flagword2 into validstring
    put validstring into card field flagword2
    select after last char of card field flagword2
  end if
  get offset("/", card field flagword2)
  if it <> 0 then
    beep
    delete char it of card field flagword2
  end if
end idle
end idle
on tabKey
  put "flagword" into field field_name
  select after last char of card field flagword
end tabKey
** CARD #7, BUTTON #1: Enter ********************************************************************************
on mouseUp
  put "Checking Data" into line 5 of card field enter
  show card field enter
  set cursor to 4
  if card field flagword is empty then
    hide card field enter
    beep
    answer "Field 1 is MANDATORY. It is empty" with "return"
    put "flagword" into field field_name
    select after last char of card field flagword
    exit mouseUp
  end if
  if card field flagword2 is empty then
    hide card field enter
    beep
    answer "Field 2 is MANDATORY. It is empty" with "return"
    put "flagword2" into field field_name
    select after last char of card field flagword2
    exit mouseUp
  end if
  put "Entering Data" into line 5 of card field enter
  put "FLAGWORD/" & card field flagword & "/" & card field flagword2 & "/" & return after last char of card field test of card id 3502
  put empty into field field_name
  hide card field enter
  pop card
  exit mouseUp
end mouseUp

** CARD #7, BUTTON #2: Cancel ********************************************************************************
on mouseUp
  put empty into card field flagword
  put empty into card field flagword2
  put empty into field field_name
  pop card
end mouseUp

** CARD #7, BUTTON #3: Delete ********************************************************************************
on mouseUp
  set lockScreen to true
  go to card scratch
  repeat with j = 1 to the number of lines in card field test
    if "flagword/" is in line j of card field test then
      delete line j of card field test
      exit repeat
    end if
  end repeat

72
end repeat
go to card flagword
end mouseUp

** CARD #8: timeloc ******************************************
on openCard
  global status
  put status into card field report_status
  hide card field enter
  put "day_time" into field field_name
  select after last char of card field day_time
end openCard

on idle
  if "day_time" is in field field_name then
    send idle to card field day_time
  end if
  if "location" is in field field_name then
    send idle to card field location
  end if
  if "report_status" is in field field_name then
    send idle to card field report_status
  end
end idle

** CARD #8, FIELD #2: day_time *******************************
on mouseEnter
  put "day_time" into field field_name
  select after last char of card field day_time
end mouseEnter

on openField
  put "day_time" into field field_name
end openField

on idle
  put the number of chars in card field day_time into temp
  if temp > 7 then
    beep
    put char 1 to 7 of card field day_time into validstring
    put validstring into card field day_time
    select after last char of card field day_time
  end if
  get offset("/",card field day_time)
  if it <> 0 then
    beep
    delete char it of card field day_time
    select after last char of card field day_time
  end if
end idle

on tabKey
  put "location" into field field_name
  select after last char of card field location
end tabKey

** CARD #8, FIELD #4: location *******************************
on mouseEnter
  put "location" into field field_name
select after last char of card field location
end mouseEnter
on openField
  put "location" into field field_name
end openField
on idle
  put the number of chars in card field location into temp
  if temp > 20 then
    beep
    put char 1 to 20 of card field location into validstring
    put validstring into card field location
    select after last char of card field location
  end if
  get offset("/", card field location)
  if it <> 0 then
    beep
    delete char it of card field location
    select after last char of card field location
  end if
end idle
on tabKey
  put "report-status" into field field_name
  select after last char of card field report-status
end tabKey
** CARD #8, FIELD #6: report-status ****************************
on mouseEnter
  put "report-status" into field field_name
  select after last char of card field report-status
end mouseEnter
on openField
  put "report-status" into field field_name
end openField
on idle
  put the number of chars in card field report-status into temp
  if temp > 5 then
    beep
    put char 1 to 5 of card field report-status into validstring
    put validstring into card field report-status
    select after last char of card field report-status
  end if
  get offset("/", card field report-status)
  if it <> 0 then
    beep
    delete char it of card field report-status
    select after last char of card field report-status
  end if
end idle
on tabKey
  put "day_time" into field field_name
  select after last char of card field day_time
end tabKey
** CARD #8, BUTTON #1: Enter ****************************
on mouseUp
   put "Checking Data" into line 5 of card field enter
   show card field enter
   set cursor to 4
   if card field day_time is empty then
      hide card field enter
      beep
      answer "Field 1 is MANDATORY. It is empty" with "return"
      put "day_time" into field field_name
      select after last char of card field day_time
      exit mouseUp
   end if
   if the length of card field day_time <> 7 then
      hide card field enter
      beep
      answer "Date-Time group must be 7 characters" with "return"
      put "day_time" into field field_name
      select after last char of card field day_time
      exit mouseUp
   end if
   if card field location is empty then
      hide card field enter
      beep
      answer "Field 2 is MANDATORY. It is empty" with "return"
      put "location" into field field_name
      select after last char of card field location
      exit mouseUp
   end if
   if card field report_status is empty then
      hide card field enter
      beep
      answer "Field 3 is MANDATORY. It is empty" with "return"
      put "report_status" into field field_name
      select after last char of card field report_status
      exit mouseUp
   else
      if card field report_status <> "INIT" and card field report_status <> "FINAL" then
         hide card field enter
         beep
         answer "Field 3 entry Must be INIT', FOLUP or FINAL'" with "return"
         put "report_status" into field field_name
         select char 1 to 5 of card field report_status
         exit mouseUp
      end if
   end if
   put "Entering Data" into line 5 of card field enter
   put "TIMELOC/" & card field day_time & "/" & card field location & "/" & card field report_status & "/" & return after last char of card id 3502
put empty into field field_name
hide card field enter
pop card
end mouseUp

** CARD #8, BUTTON #2: Cancel *********************************************
on mouseUp
  put empty into card field day_time
  put empty into card field location
  put empty into card field report_status
  put empty into field field_name
  pop card
end mouseUp

** CARD #8, BUTTON #3: Delete *********************************************
on mouseUp
  set lockScreen to true
  go to card scratch
  repeat with j = 1 to the number of lines in card field test
    if "timeloc/" is in line j of card field test then
      delete line j of card field test
    exit repeat
  end if
end rep at
  go to card timeloc
end mouseUp

** CARD #9: narr *********************************************
on openCard
  hide card field enter
  show card field narr_id
  click at 500,200
  put "NARR/" into card field narr_field
  put "narr_field" into field field_name
  select after last char of card field narr_field
end openCard
on idle
  send idle to card field narr_field
end idle
** CARD #9, FIELD #1: narr_field *********************************************
on idle
  if "NARR/" is not in line 1 of card field 1 then
    get the length of line 1 of card field 1
    put char 5 to it of line 1 of card field 1 into tempstring
    put "NARR/" & tempstring into line 1 of card field 1
    select after char 5 of line 1 of card field 1
  end if
  repeat with j = 1 to the number of lines in card field 1
    GET the length of line j of card field 1
    if it > 69 then
      beep
      put char 1 to 69 of line j of card field 1 into ¬
line j of card field 1
answer "Line is longer than 69 characters" with "return"
select after char 69 of line j of card field 1
exit repeat
end if
end repeat
end idle
on tabKey
end tabKey
** CARD #9, BUTTON #1: Enter *************************************************************
on mouseUp
put "Checking Data" into line 5 of card field enter
show card field enter
set cursor to 4
repeat while the length of last line of card field narr_field = 0
delete last line of card field narr_field
end repeat
if the number of lines in card field narr_field = 1 then
if offset(return,card field narr_field) = 6 then
beep
hide card field enter
answer "Field is empty" with "return"
put "NARR/" into card field narr_field
select after last char of card field narr_field
exit mouseUp
end if
end if
if the length of card field narr_field < 6 then
beep
hide card field enter
answer "Field is empty" with "return"
put "NARR/" into card field narr_field
select after last char of card field narr_field
exit mouseUp
end if
put offset("/",card field narr_field) into temp
if temp <> 0 then
hide card field enter
beep
answer "EOSM in field" with "return"
if temp = 5 then
select char 6 of card field narr_field
exit mouseUp
else
select char temp to temp + 1 of card field narr_field
exit mouseUp
end if
end if
put "Entering Data" into line 5 of card field enter
put the number of lines in card field narr_field into temp
if temp > 1 then
if the length of line temp of card field narr_field > 67 then
put the number of words in line temp of card field narr_field →
into temp1
put line 1 to temp - 1 of card field narr_field & return after →
last char of card field "test" of card id 3502
put word 1 to temp1 - 1 of line temp of card field narr_field & →
return after last char of card field test of card id 3502
put word temp1 of line temp of card field narr_field & "/f" & →
return after last char of card field test of card id 3502
else
put card field narr_field & "/f" & return →
after last char of card field test of card id 3502
end if
else
if the length of card field narr_field > 67 then
put the number of words in card field narr_field into temp1
put word 1 to temp1 - 1 of card field narr_field & return →
after last char of card field test of card id 3502
put word temp1 of card field narr_field & "/f" & return →
after last char of card field test of card id 3502
else
put card field narr_field & "/f" & return →
after last char of card field test of card id 3502
end if
end if
put empty into field field_name
hide card field enter
pop card
end mouseUp

** CARD #9, BUTTON #2: Cancel ********************************
on mouseUp
put empty into card field narr_field
put empty into field field_name
pop card
end mouseUp

** CARD #9, BUTTON #3: Delete ********************************
on mouseUp
set cursor to 4
set lockScreen to true
go to card scratch
put empty into firstline
repeat with j = 1 to the number of lines in card field test
if "narr/" is in line j of card field test then
put j into firstline
put j into lastline
if offset("/f", line j of card field test) = 0 then
repeat with k = j to the number of lines in card field test
if offset("/f", line k of card field test) = 0 then
next repeat
else
put k into lastline
exit repeat
end if
end repeat
end if
exit repeat
end if
end repeat
if firstline is not empty then
delete line firstline to lastline of card field test
end if
go to card narr
end mouseUp

** CARD #10: gentext ************************************
on openCard
  hide card field enter
  if the mouseV > 197 and the mouseV < 278 then
    put "gentext_field" into field field_name
    select after last char of card field gentext_field
  else
    put "text_indicator" into field field_name
    select after last char of card field text_indicator
  end if
  global msgtype
  if msgtype is "OPREP-3" then
    put "INCIDENT IDENTIFICATION AND DETAILS" into card field-
    text_indicator
    select after last char of card field gentext_field
  end if
end openCard
on idle
  if field field_name contains "text_indicator" then
    send idle to card field text_indicator
  else
    send idle to card field gentext_field
  end if
end idle

** CARD #10, FIELD #2: text_indicator **********************************
on mouseEnter
  put "text_indicator" into field field_name
  select after last char of card field text_indicator
end mouseEnter
on openField
  put "text_indicator" into field field_name
end openField'd
on idle
  put card field text_indicator into tempstring
  get the length of tempstring
  if it > 58 then
    beep
  put char 1 to 58 of me into validstring
put validstring into me
select after char 58 of me
end if
get offset(","tempstring)
if it <> 0 then
beep
delete char it of card field text_indicator
select after last char of card field text_indicator
end if
end idle
on returnInField
tabKey
end returnInField
on tabkey
put "gentext_field" into field fieldname
if card field gentext_field = return then
select before last char of card field gentext_field
else
select after last char of card field gentext_field
end if
end tabkey

** CARD #10, FIELD #4: gentext_field
******************************************************************************
on mouseDown
beep
beep
send mouseUp to card "gentext"
end mouseDown
on mouseEnter
put "gentext_field" into field fieldname
if card field gentext_field = return then
select before last char of card field gentext_field
else
select after last char of card field gentext_field
end if
end mouseEnter
on idle
end idle
on tabkey
put "text_indicator" into field fieldname
select after last char of card field text_indicator
tabkey

** CARD #10, BUTTON #1: Enter
******************************************************************************
on mouseUp
put "Checking Data" into line 5 of card field enter
show card field enter
set cursor to 4
if card field text_indicator is empty then
hide card field enter
beep
answer "Field 1 is MANDATORY It is empty" with "return"
put "text_indicator" into field field_name
select after last char of card field text_indicator
exit mouseUp
end if
repeat
if the number of lines in card field gentext_field > 1 then
if last line of card field gentext_field is empty then
    delete last line of card field gentext_field
else
    exit repeat
end if
else
if the length of line 1 of card field gentext_field <= 1 then
    if card field gentext_field = return or "-
    card field gentext_field is empty then
        beep
        hide card field enter
        answer "Field 2 is MANDATORY It is empty" with "return"
        select before last char of card field gentext_field
        exit mouseUp
        exit repeat
    end if
    end if
    exit repeat
end if
end repeat
put offset("//",card field gentext_field) into temp
if temp <> 0 then
    hide card field enter
    beep
    answer "EOSM in field" with "return"
    select char temp to temp + 1 of card field gentext_field
    exit mouseUp
end if
put " " Entering Data" into line 5 of card field enter
put "GENTEXT/" & card field text_indicator & "/" into tempstring
repeat with j = 1 to the number of words in card field gentext_field
if j = the number of words in card field gentext_field then
    put the length of word j of card field gentext_field into lastword
    if the length of tempstring + lastword < 67 then
        put " " & word j of card field gentext_field after last char of tempstring
        put tempstring & "/" & return after last char of card field test of card id 3502
        put empty into field field_name
        hide card field enter
        pop card
        exit repeat
    else
        beep
        answer "Field 3 is MANDATORY It is empty" with "return"
        select before last char of card field gentext_field
        exit mouseUp
        exit repeat
    end if
    exit repeat
end if
end repeat
81
put tempstring & return after last char of card field test of card id 3502
put word j of card field gentext_field & "/" & return after —
last char of card field test of card id 3502
put empty into field field_name
hide card field enter
pop card
exit repeat
end if
end if
put the length of word j of card field gentext_field into wordlength
put the length of tempstring into linelength
if linelength + wordlength < 69 then
  if j = 1 then
    put word j of card field gentext_field after —
    last char of tempstring
  next repeat
else
  put " " & word j of card field gentext_field after —
  last char of tempstring
  next repeat
end if
else
  put tempstring & return after last char of card field test —
  of card id 3502
  put empty into tempstring
  put word j of card field gentext_field into tempstring
  next repeat
end if
end repeat
end if
end if
end mouseUp

** CARD #10, BUTTON #2: Cancel

on mouseUp
  put empty into card field text_indicator
  put empty into card field gentext_field
  put empty into field field_name
  hide card field enter
  pop card
end mouseUp

** CARD #10, BUTTON #3: Delete

on mouseUp
  set cursor to 4
  set lockScreen to true
  go to card scratch
  put empty into firstline
  repeat with j = 1 to the number of lines in card field test
    if "gentext/" is in line j of card field test then
      put j into firstline
      put j into lastline
if offset("/", line j of card field test) = 0 then
    repeat with k = j to the number of lines in card field test
        if offset("/", line k of card field test) = 0 then
            next repeat
        else
            put k into lastline
            exit repeat
        end if
    end repeat
end if
end repeat

if firstline is not empty then
    delete line firstline to lastline of card field test
end if

go to card gentext
end mouseUp

** CARD #11: rmks ************************************
on openCard
    hide card field enter
    show card field rmks_id
    click at 500,200
    put "RMKS/" into card field rmks_field
    put "rmks_field" into field fieldname
    select after last char of card field rmks_field
end openCard

on idle
    send idle to card field rmks_field
end idle

** CARD #11, FIELD #1: rmks_field ************************************
on idle
    if "RMKS/" is not in line 1 of card field 1 then
        get the length of line 1 of card field 1
        put char 5 to it of line 1 of card field 1 into tempstring
        put "RMKS/" & tempstring into line 1 of card field 1
        select after char 5 of line 1 of card field 1
    end if
    repeat with j = 1 to the number of lines in card field 1
        GET the length of line j of card field 1
        if it > 69 then
            beep
            put char 1 to 69 of line j of card field 1 into -
            line j of card field 1
            answer "Line is longer than 69 characters" with "return"
            select after char 69 of line j of card field 1
            exit repeat
        end if
    end repeat
end idle

on tabKey
** CARD #11, BUTTON #1: Enter ****************************************
on mouseUp
  put "Checking Data" into line 5 of card field enter
  show card field enter
  set cursor to 4
  repeat while the length of last line of card field rmks_field = 0
    delete last line of card field rmks_field
  end repeat
  if the number of lines in card field rmks_field = 1 then
    if offset(return,card field rmks_field) = 6 then
      beep
      hide card field enter
      answer "Field is empty" with "return"
      put "RMKS/" into card field rmks_field
      select after last char of card field rmks_field
      exit mouseUp
    end if
  end if
  if the length of card field rmks_field < 6 then
    beep
    hide card field enter
    answer "Field is empty" with "return"
    put "RMKS/" into card field rmks_field
    select after last char of card field rmks_field
    exit mouseUp
  end if
  put offset("/",card field rmks_field) into temp
  if temp <= 0 then
    hide card field enter
    beep
    answer "EOSM in field" with "return"
    if temp = 5 then
      select char 6 of card field rmks_field
      exit mouseUp
    else
      select char temp to temp + 1 of card field rmks_field
      exit mouseUp
    end if
  end if
  put "Entering Data" into line 5 of card field enter
  put the number of lines in card field rmks_field into temp
  if temp > 1 then
    if the length of line temp of card field rmks_field > 67 then
      put the number of words in line temp of card field rmks_field into temp1
      put line 1 to temp - 1 of card field rmks_field & return after
      last char of card field "test" of card id 3502
      put word 1 to temp1 - 1 of line temp of card field rmks_field &
      return after last char of card field test of card id 3502
      put word temp1 of line temp of card field rmks_field & "/" &
      return after last char of card field test of card id 3502
  end if

84
else
  put card field rmks_field & "/" & return —
  after last char of card field test of card id 3502
end if
else
  if the length of card field rmks_field > 67 then
    put the number of words in card field rmks_field into temp1
    put word 1 to temp1 - 1 of card field rmks_field & return —
    after last char of card field test of card id 3502
    put word temp1 of card field rmks_field & "/" & return —
    after last char of card field test of card id 3502
  else
    put card field rmks_field & "/" & return —
    after last char of card field test of card id 3502
  end if
end if
end if
put empty into field field-name
hide card field enter
pop card
end mouseUp

** CARD #11, BUTTON #2: Cancel **********************************************
on mouseUp
  put empty into card field rmks_field
  put empty into field field-name
  pop card
end mouseUp

** CARD #11, BUTTON #3: Delete **************************************
on mouseUp
  set cursor to 4
  set lockScreen to true
  go to card scratch
  put empty into firstline
  repeat with j = 1 to the number of lines in card field test
    if "rmks/" is in line j of card field test then
      put j into firstline
      put j into lastline
      if offset("/",line j of card field test) = 0 then
        repeat with k = j to the number of lines in card field test
          if offset("/",line k of card field test) = 0 then
            next repeat
          else
            put k into lastline
            exit repeat
          end if
        end repeat
      end if
    end if
  end repeat
if firstline is not empty then
** CARD #12: close text ****************************

on openCard
    hide card field enter
    put "decl_inst" into field field_name
    select after last char of card field decl_inst
end openCard

on idle
    send idle to card field decl_inst
end idle

** CARD #12, FIELD #2: decl_inst ****************************

on mouseEnter
    put "decl_inst" into field field_name
    select after last char of card field decl_inst
end mouseEnter

on openField
    put "decl_inst" into field field_name
end openField

on idle
    put the number of chars in card field decl_inst into temp
    if temp > 58 then
        beep
        put char 1 to 58 of card field decl_inst into validstring
        put validstring into card field decl_inst
        select after last char of card field decl_inst
    end if
    get offset("/",card field decl_inst)
    if it <> 0 then
        beep
        delete char it of card field decl_inst
        select after last char of card field decl_inst
    end if
end idle

** CARD #12, BUTTON #1: Enter ****************************

on mouseUp
    put " Checking Data" into line 5 of card field enter
    show card field enter
    set cursor to 4
    if card field decl_inst is empty then
        hide card field enter
        beep
        answer "Field is empty" with "return"
        put "decl_inst" into field field_name
        select after last char of card field decl_inst
        exit mouseUp
    end if
    put " Entering Data" into line 5 of card field enter
put "CLOSTEXT/" & card field decl_inst & "/" & return after last char of card field test of card id 3502
put empty into field field_name
hide card field enter
pop card
end mouseUp

** CARD #12, BUTTON #2: Cancel **********************************************
on mouseUp
put empty into card field decl_inst
put empty into field field_name
pop card
end mouseUp

** CARD #12, BUTTON #3: Delete **********************************************
on mouseUp
set lockScreen to true
go to card scratch
repeat with j = 1 to the number of lines in card field test
if "clostext/" is in line j of card field test then
delete line j of card field test
exit repeat
end if
end repeat
go to card clostext
end mouseUp

** CARD #13: decl **********************************************
on openCard
hide card field enter
put "decl_inst" into field field_name
select after last char of card field decl_inst
end openCard
on idle
send idle to card field decl_inst
end idle

** CARD #13, FIELD #2: decl_inst **********************************************
on mouseEnter
put "decl_inst" into field field_name
select after last char of card field decl_inst
end mouseEnter
on openField
put "decl_inst" into field field_name
end openField
on idle
put the number of chars in card field decl_inst into temp
if temp > 25 then
beep
put char 1 to 25 of card field decl_inst into validstring
put validstring into card field decl_inst
select after last char of card field decl_inst
end if
get offset("/", card field decl_inst)
if it <= 0 then
  beep
  delete char it of card field decl_inst
  select after last char of card field decl_inst
end if
end idle

** CARD #13, BUTTON #1: Enter **************************************************************************
on mouseUp
  put "Checking Data" into line 5 of card field enter
  show card field enter
  set cursor to 4
  if card field decl_inst is empty then
    beep
    hide card field enter
    answer "Field is empty" with "return"
    put "decl/" into field field_name
    select after last char of card field decl_inst
    exit mouseUp
  end if
  put "Entering Data" into line 5 of card field enter
  put "DECL/ & card field decl_inst & "/" & return after last char " of card field test of card id 3502
  put empty into field field_name
  pop card
end mouseUp

** CARD #13, BUTTON #2: Cancel **************************************************************************
on mouseUp
  put empty into card field decl_inst
  put empty into field field_name
  pop card
end mouseUp

** CARD #13, BUTTON #3: Delete **************************************************************************
on mouseUp
  set lockScreen to true
  go to card scratch
  repeat with j = 1 to the number of lines in card field test
    if "decl/" is in line j of card field test then
      delete line j of card field test
      exit repeat
    end if
  end repeat
  go to card decl
end mouseUp

** CARD #14, FIELD #1: test **************************************************************************
on idle
  if "AMPN/" is not in line 1 of card field 1 then
    get the length of line 1 of card field 1

88
put char 5 to it of line 1 of card field 1 into tempstring
put "AMPN" & tempstring into line 1 of card field 1
select after char 5 of line 1 of card field 1
end if
repeat with j = 1 to the number of lines in card field 1
GET the length of line j of card field 1
if it > 69 then
  beep
  put char 1 to 69 of line j of card field 1 into -
  line j of card field 1
  answer "Line is longer than 69 characters" with "return"
  select after char 69 of line j of card field 1
  exit repeat
end if
end repeat
end repeat
end idle

** CARD #14, BUTTON #1: Cancel ************ ::: ****************************************
on mouseUp
  put empty into field field_name
  pop card
end mouseUp
APPENDIX D. TRAINING STACK SCRIPTS

SCRIPTS FOR STACK: Training

**STACK SCRIPT********************************

function validDate date
    put date into tempdate
    if the length of tempdate < 6 or the length of tempdate > 8 then
        return false
    else
        if the length of tempdate = 6 then
            if char 1 of tempdate is not in "123456789" then
                return false
            end if
            if char 2 of tempdate <> "/" then
                return false
            end if
            if char 3 of tempdate is not in "123456789" then
                return false
            end if
            if char 4 of tempdate <> "/" then
                return false
            end if
            if char 5 of tempdate is not in "1234567890" then
                return false
            end if
            if char 6 of tempdate is not in "1234567890" then
                return false
            end if
        end if
        if the length of tempdate = 7 then
            if char 3 of tempdate = "" then
                if char 1 of tempdate is not in "12" then
                    return false
                end if
                if char 2 of tempdate is not in "012" then
                    return false
                end if
                if char 4 of tempdate is not in "123456789" then
                    return false
                end if
                if char 5 of tempdate <> "" then
                    return false
                end if
                if char 6 of tempdate is not in "1234567890" then
                    return false
                end if
            end if
        end if
    end if
end if
end if
if char 7 of tempdate is not in "1234567890" then
    return false
end if
else
    if char 2 of tempdate = "/" then
        if char 1 of tempdate is not in "123456789" then
            return false
        end if
        if char 3 of tempdate is not in "123" then
            return false
        end if
        if char 4 of tempdate is not in "1234567890" then
            return false
        end if
        if char 5 of tempdate <> "/" then
            return false
        end if
        if char 6 of tempdate is not in "1234567890" then
            return false
        end if
        if char 7 of tempdate is not in "1234567890" then
            return false
        end if
    end if
    if char 1 of tempdate = 2 then
        if char 3 of tempdate = 3 then
            return false
        end if
        if char 3 of tempdate = 2 then
            if char 4 of tempdate = 9 then
                put char 6 to 7 of tempdate into year
                if year mod 4 <> 0 then
                    return false
                end if
            end if
        else
            if char 3 of tempdate = 2 then
                if char 4 of tempdate = 9 then
                    put char 6 to 7 of tempdate into year
                    if year mod 4 <> 0 then
                        return false
                    end if
                end if
            end if
        end if
    end if
    if char 1 of tempdate is in "469" then
        if char 3 of tempdate = 3 then
            if char 4 of tempdate <> 0 then
                return false
            end if
        end if
        if char 3 of tempdate = 3 then
            if char 4 of tempdate > 1 then
                return false
            end if
        end if
    end if
end if
end if
if the length of tempdate = 8 then
  if char 3 of tempdate <> "1" then
    return false
  end if
  if char 6 of tempdate <> "8" then
    return false
  end if
  if char 1 of tempdate <> 1 then
    return false
  end if
  if char 2 of tempdate is not in "012" then
    return false
  end if
  if char 4 of tempdate is not in "123" then
    return false
  end if
  if char 5 of tempdate is not in "1234567890" then
    return false
  end if
  if char 7 of tempdate is not in "1234567890" then
    return false
  end if
  if char 8 of tempdate is not in "1234567890" then
    return false
  end if
  if char 2 of tempdate = 1 then
    if char 4 of tempdate = 3 then
      if char 5 of tempdate <> 0 then
        return false
      end if
    end if
  end if
end if
end if
end if
end if
return true
end validDate

function validQtr qtr
  if the length of qtr <> 4 then
    return false
  end if
  if char 1 of qtr is not in "1234" then
    return false
  end if
  if char 2 of qtr is not in "-\-" then
    return false
  end if
if char 3 of qtr is not in "1234567890" then
    return false
end if
if char 4 of qtr is not in "1234567890" then
    return false
end if
return true
end validQtr

function goodDate start, date
    convert date to seconds
    convert start to seconds
    if date < start then
        return false
    end if
    convert date to dateItems
    convert start to dateItems
    if item 2 of date > item 2 of start + 2 then
        return false
    end if
    if item 1 of date > item 1 of start then
        return false
    end if
    return true
end goodDate

function expDate compDate, period
    convert compDate to dateItems
    put item 2 of compDate + period into month
    repeat until month < 13
        put month - 12 into month
        put item 1 of compDate + 1 into item 1 of compDate
    end repeat
    put month into item 2 of compDate
    if item 2 of compDate = 2 and item 3 of compDate > 29 then
        put "29" into item 3 of compDate
    end if
    convert compDate to short date
    return compDate
end expDate

function milDate date
    if date is empty then
        return empty
    end if
    convert date to abbr date
    if the length of item 2 of date = 6 then
        put "0" & char 6 of item 2 of date into temp
    else
        put char 6 to 7 of item 2 of date into temp
    end if
    put space & char 2 of item 2 of date after last char of temp
    put numToChar(charToNum(char 3 of item 2 of date) - 32) --
    after last char of temp
    put numToChar(charToNum(char 4 of item 2 of date) - 32) --
after last char of temp
put space & char 4 to 5 of item 3 of date after last char of temp
return temp
end milDate
function msgDate date
if date is empty then
  return empty
end if
convert date to dateItems
put char 3 to 4 of item 1 of date into temp
if the length of item 2 of date = 1 then
  put "0" & item 2 of date after last char of temp
else
  put item 2 of date after last char of temp
end if
if the length of item 3 of date = 1 then
  put "0" & item 3 of date after last char of temp
else
  put item 3 of date after last char of temp
end if
return temp
end msgDate
function convertQtr qtr
put char 1 of qtr * 3 - 2 into tempdate
put "/" & 1 after last char of tempdate
put "/" & char 3 to 4 of qtr after last char of tempdate
return tempdate
end convertQtr

** BACKGROUND #1: Operations ****************************
on openStack
  hide message box
  show menuBar
  pass openStack
end openStack

** CARD #1, BUTTON #1: return *******************************
on mouseUp
  go to operations
end mouseUp

** CARD #1, BUTTON #2: exit *******************************
on mouseUp
  go argos
end mouseUp

** CARD #1, BUTTON#3:Training
******************************************************************************
on mouseDown
  put "PQS,New Schedule,Record Accomplishment,Modify Schedule,Delete Schedule,Draw Chart" into menu1
put return & "STR-TRADA,View-Update Data base,Draft Trarep" after menu1
put return & "Lesson Plans" after menu1
get HPopupMenu(menu1,0,80,65)
if it is not zero then
  put Item 1 of it into TheLine
  put Item 2 of it into ThItem
  If TheLine = 1 and ThItem = 2 then
    go to card newsked
  end if
  If TheLine = 1 and ThItem = 3 then
    go to card record
  end if
  If TheLine = 1 and ThItem = 4 then
    go to card modify
  end if
  If TheLine = 1 and ThItem = 5 then
    go to card skedfile
  end if
  if TheLine = 1 and ThItem = 6 then
    go to card draw
  end if
  if TheLine = 2 and ThItem = 2 then
    go to card view_data
  end if
  if TheLine = 2 and ThItem = 3 then
    go to card trarep
  end if
end if
end mouseDown

** CARD #2: newsked **************************************
on openCard
  put card field start into oldstart
  put empty into card field title
  put empty into card field start
  put empty into card field sked_box
  ask "Title of New schedule"
  if it is not empty then
    put it into card field title
  repeat until validQtr(it)
  ask "Enter calendar quarter of schedule"
  if validQtr(it) then
    put convertQtr(it) into card field start
    select after last char of card field enter_box
    exit repeat
  else
    ask "Enter calendar quarter of schedule"
    if validQtr(it) then
      put convertQtr(it) into card field start
      select after last char of card field enter_box
      exit repeat
    end if
end if
end repeat
if oldstart <> card field start then
send mouseUp to card button draw
end if
select after last char of card field enter_box
else
send mouseUp to card button "return"
end if
end openCard

** CARD #2, FIELD #4: enter_box

on tabKey
send mouseUp to card button "enter_info"
end tabKey

** CARD #2, BUTTON #1: Save Sked

on mouseUp
if card field sked_box is empty then
answer "No schedule to save" with "return"
exit mouseUp
else
set lockMessages to true
set lockScreen to true
put card field start into startdate
put card field title into skedtitle
put card field title & " " into skedname
put the length of card field start into len
if len = 6 then
put (char 1 of card field start + 2)/3 after last char of skedname
put char len - 2 to len of card field start after last char of " skedname
go to card skedname
else
put (char 1 to 2 of card field start + 2)/3 after last char " of skedname
put char len - 2 to len of card field start after last char of " skedname
go to card skedname
end if
if the result is empty then
beep
exit mouseUp
else
if it is "Cancel" then
send mouseUp to card button "change title"
exit mouseUp
else
if it is "Change title" then
send mouseUp to card button "change title"
exit mouseUp
else
go to card skedname
put card field sked_box of card newsked into card field sked
put skedtitle into card field title
put startdate into card field start
go to card newsked
put empty into card field title
put empty into card field sked_box
exit mouseUp
end if
end if
set lockscreen to true
doMenu "new card"
set the name of this card to skedname
go to card skedname
doMenu "new field"
set name of card field 1 to "sked"
set style of card field 1 to scrolling
set rect of card field 1 to 1,26,510,280
set textfont of card field 1 to courier
set fontsize of card field 1 to 12
doMenu "new field"
set name of card field 2 to "title"
set style of card field 2 to transparent
set rect of card field 2 to 0,6,220,23
set textfont of card field 2 to courier
set fontsize of card field 2 to 12
doMenu "new field"
set name of card field 3 to "start"
set style of card field 3 to transparent
set rect of card field 3 to 427,6,509,23
set textfont of card field 3 to courier
set fontsize of card field 3 to 12
put card field sked_box of card newsked into card field sked
put skedtitle into card field title
put startdate into card field start
choose browse tool
put empty into card field title of card newsked
put empty into card field sked_box of card newsked
go to card skedfile
put skedname & return after last char of card field listing
set lockMessages to false
go to card newsked
set lockscreen to false
select after last char of card field enter_box
end if
end mouseUp

** CARD #2, BUTTON #2: Delete Sked **********************
on mouseUp
  if card field sked_box is empty then
    answer "No schedule to delete" with "return"
  else
    put empty into card field enter_box
**CARD #2, BUTTON #3: Return**

on mouseUp
  answer "Unsaved schedule will be lost" with "OK" or "Return"
  if it is "return" then
    select after last char of card field enter_box
    exit mouseUp
  end if
  put empty into card field title
  put empty into card field sked_box
  put empty into card field start
  send openCard to card newsked
end mouseUp

**CARD #2, BUTTON #4: enter_info**

on mouseUp
  if card field title is empty then
    answer "Schedule must have a title" with "Cancel" or "Title"
    if it is "cancel" then
      exit mouseUp
    else
      send mouseUp to card button "change title"
    end if
  end if
  if card field start is empty then
    answer "No schedule quarter entered" with "Cancel" or "Enter"
    if it is "cancel" then
      exit mouseUp
    else
      send mouseUp to card button "change qtr"
      send mouseUp to card button "draw"
    end if
  end if
  if line 1 of card field enter_box is empty then
    ask "No lesson name, Enter lesson name"
    if it is empty then
      put empty into card field enter_box
      select after last char of card field enter_box
      exit mouseUp
    end if
    put it into line 1 of card field enter_box
  end if
  put line 1 of card field enter_box & "," after last char of tempname
  put line 2 of card field enter_box into it
  repeat until validDate(it) is true and
goodDate(card field start, it) is true
if validDate(it) is false then
    ask it && "Not a valid date, Enter correct date"
end if
if it is empty then
    select after last char of card field enter_box
    exit mouseUp
end if
if goodDate(card field start, it) is false then
    ask "Date must be in sked quarter, enter date"
end if
if it is empty then
    select after last char of card field enter_box
    exit mouseUp
end if
end repeat
if the number of lines in card field sked_box >= 15 then
    answer "Schedule is full (15 Items max)" with "Cancel" or "Save sked"
    if it is "cancel" then
        exit mouseUp
    else
        send mouseUp to card button "save sked"
        exit mouseUp
    end if
end if
put it after last char of tempname
put tempname & ",s" & return after last char of card field sked_box
put empty into card field enter_box
end mouseUp

** CARD #2, BUTTON #5: Delete Item ******************************************

on mouseUp
if card field sked_box is empty then
    answer "No lessons to delete" with "return"
    exit mouseUp
end if
ask "Enter lesson name"
if it is empty then
    select after last char of card field enter_box
    exit mouseUp
else
    put it into tgt
    repeat with j = 1 to the number of lines in card field sked_box
    if tgt = item 1 of line j of card field sked_box then
        delete line j of card field sked_box
        select after last char of card field enter_box
        exit mouseUp
    end if
    if j = the number of lines in card field sked_box then
        answer "Lesson not found" with "Return"
    end if
end repeat
select after last char of card field enter_box
end if
end repeat
end if
end mouseUp

** CARD #2, BUTTON #6: Change Title

on mouseUp
ask "Enter new title"
if it is empty then
select after last char of card field enter_box
exit mouseUp
else
put it into card field title
select after last char of card field enter_box
end if
end mouseUp

** CARD #2, BUTTON #7: Change Qtr

on mouseUp
ask "Enter new Quarter (Q/YY)"
if it is empty then
select after last char of card field enter_box
exit mouseUp
else
if validQtr(it) then
put char 1 of it * 3 - 2 into tempdate
put " /" & 1 after last char of tempdate
put " /" & char 3 to 4 of it after last char of tempdate
put tempdate into card field start
select after last char of card field enter_box
exit mouseUp
else
repeat until validQtr(it)
ask "Invalid. Enter calendar quarter of schedule"
if it is empty then
select after last char of card field enter_box
exit mouseUp
end if
if validQtr(it) then
put char 1 of it * 3 - 2 into tempdate
put " /" & 1 after last char of tempdate
put " /" & char 3 to 4 of it after last char of tempdate
put tempdate into card field start
select after last char of card field enter_box
exit mouseUp
end if
end repeat
end if
end if
end mouseUp
** CARD #2, BUTTON #8: DRAW ********************************************

on mouseUp
  if card field start is empty then
    exit mouseUp
  end if
  set cursor to 4
  show card field working
  set lockScreen to true
  choose pencil tool
  click at 50,110
  click at 400,100
  choose select tool
  drag from 0,80 to 190,290
  type "x" with commandKey
  drag from 360,80 to 513,200
  type "x" with commandKey
  choose text tool
  set textSize to 9
  set textFont to geneva
  set textAlign to left
  set textStyle to plain
  set textHeight to 9
  click at 49,103
  type "S"
  click at 63,103
  type "M"
  click at 79,103
  type "T"
  click at 94,103
  type "W"
  click at 109,103
  type "T"
  click at 126,103
  type "F"
  click at 140,103
  type "S"
  put card field start into temp
  convert temp to long date
  click at 48,91
  type word 1 of item 2 of temp & item 3 of temp
  convert temp to dateItems
  put item 2 of temp into mon
  put 115 into lineLoc
  repeat while item 2 of temp = mon
    put item 7 of temp into loc
    click at 31 + 15 * loc, lineLoc
    if item 3 of temp > 9 then
      type item 3 of temp
    else
      type " " & item 3 of temp
    end if
  end repeat
  if loc = 7 then
put lineloc + 13 into lineloc
end if
convert temp into seconds
put 8640C + temp into temp
convert temp to dateltems
end repeat
convert temp to long date
click at 48,204
type word 1 of item 2 of temp & item 3 of temp
convert temp to dateltems
put item 2 of temp into mon
put 216 into lineloc
repeat while item 2 of temp = mon
put item 7 of temp into loc
click at 31 + 15 * loc,lineloc
if item 3 of temp > 9 then
type item 3 of temp
else
type " " & item 3 of temp
end if
if loc = 7 then
put lineloc + 13 into lineloc
end if
convert temp to seconds
put 86400 + temp into temp
convert temp to dateltems
end repeat
click at 387,103
type "S"
click at 401,103
type "M"
click at 417,103
type "T"
click at 432,103
type "W"
click at 447,103
type "T"
click at 464,103
type "F"
click at 478,103
type "S"
convert temp to long date
click at 386,91
type word 1 of item 2 of temp & item 3 of temp
convert temp to dateltems
put item 2 of temp into mon
put 115 into lineloc
repeat while item 2 of temp = mon
put item 7 of temp into loc
click at 369 + 15 * loc,lineloc
if item 3 of temp > 9 then
type item 3 of temp
else
    type " & item 3 of temp
end if
if loc = 7 then
    put lineloc + 13 into lineloc
end if
convert temp to seconds
put 86400 + temp into temp
convert temp to datetems
end repeat
choose browse tool
hide card field working
end mouseUp

** CARD #3, BUTTON #1: Return ****************************
on mouseUp
    go to card training
end mouseUp

** CARD #3, BUTTON #2: Delete Schedule ****************************
on mouseUp
    set lockScreen to true
    set lockMessages to true
    if card field listing is empty then
        answer "No schedules on file" with "Return"
        exit mouseUp
    end if
    ask "Enter name of schedule to delete"
    if it is empty then
        exit mouseUp
    else
        put "card" && quote & it & quote into tempname
        go to tempname
    end if
    if the result is not empty then
        repeat until the result is empty
            go to card "skedfile"
            ask "Schedule not found, Enter schedule name"
            if it is empty then
                exit mouseUp
            end if
            put "card" && quote & it & quote into tempname
            go to tempname
        end repeat
    end if
    doMenu "delete card"
    go to card skedfile
    repeat with j = 1 to the number of lines in card field listing
        if it is in line j of card field listing then
            delete line j of card field listing
        end if
    exit repeat
end if
end repeat
end mouseUp

** CARD #4: record ******************************************************
on openCard
put card field start into oldstart
put card field listing of card skedfile into card field listing
show card field listing
hide card field title
set lockScreen to true
set lockMessages to true
ask "Enter schedule name"
if it is empty then
  hide card field listing
  show card field title
  send mouseUp to card button "return"
  exit openCard
else
  put quote & it & quote into tempname
  put tempname into card field skedname
  put "card" & quote & it & quote into tempname
  go to tempname
end if
if the result is not empty then
  repeat until the result is empty
    go to card "record"
    ask "Schedule not found, Enter schedule name"
    if it is empty then
      hide card field listing
      show card field title
      send mouseUp to card button "return"
      exit openCard
    end if
    put quote & it & quote into tempname
    put tempname into card field skedname
    put "card" & quote & it & quote into tempname
    go to tempname
  end repeat
end if
put the short id of this card into card field card_id of card record
put card field sked into card field sked_box of card record
put card field title into card field title of card record
put card field start into card field start of card record
go to card "record"
set lockScreen to false
if oldstart < card field start then
  send mouseUp to card button draw
end if
hide card field listing
show card field title
set lockMessages to false
set it after last line of card field enter_box
end openCard
** CARD #4, FIELD #1: enter_box ***********************************************
on tabKey
  send mouseUp to card button "enter_info"
end tabKey
** CARD #4, BUTTON #1: Return ***********************************************
on mouseUp
  answer "Unsaved changes will be lost" with "OK" or "Return"
  if it is "return" then
    exit mouseUp
  end if
  put empty into card field skedbox
  put empty into card field title
  put empty into card field enterbox
  go to card training
end mouseUp

** CARD #4, BUTTON #2: enter_info *********************************************
on mouseUp
  if line 1 of card field enterbox is empty then
    ask "No lesson name, Enter lesson name"
    if it is empty then
      put empty into card field enterbox
      select after last char of card field enterbox
      exit mouseUp
    end if
    put it into line 1 of card field enterbox
  end if
  put line 1 of card field enterbox into lessonname
  put line 1 of card field enterbox & ",," after last char of tempname
  put line 2 of card field enterbox into it
  repeat until validDate(it) is true and goodDate(card field startit) is true
    if validDate(it) is false then
      ask it & & "Not a valid date, Enter correct date"
    end if
    if it is empty then
      select after last char of card field enterbox
      exit mouseUp
    end if
  if goodDate(card field startit) is false then
    ask "Date must be in sked quarter, enter date"
  end if
  if it is empty then
    select after last char of card field enterbox
    exit mouseUp
  end if
end repeat
  put it into tempdate
  put tempdate after last char of tempname
repeat with j = 1 to the number of lines in card field skedbox
if lessonname = item 1 of line j of card field sked_box then
    put tempname & ",c" into line j of card field sked_box
    put empty into card field enter_box
    select after last char of card field enter_box
    exit repeat
end if
if j = the number of lines in card field sked_box then
    answer "Lesson not found" with "return"
    put empty into card field enter_box
    select after last char of card field enter_box
    exit mouseUp
end if
end repeat
end mouseUp

** CARD #4, BUTTON #3: Save Changes
on mouseUp
if card field sked_box is empty then
    answer "No schedule to save" with "Return"
    exit mouseUp
end if
put card field card_id into card_id
put card field sked_box into card field sked of card id card_id
put empty into card field skedname
put empty into card field sked_box
put empty into card field title
put empty into card field enter_box
show card field "msg"
wait 60
hide card field "msg"
send openCard to card "record"
end mouseUp

** CARD #4, BUTTON #4: Cancel Changes
on mouseUp
put card field card_id into card_id
put card field sked of card id card_id into card field sked_box
select after last char of card field enter_box
end mouseUp

** CARD #4, BUTTON #5: Get Schedule
on mouseUp
put card field start into oldstart
put card field listing of card skedfile into card field listing
show card field listing
hide card field title
set lockMessages to true
set lockScreen to true
ask "Enter schedule name"
if it is empty then
    hide card field listing
    show card field title
send mouseUp to card button "return"
exit mouseUp
else
  put quote & it & quote into tempname
  put tempname into card field skedname
  put "card" & & quote & it & quote into tempname
  go to tempname
end if
if the result is not empty then
  repeat until the result is empty
    go to card "record"
    ask "Schedule not found, Enter schedule name"
    if it is empty then
      hide card field listing
      show card field title
      send mouseUp to card button "return"
      exit mouseUp
    end if
    put quote & it & quote into tempname
    put tempname into card field skedname
    put "card" & & quote & it & quote into tempname
    go to tempname
  end repeat
end if
put the short id of this card into card field cardid of card record
put card field sked into card field sked_box of card record
put card field title into card field title of card record
put card field start into card field start of card record
go to card "record"
set lockScreen to false
if oldstart <> card field start then
  send mouseUp to card button draw
end if
hide card field listing
show card field title
set lockScreen to false
select after last char of card field enterbox of card record
end mouseUp

** CARD #4, BUTTON #6: Delete Change ***********************************
on mouseUp
  ask "Enter lesson name"
  if it is empty then
    select after last char of card field enterbox
    exit mouseUp
  end if
  put it into lessonname
  put it & "," after last char of tempname
  ask "Enter date scheduled"
  if it is empty then
exit mouseUp
end if
repeat until validDate(it) is true and -
goodDate(card field start, it) is true
if validDate(it) is false then
  ask it & "Not a valid date, Enter correct date"
end if
if it is empty then
  select after last char of card field enter_box
  exit mouseUp
end if
if goodDate(card field start, it) is false then
  ask "Date must be in sked quarter, enter date"
end if
if it is empty then
  select after last char of card field enter_box
  exit mouseUp
end if
end repeat
put it into tempdate
put tempdate after last char of tempname
repeat with j = 1 to the number of lines in card field sked_box
  if lessonname = item 1 of line j of card field sked_box then
    put tempname & ",s" into line j of card field sked_box
    put empty into card field enter_box
    select after last char of card field enter_box
    exit repeat
  end if
if j = the number of lines in card field sked_box then
  answer "Lesson not found" with "return"
end if
end repeat
end mouseUp

** CARD #4, BUTTON #7: DRAW **********************************************
on mouseUp
if card field start is empty then
  exit mouseUp
end if
show card field working
set cursor to 4
set lockScreen to true
choose pencil tool
click at 50,110
click at 400,100
choose select tool
drag from 0,80 to 160,290
type "x" with commandKey
drag from 360,80 to 513,200
type "x" with commandKey
choose text tool
set textSize to 9

108
set textFont to geneva
set textAlign to left
set textStyle to plain
set textHeight to 9

click at 49,103
type "S"
click at 63,103
type "M"
click at 79,103
type "T"
click at 94,103
type "W"
click at 109,103
type "T"
click at 126,103
type "F"
click at 140,103
type "S"

put card field start into temp
convert temp to long date

click at 48,91
type word 1 of item 2 of temp && item 3 of temp
convert temp to dateItems
put item 2 of temp into mon
put 115 into lineloc
repeat while item 2 of temp = mon
  put item 7 of temp into loc
  click at 31 + 15 * loc,lineloc
  if item 3 of temp > 9 then
    type item 3 of temp
  else
    type " & item 3 of temp
  end if
if loc = 7 then
  put lineloc + 13 into lineloc
end if
convert temp to seconds
put 86400 + temp into temp
convert temp to dateItems
end repeat
convert temp to long date

click at 48,204
type word 1 of item 2 of temp && item 3 of temp
convert temp to dateItems
put item 2 of temp into mon
put 216 into lineloc
repeat while item 2 of temp = mon
  put item 7 of temp into loc
  click at 31 + 15 * loc,lineloc
  if item 3 of temp > 9 then
    type item 3 of temp
  else
    type item 3 of temp
  end if

109
type " " & item 3 of temp
end if
if loc = 7 then
    put lineloc + 13 into lineloc
end if
convert temp to seconds
put 86400 + temp into temp
convert temp to dateItems
end repeat

click at 387,103
type "S"
click at 401,103
type "M"
click at 417,103
type "T"
click at 432,103
type "W"
click at 447,103
type "T"
click at 464,103
type "F"
click at 478,103
type "S"
convert temp to long date

click at 386,91
type word 1 of item 2 of temp & item 3 of temp
convert temp to dateItems
put item 2 of temp into mon
put 115 into lineloc
repeat while item 2 of temp = mon
    put item 7 of temp into loc
    click at 369 + 15 * loc,lineloc
    if item 3 of temp > 9 then
        type item 3 of temp
    else
        type " " & item 3 of temp
    end if
    if loc = 7 then
        put lineloc + 13 into lineloc
    end if
    convert temp to seconds
    put 86400 + temp into temp
    convert temp to dateItems
end repeat

choose browse tool
hide card field working
end mouseUp

** CARD #5: modify ******************************************
on openCard
    put card field start into oldstart
    put card field listing of card skedfile into card field listing
show card field listing
hide card field title
set lockMessages to true
set lockScreen to true
ask "Enter schedule name"
if it is empty then
  send mouseUp to card button "return"
  exit openCard
else
  put quote & it & quote into tempname
  put tempname into card field skedname
  put "card" && quote & it & quote into tempname
  go to tempname
end if
if the result is not empty then
  repeat until the result is empty
    go to card "modify"
    ask "Schedule not found, Enter schedule name"
    put quote & it & quote into tempname
    put tempname into card field skedname
    put "card" && quote & it & quote into tempname
    go to tempname
  end repeat
  put the short id of this card into card field cardid of card modify
  put card field sked into card field sked_box of card modify
  put card field title into card field title of card modify
  put card field start into card field start of card modify
  go to card "modify"
  set lockScreen to false
  if oldstart <> card field start then
    send mouseUp to card button draw
  end if
  hide card field listing
  show card field tide
  select after last char of card field enter_box
end openCard
** CARD #5, FIELD #1: enter_box ******************************************************
on tabKey
  send mouseUp to card button "enter_info"
end tabKey
** CARD #5, BUTTON #1: Return ***********************************************************
on mouseUp
  answer "Unsaved changes will be lost" with "OK" or "Return"
  if it is "return" then
    exit mouseUp
  end if
  put empty into card field sked_box
  put empty into card field title
  put empty into card field enter_box
  go to card training
end mouseUp

111
** CARD #5, BUTTON #2: Delete Lesson *******************************
on mouseUp
  if visible of card field listing is true then
    answer "Get schedule first" with "Cancel" or "Get Sked"
    if it is "cancel" then
      exit mouseUp
    else
      send mouseUp to card button "get sked"
      exit mouseUp
    end if
  end if
ask "Enter lesson name"
if it is empty then
  exit mouseUp
else
  put it into tgt
repeat with j = 1 to the number of lines in card field sked_box
  if tgt = item 1 of line j of card field sked_box then
    delete line j of card field sked_box
    exit mouseUp
  end if
if j = the number of lines in card field sked_box then
  answer "Lesson not found" with "Return"
end if
end repeat
end if
end mouseUp

** CARD #5, BUTTON #3: Change Date *******************************
on mouseUp
  if visible of card field listing is true then
    answer "Get schedule first" with "Cancel" or "Get Sked"
    if it is "cancel" then
      exit mouseUp
    else
      send mouseUp to card button "get sked"
      exit mouseUp
    end if
  end if
ask "Enter lesson name"
if it is empty then
  exit mouseUp
else
  put it into tgt
repeat with j = 1 to the number of lines in card field sked_box
  if tgt = item 1 of line j of card field sked_box then
    put j into linenum
    exit repeat
  end if
if j = the number of lines in card field sked_box then
  answer "Lesson not found" with "Return"
exit mouseUp
end if
end repeat
ask "Enter new sked date"
if it is empty then
exit mouseUp
else
repeat until validDate(it) is true and —
goodDate(card field start, it) is true
if validDate(it) is false then
ask it && "Not a valid date, Enter correct date"
end if
if it is empty then
select after last char of card field enter_box
exit mouseUp
end if
if goodDate(card field start, it) is false then
ask "Date must be in sked quarter, enter date"
end if
if it is empty then
select after last char of card field enter_box
exit mouseUp
end if
end repeat
put it into item 2 of line linenum of card field sked_box
end if
end if
end mouseUp

** CARD #5, BUTTON #4: Add Lesson *******************************************
on mouseUp
if visible of card field listing is true then
answer "Get schedule first" with "Cancel" or "Get Sked"
if it is "cancel" then
exit mouseUp
else
send mouseUp to card button "get sked"
exit mouseUp
end if
end if
put empty into card field enter_box
select after last char of card field enter_box
end mouseUp

** CARD #5, BUTTON #5: Save Changes *******************************************
on mouseUp
if visible of card field listing is true then
answer "Get schedule first" with "Cancel" or "Get Sked"
if it is "cancel" then
exit mouseUp
else
send mouseUp to card button "get sked"
exit mouseUp
end if
end if
put card field card_id into card_id
put card field sked_box into card field sked of card id card_id
put empty into card field skedname
put empty into card field sked_box
put empty into card field title
put empty into card field enter_box
send openCard to card modify
end mouseUp

** CARD #5, BUTTON #6: Cancel Changes *****************************************************
on mouseUp
if visible of card field listing is true then
  answer "Get schedule first" with "Cancel" or "Get Sked"
  if it is "cancel" then
    exit mouseUp
  else
    send mouseUp to card button "get sked"
    exit mouseUp
  end if
end if
put card field card_id into card_id
put card field sked of card id card_id into card field sked_box
select after last char of card field enter_box
end mouseUp

** CARD #5, BUTTON #7: Get Sked ************************************************************
on mouseUp
  put card field start into oldstart
  put card field listing of card skedfile into card field listing
  set lockMessages to true
  show card field listing
  hide card field title
  set lockScreen to true
  ask "Enter schedule name"
  if it is empty then
    send mouseUp to card button "return"
    exit mouseUp
  else
    put quote & it & quote into tempname
    put tempname into card field skedname
    put "card" & & quote & it & quote into tempname
go to tempname
  end if
if the result is not empty then
  repeat until the result is empty
    go to card "modify"
    ask "Schedule not found, Enter schedule name"
    put quote & it & quote into tempname
    put tempname into card field skedname
put "card" & & quote & it & quote into tempname
go to tempname
end repeat
end if
put the short id of this card into card field card_id of card modify
put card field sked into card field sked_box of card modify
put card field title into card field title of card modify
put card field start into card field start of card modify
go to card "modify"
set lockScreen to false
if oldstart <> card field start then
  send mouseUp to card button draw
end if
hide card field listing
show card field title
select after last char of card field enter_box
end mouseUp

** CARD #5, BUTTON #8: DRAW ***********************************************
on mouseUp
if card field start is empty then
  exit mouseUp
end if
set cursor to 4
show card field working
set lockScreen to true
choose pencil tool
click at 50,110
click at 400,100
choose select tool
drag from 0,80 to 160,290
type "x" with commandKey
drag from 360,80 to 513,200
type "x" with commandKey
choose text tool
set textSize to 9
set textFont to geneva
set textAlign to left
set textStyle to plain
set textHeight to 9
click at 49,103
   type "S"
click at 63,103
   type "M"
click at 79,103
   type "T"
click at 94,103
   type "W"
click at 109,103
   type "T"
click at 126,103
   type "F"
click at 140,103
  type "S"
  put card field start into temp
  convert temp to long date
  click at 48,91
  type word 1 of item 2 of temp & item 3 of temp
  convert temp to dateItems
  put item 2 of temp into mon
  put 115 into lineloc
  repeat while item 2 of temp = mon
    put item 7 of temp into loc
    click at 31 + 15 * loc,lineloc
    if item 3 of temp > 9 then
      type item 3 of temp
    else
      type "" & item 3 of temp
    end if
    if loc = 7 then
      put lineloc + 13 into lineloc
    end if
    convert temp to seconds
    put 86400 + temp into temp
    convert temp to dateItems
  end repeat
  convert temp to long date
  click at 48,204
  type word 1 of item 2 of temp & item 3 of temp
  convert temp to dateItems
  put item 2 of temp into mon
  put 216 into lineloc
  repeat while item 2 of temp = mon
    put item 7 of temp into loc
    click at 31 + 15 * loc,lineloc
    if item 3 of temp > 9 then
      type item 3 of temp
    else
      type "" & item 3 of temp
    end if
    if loc = 7 then
      put lineloc + 13 into lineloc
    end if
    convert temp to seconds
    put 86400 + temp into temp
    convert temp to dateItems
  end repeat
  click at 387,103
  type "S"
  click at 401,103
  type "M"
  click at 417,103
  type "T"
  click at 432,103

116
type "W"
click at 447,103
type "T"
click at 464,103
type "F"
click at 478,103
type "S"
convert temp to long date
click at 386,91
type word 1 of item 2 of temp & item 3 of temp
click to dateItems
put item 2 of temp into mon
put 115 into lineloc
repeat while item 2 of temp = mon
put item 7 of temp into loc
click at 369 + 15 * loc,lineloc
if item 3 of temp > 9 then
type item 3 of temp
else
type " " & item 3 of temp
end if
if loc = 7 then
put lineloc + 13 into lineloc
end if
convert temp to seconds
put 86400 + temp into temp
click to dateItems
end repeat
choose browse tool
hide card field working
end mouseUp

** CARD #5, BUTTON #9: enter_info ************************************
on mouseUp
if card field title is empty then
   answer "Schedule must have a title" with "Cancel" or "Title"
   if it is "cancel" then
      exit mouseUp
   else
      send mouseUp to card button "change title"
   end if
end if
if card field start is empty then
   answer "No schedule quarter entered" with "Cancel" or "Enter"
   if it is "cancel" then
      exit mouseUp
   else
      send mouseUp to card button "change qtr"
      send mouseUp to card button "draw"
   end if
end if
if line 1 of card field enter_box is empty then

ask "No lesson name, Enter lesson name"
if it is empty then
  put empty into card field enter_box
  select after last char of card field enter_box
  exit mouseUp
end if
put it into line 1 of card field enter_box
end if
put line 1 of card field enter_box & "," after last char of tempname
put line 2 of card field enter_box into it
repeat until validDate(it) is true and goodDate(card field s, sot, it) is true
  ask it & & "Not a valid date, Enter correct date"
end if
if it is empty then
  select after last char of card field enter_box
  exit mouseUp
end if
if goodDate(card field start, it) is false then
  ask "Date must be in sked quarter, enter date"
end if
if it is empty then
  select after last char of card field enter_box
  exit mouseUp
end if
end repeat
if the number of lines in card field sked_box >= 15 then
  answer "Schedule is full (15 Items max)" with "Cancel" or "Save sked"
  if it is "cancel" then
    exit mouseUp
  else
    send mouseUp to card button "save sked"
  end if
end if
put it after last char of tempname
put tempname & "," & return after last char of card field sked_box
put empty into card field enter_box
select after last char of card field enter_box
end mouseUp

** CARD #6: draw *********************************************
on openCard
  put card field listing of card skedfile into card field listing
  show card field header
  show card field list_head
  show card field listing
end openCard
on idle
** CARD #6, BUTTON #1: Return  ******************************************
on mouseUp
  send mouseUp to card button erase
  put empty into card field sked_box
  put empty into card field start
  put empty into card field title
  put empty into card field listing
  go to card training
end mouseUp

** CARD #6, BUTTON #2: Print  ******************************************
on mouseUp
  set cursor to 4
  show card field shield
  open printing
  print this card
  close printing
end mouseUp

** CARD #6, BUTTON #3: Erase  ******************************************
on mouseUp
  set cursor to 4
  set lockScreen to true
  choose pencil tool
  drag from 0,0 to 0,1
  choose select tool
  drag from 0,0 to 512,342
  type "x" with commandKey
  show card field header
  choose browse tool
end mouseUp

** CARD #6, BUTTON #4: Draw  ******************************************
on mouseUp
  set cursor to 4
  set lockScreen to true
  if card field start is empty then
      answer "Get a schedule to draw" with "Cancel" or "Get Sked"
      if it = "cancel" then
          exit mouseUp
      else
          send mouseUp to card button "get sked"
      end if
  end if
  if card field start is empty then
      exit mouseUp
  end if
  hide card field header
  hide card field listing
  hide card field list_head
  hide card field shield
end idle
choose bucket tool
set pattern to 1
click at 100,100
choose text tool
set the textFont to courier
set the textSize to 10
set the textStyle to plain
set the textHeight to 9
set textAlign to left
put card field start into startsecs
convert startsecs to seconds.
repeat with j = 1 to the number of lines in card field sked_box
  click at 30,65 + j * 15
  type char 1 to 11 of item 1 of line j of card field sked_box
  put item 2 of line j of card field sked_box into secs
  convert secs to seconds
  put (secs - startsecs)/86400 into temp
  if secs = startsecs then
    click at 99,65 + j * 15
  else
    convert secs to dateitems
    if item 7 of secs = 2 then
      click at 4 * temp + 99,65 + j * 15
    else
      click at 4 * temp + 98,65 + j * 15
    end if
  end if
end repeat
set the textFont to geneva
set the textSize to 9
set the textStyle to plain
set the textHeight to 9
choose line tool
drag from 98,61 to 466,61
repeat with j = 1 to 16
  put 53 + 15 * j into horiz
  drag from 466,horiz to 28,horiz
end repeat
drag from 27,68 to 27,293
drag from 98,68 to 98,293
put 94 into loc
repeat with j = 1 to 93
  drag from loc + 4 * j,62 to loc + 4 * j,67
end repeat
drag from 466,68 to 466,293
choose browse tool
put card field start into begin
convert begin to dateitems
get last char of begin
put it into temp
if temp > 2 then
put temp - 3 into temp
else
if temp = 2 then
put 6 into temp
else
put 5 into temp
end if
end if
put 122 - 4 * temp into location
put begin into tempdate
put last char of begin into temp
put (9 - temp) mod 7 into temp
convert tempdate to seconds
put tempdate + 86400 * temp into tempdate
set the textFont to geneva
set the textSize to 9
set the textStyle to plain
set the textHeight to 9
set textAlign to left
repeat until location > 465
choose line tool
drag from location,53 to location,292
choose text tool
convert tempdate to abbr date
put the length of word 3 of tempdate into len
put char 1 to len - 1 of word 3 of tempdate into day
click at location + 2,58
type day
if day < 8 then
put char 2 of item 2 of tempdate into mon
put numToChar(charToNum(char 3 of item 2 of tempdate) - 32) after-
last char of mon
put numToChar(charToNum(char 4 of item 2 of tempdate) - 32) after-
last char of mon
type mon
end if
convert tempdate to seconds
put 604800 + tempdate into tempdate
put location + 28 into location
end repeat
click at 28,66
type "LESSON NAME"
click at 256,25
set textAlign to center
set textHeight to 12
set textSize to 12
type "TRAINING SCHEDULE:" & & card field title
click at 256,40
put card field start into temp
convert temp to long date
type word 1 of item 2 of temp & & ": "

121
convert temp to seconds
put temp + 6048000 into temp
convert temp to long date
type word 1 of item 2 of temp && item 3 of temp
choose browse tool
end mouseUp

** CARD #6, BUTTON #5: Get Sked

on mouseUp
    set cursor to 4
    show card field list_head
    show card field listing
    set lockScreen to true
    set lockMessages to true
    put card field listing of card skedfile into card field listing
    if card field listing is empty then
        answer "No schedules on file" with "Return"
        exit mouseUp
    end if
    ask "Enter schedule name"
    if it is empty then
        exit mouseUp
    else
        put quote & it & quote into tempname
        put "card" && quote & it & quote into tempname
        go to tempname
    end if
    if the result is not empty then
        repeat until the result is empty
            go to card "draw"
            ask "Schedule not found, Enter schedule name"
            if it is empty then
                exit mouseUp
            end if
            put quote & it & quote into tempname
            put "card" && quote & it & quote into tempname
            go to tempname
        end repeat
    end if
    put card field sked into card field sked_box of card draw
    put card field title into card field title of card draw
    put card field start into card field start of card draw
    go to card "draw"
    hide card field list_head
    hide card field listing
end mouseUp

** CARD #11, BUTTON #1: RETURN

on mouseUp
    go to card training
end mouseUp
** CARD #11, BUTTON #2: draft *****************************************************

on mouseUp
    set cursor to 4
    set lockScreen to true
    set lockMessages to true
    put empty into card field tempmsg
    global oldnum, olddtg
    put card field serno into oldnum
    put card field lastdtg into olddtg
    put card field serno + 1 into tempnum
    if tempnum = 1000 then
        put "001" into tempnum
    end if
    if tempnum < 10 then
        put "00" & tempnum into tempnum
    end if
    if tempnum < 100 and tempnum > 9 then
        put "0" & tempnum into tempnum
    end if
    put empty into tempmsgline
    go to card draw
    repeat with k = 1 to 4
        go to next card
        put field 2 into field 3
        repeat with j = 1 to the number of lines in field 1
            if item 9 of line j of field 1 is "PDG" then
                put tempnum into item 9 of line j of field 1
                put item 1 of line j of field 1 & "/" & msgDate(item 5 of line j of field 1) after last char of 
                tempmsgline
                if item 8 of line j of field 1 = "OBS " then
                    put "/0/" after last char of tempmsgline
                else
                    if item 8 of line j of field 1 = "S-OBS" then
                        put "/2/" after last char of tempmsgline
                    else
                        put "/4/" after last char of tempmsgline
                    end if
                end if
            end if
            put item 7 of line j of field 1 & "/A/" & 
            item 2 of line j of field 1 & return after last char of 
            tempmsgline
            find whole item 1 of line j of field 1 in field 2
            put word 2 of the foundLine into newlinenum
            put item 1 of line j of field 1 & space into templine
            put item 2 of line j of field 1 into temp
            repeat until the length of temp = 13
                put space after last char of temp
            end repeat
            put temp after last char of templine
            put item 3 of line j of field 1 into temp
            repeat until the length of temp = 20
        end repeat
    end repeat
put space after last char of temp
end repeat
put temp after last char of templine
put "M-" & item 4 of line j of field 1 & space & after last char of templine
if item 5 of line j of field 1 is empty then
    put space & space & space & space & space & space after last char of templine
else
    put mildate(item 5 of line j of field 1) & space
& space after last char of templine
end if
if item 6 of line j of field 1 is empty then
    put space & space & space & space & space & space after last char of templine
else
    put mildate(item 6 of line j of field 1) & space
& space after last char of templine
end if
put item 7 of line j of field 1 & space &-
item 8 of line j of field 1 & space &-
item 9 of line j of field 1 after last char of templine
put templine into line newlinenum of field 2
end if
end repeat
end repeat
go to card trarep
if tempmmsgline is empty then
    answer "No accomplishments to report" with "return"
    put oldnum into card field serno
    put olddtg into card field lastdtg
    exit mouseUp
end if
put card field header into card field tempmsg
put tempnum after last char of card field tempmsg
put the date into today
convert today to dateitems
if item 3 of today < 10 then
    put "0" & item 3 of today into dtg
else
    put item 3 of today into dtg
end if
put the long time into now
if the length of now = 7 then
    put "0" & char 1 of now & char 3 to 4 of now after last char of dtg
else
    put char 1 to 2 of now & char 4 to 5 of now after last char of dtg
end if
put "Z " after last char of dtg
convert today to abbr date
put char 2 of item 2 of today after last char of dtg
put numToChar(charToNum(char 3 of item 2 of today) - 32) after −
last char of dig
put numToChar(charToNum(char 4 of item 2 of today) - 32) after last char of dig
put space & char 4 to 5 of item 3 of today after last char of dig
put " AS OF " & dig & return after last char of card field tempmsg
put card field line 1 & return after last char of card field tempmsg
put tempmsgsline after last char of card field tempmsg
answer "Any Air Controller data to report ?" with "Yes" or "No"
if it is "yes" then
  repeat until it is "no"
    answer "Choose qualification type" with "AICS" or "AIC" or "ASAC"
    if it is "aics" then
      put "AICS" into qualline
    else
      if it is "aic" then
        put "AIC" into qualline
      else
        put "ASAC" into qualline
      end if
    end if
  end if
ask "Enter Name (Last, FMI.)"
if it is empty then
  exit repeat
end if
put space & it after last char of qualline
ask "Enter rank/rate"
if it is empty then
  exit repeat
end if
put "/" & it after last char of qualline
ask "Enter PRD"
if it is empty then
  exit repeat
end if
put "/" & it after last char of qualline
ask "Enter total hours/intercepts since desig"
if it is empty then
  exit repeat
end if
put "/" & it after last char of qualline
put qualline & return after last char of card field tempmsg
answer "Any more Air Controller data to report ?" with "No" or "Yes"
end repeat
end if
answer "Any Gram Analysis data to report ?" with "Yes" or "No"
if it is "yes" then
  repeat until it is "no"
    ask "Enter Name (Last, FMI.)"
    if it is empty then
      exit repeat
    end if
  end if
125
put it into qualline
ask "Enter rank/rate"
if it is empty then
exit repeat
end if
put "/" & it after last char of qualline
ask "Enter PRD"
if it is empty then
exit repeat
end if
put "/" & it after last char of qualline
ask "Enter total hours in month"
if it is empty then
exit repeat
end if
put "/" & it after last char of qualline
put qualline & return after last char of card field tempmsg
answer "Any more Gram Analysis data to report?" with "No" or "Yes"
end repeat
end if
put "MOB" into tempmrating
put char 5 to 9 of card field mob_mratin of card mob_n into pct
put 100 * pct into pct
set numberFormat to "00"
put " " & pct & ";" & "(" & "M" & char 3 -
of card field mob_mratin of card mob_n & ")" & return after 
last char of tempmrating
put tempmrating after last char of card field tempmsg
put the date into today
convert today to short date
put the number of chars in today into length
put char length - 1 to length of today into year
put year + 6 into year
put year into char length - 1 to length of today
put "DECL " & mildate(today) & return & "BT" & return after 
last char of card field tempmsg
put tempnum into card field serno
put dtg into card field lastdtg
end mouseUp

** CARD #11, BUTTON #3: Cancel ************************************************
on mouseUp
set cursor to 4
set lockScreen to true
set lockMessages to true
global oldnum,olddtg
put card field serno into tempserno
put oldnum into card field serno
put olddtg into card field lastdtg
put empty into card field tempmsg
go to card draw

126
repeat with \( k = 1 \) to 4
  go to next card
  put field 3 into field 2
repeat with \( j = 1 \) to the number of lines in field 1
  if item 9 of line \( j \) of field 1 = tempserno then
    put "PDG" into item 9 of line \( j \) of field 1
  end if
end repeat
end repeat
  go to card trarep
end mouseUp

** CARD #11, BUTTON #4: Print ****************************
on mouseUp
  set textFont to courier
  set textSize to 12
  set textStyle to plain
  PrintField(card field tempmsg)
  reset paint
end mouseUp

** CARD #12: view_data ****************************
on openCard
  put empty into field viewer
  put empty into card field marea
  put empty into card field mrating
  put empty into card field as_of
  put empty into card field update
end openCard

** CARD #12, FIELD #2: marea ****************************
on upperCase
  if card field marea is not empty then
    repeat with \( j = 1 \) to the length of card field marea
      get char \( j \) of card field marea
      if charToNum(it) > 96 and charToNum(it) < 123 then
        put numToChar(charToNum(it) - 32) into char \( j \) of card field marea
      end if
    end repeat
  end if
end upperCase

** CARD #12, BUTTON #1: Get Data ****************************
on mouseUp
  hide field viewer
  hide card field header
  hide card field title
  hide card field trnum
  show card field missionareas
  ask "Enter mission area"
  if it is empty then
    hide card field missionareas
    exit mouseUp
  else
put it into area
if it is not in card field missionareas or the length of it < 3 then
  repeat
    answer "Invalid mission area, Try again"
    ask "Enter mission area"
    if it is empty then
      hide card field missionareas
      exit mouseUp
    else
      put it into area
      if it is not in card field missionareas or the length of it < 3 or it is in "mission areas" then
        next repeat
      end if
      exit repeat
    end if
  end repeat
end if
end if
set cursor to 4
set lockScreen to true
if area is "mob-e" then
  put "mob_e" into card field marea
end if
if area is "mob-d" then
  put "mob_d" into card field marea
end if
if area is "mob-s" then
  put "mob_s" into card field marea
end if
if area is "mob-n" then
  put "mob_n" into card field marea
end if
if area is "mob" then
  put "mob" into card field marea
end if
if area <> "mob" then
  put card field marea into tempname
  put field tempname of card tempname into field viewer
  put card field mrating of card tempname into card field mrating
  put card field as_of of card tempname into card field as_of
  put card field update of card tempname into card field update
else
  if area = "mob" then
    put area into card field marea
    put field mob_e of card mob_e into field viewer
    put field mob_d of card mob_d after last char of field viewer
    put field mob_s of card mob_s after last char of field viewer
    put field mob_n of card mob_n after last char of field viewer
    put card field mob_mrating of card mob_n into card field mrating
  end if
end if
put card field mob_as_of of card mob_n into card field as_of
put card field update of card mob_n into card field update
else
  put card field marea into tempname
  put field tempname of card tempname into field viewer
  put card field mrating of card tempname into card field mrating
  put card field as_of of card tempname into card field as_of
  put card field update of card tempname into card field update
end if
end if
send upperCase to card field marea
hide card field missionareas
show field viewer
show card field header
click at 504,103
set lockscreen to false
end mouseUp

** CARD #12, BUTTON #2: RETURN ****************************
on mouseUp
  go to card training
end mouseUp

** CARD #12, BUTTON #3: M-Rating ****************************
on mouseUp
  if card field marea is empty then
    answer "No data present" with "return"
    exit mouseUp
  end if
  set numberFormat to "0.###"
  set cursor to 4
  set lockScreen to true
  set lockMessages to true
  put 0 into m1
  put 0 into m2
  put 0 into m3
  put 0 into m4
  put card field marea into pma
  if pma <> "mob" then
    go to card pma
  repeat with j = 1 to the number of lines in field 1
    if item 4 of line j of field 1 = 1 then
      put m1 + 1 into m1
    end if
    if item 4 of line j of field 1 = 2 then
      put m2 + 1 into m2
    end if
    if item 4 of line j of field 1 = 3 then
      put m3 + 1 into m3
    end if
    if item 4 of line j of field 1 = 4 then
      put m4 + 1 into m4
  end repeat
**
end if
end repeat
else
go to card draw
repeat with cardnum = 1 to 4
go to next card
repeat with j = 1 to the number of lines in field 1
if item 4 of line j of field 1 = 1 then
  put m1 + 1 into m1
end if
if item 4 of line j of field 1 = 2 then
  put m2 + 1 into m2
end if
if item 4 of line j of field 1 = 3 then
  put m3 + 1 into m3
end if
if item 4 of line j of field 1 = 4 then
  put m4 + 1 into m4
end if
end repeat
end repeat
end if
go to card view_data
put (4 * m1 + 3 * m2 + 2 * m3)/((4 * (m1 + m2 + m3 + m4))) into readfactor
if readfactor > 0.849 then
  put "M-1" && readfactor into card field mrating
else
  if readfactor > 0.699 then
    put "M-2" && readfactor into card field mrating
  else
    if readfactor > 0.549 then
      put "M-3" && readfactor into card field mrating
    else
      put "M-4" && readfactor into card field mrating
    end if
  end if
end if
get the short date
put mildate(it) into card field as_of
if card field marea <> "mob" then
  put card field mrating into card field mrating of card pma
  put card field as_of into card field as_of of card pma
else
  put card field mrating into card field mob_mrating of card mob_n
  put card field as_of into card field mob_as_of of card mob_n
end if
end mouseUp

** CARD #12, BUTTON #4: enter data ****************************
on mouseUp
  set lockScreen to true
set lockMessages to true
if card field marea is empty then
    answer "No data present" with "Cancel" or "Get Data"
    if it is "cancel" then
        exit mouseUp
    else
        set lockScreen to false
        send mouseUp to card button "get data"
        if card field marea is empty then
            exit mouseUp
        end if
        set lockScreen to true
    end if
else
    put card field marea into cardname
    ask "Enter exercise id"
    if it is empty then
        exit mouseUp
    end if
end if
put card field marea into cardname
ask "Enter exercise id"
if it is empty then
    exit mouseUp
end if
put it into idnum
if cardname is "mob" then
    if char 3 of idnum = "0" then
        put "mob_e" into cardname
    else
        if char 3 of idnum = "2" then
            put "mob_d" into cardname
        else
            if char 3 of idnum = "3" then
                put "mob_s" into cardname
            else
                put "mob_n" into cardname
            end if
        end if
    end if
end if
go to card cardname
find whole idnum in field 1
repeat until the result is empty
    find whole idnum in field 1
    if the result <> empty or the length of idnum <> 6 then
        answer "Invalid number, try again" with "return"
        ask "Enter exercise id"
        if it is empty then
            go to card view_data
            exit mouseUp
        else
            put it into idnum
        end if
    end if
end repeat
put word 2 of the foundLine into linenum
ask "Enter completion date (MM/DD/YY)"
if it is empty then
    go to card view_data
    exit mouseUp
end if
put it into compdate
if validDate(compdate) is false then
    repeat until validDate(compdate) is true
        ask "Invalid date, Enter completion date (MM/DD/YY)"
        if it is empty then
            go to card view_data
            exit mouseUp
        end if
        put it into compdate
    end repeat
end if
ask "Enter score (Must be 4 digits or 'NONE')" with "NONE"
if it is empty then
    go to card view_data
    exit mouseUp
end if
put it into score
repeat with charnum = 1 to the length of score
    if char charnum of score is not in "NOE0123456789" then
        put true into invalid
        exit repeat
    else
        put false into invalid
    end if
end repeat
if the length of score <> 4 or invalid is true then
    repeat until the length of score = 4 and invalid is false
        ask "Score must be 4 digits (No decimal point)"
        if it is empty then
            go to card view_data
            exit mouseUp
        end if
        put it into score
    end repeat
end repeat
answer "What evaluation method ?" with "Equivalent" or "Observed" or "Self Observed"
if it is "equivalent" then
    put "EQUIV" into eval
else
if it is "Observed" then
  put "OBS " into eval
else
  put "S-OBS" into eval
end if
end if
put compdate into item 5 of line linenum of field 1
if char 4 of idnum = 5 then
  put expDate(compdate,item 10 of line linenum of field 1) into item 6 of line linenum of field 1
else
  put expDate(compdate,21) into item 6 of line linenum of field 1
end if
put "1" into item 4 of line linenum of field 1
put score into item 7 of line linenum of field 1
put eval into item 8 of line linenum of field 1
put "PDG" into item 9 of line linenum of field 1
find idnum in field 2
put word 2 of the foundLine into newlinenum
put item 1 of line linenum of field 1 & space into templine
put item 2 of line linenum of field 1 into temp
repeat until the length of temp = 13
  put space after last char of temp
end repeat
put temp after last char of templine
put item 3 of line linenum of field 1 into temp
repeat until the length of temp = 20
  put space after last char of temp
end repeat
put temp after last char of templine
put "M-" & item 4 of line linenum of field 1 & space & space after last char of templine
if item 5 of line linenum of field 1 is empty then
  put space & space & space & space & space & space & space & space & space after last char of templine
else
  put mildate(item 5 of line linenum of field 1) & space & space after last char of templine
end if
if item 6 of line linenum of field 1 is empty then
  put space & space & space & space & space & space & space & space & space after last char of templine
else
  put mildate(item 6 of line linenum of field 1) & space & space after last char of templine
end if
put item 7 of line linenum of field 1 & space & space & space & space & space & space after last char of templine
item 8 of line linenum of field 1 after last char of templine
put templine into line newlinenum of field 2
if card field marea of card viewdata <> "mob" then
  put field 2 into field viewer of card view_data
end if
go to card view_data
else
go to card view_data
find whole idnum in field viewer
put word 2 of the foundLine into here
put templine into line here of field viewer
click at 100,100
end if
end mouseUp

** CARD #12, BUTTON #5: Update ****************************
on mouseUp
if card field marea is empty then
  answer "No data present" with "Return"
  exit mouseUp
end if
set cursor to 4
set lockScreen to true
set lockMessages to true
put the seconds into today
if card field marea <> "mob" then
goto card cardname
repeat with j = 1 to the number of lines in field 1
  if item 6 of line j of field 1 is empty then
    next repeat
  else
    put item 6 of line j of field 1 into baddate
    convert baddate to seconds
    if baddate > today then
      next repeat
    else
      if char 4 of item 1 of line j of field 1 <> "5" then
        put empty into item 6 of line j of field 1
        put "4" into item 4 of line j of field 1
        find item 1 of line j of field 1 in field 2
        put word 2 of the foundLine into linenum
        put item 1 of line j of field 1 & space into templine
        put item 2 of line j of field 1 into temp
        repeat until the length of temp = 13
          put space after last char of temp
        end repeat
        put temp after last char of templine
        put item 3 of line j of field 1 into temp
        repeat until the length of temp = 20
          put space after last char of temp
        end repeat
        put temp after last char of templine
        put "M-" & item 4 of line j of field 1 & space & space-
          after last char of templine
        if item 5 of line j of field 1 is empty then
          put space & space & space & space & space & space & space &
space & space & space & space after last char of templine
else
    put mildate(item 5 of line j of field 1) & space & space after last char of templine
end if
if item 6 of line j of field 1 is empty then
    put space & space & space & space & space & space & space after last char of templine
else
    put mildate(item 6 of line j of field 1) & space & space after last char of templine
end if
put item 7 of line j of field 1 & space & search for line in field 2
item 8 of line j of field 1 & space & search for line in field 2
if item 9 of line j of field 1 is empty then
    put templine into line linenum of field 2
else
    put expDate(item 5 of line j of field 1, item 12 of line j of field 1) into temp3
    convert temp3 to seconds
    if temp3 < today then
        put empty into item 6 of line j of field 1
        put "4" into item 4 of line j of field 1
    else
        put expDate(item 5 of line j of field 1, item 11 of line j of field 1) into temp2
        convert temp2 to seconds
        if temp2 < today then
            convert temp3 to short date
            put temp3 into item 6 of line j of field 1
            put "3" into item 4 of line j of field 1
        else
            put expDate(item 5 of line j of field 1, item 10 of line j of field 1) into temp1
            convert temp1 to seconds
            if temp1 < today then
                convert temp2 to short date
                put temp2 into item 6 of line j of field 1
                put "2" into item 4 of line j of field 1
            end if
        end if
    end if
end if
end if
find item 1 of line j of field 1 in field 2
put word 2 of the found line into linenum
put item 1 of line j of field 1 & space into templine
put item 2 of line j of field 1 into temp
repeat until the length of temp = 13
put space after last char of temp
end repeat
put temp after last char of templine
put item 3 of line j of field 1 into temp

135
repeat until the length of temp = 20
  put space after last char of temp
end repeat
put temp after last char of templine
put "M-" & item 4 of line j of field 1 & space & after last char of templine
if item 5 of line j of field 1 is empty then
  put space & space & space & space & space & after last char of templine
else
  put mildate(item 5 of line j of field 1) & space & after last char of templine
end if
if item 6 of line j of field 1 is empty then
  put space & space & space & space & space & after last char of templine
else
  put mildate(item 6 of line j of field 1) & space & after last char of templine
end if
put item 7 of line j of field 1 & space & after last char of templine
put item 8 of line j of field 1 & space & after last char of templine
put item 9 of line j of field 1 after last char of templine
put templine into line linenum of field 2
end if
end if
end repeat
put empty into field viewer of card view_data
put field 2 of this card into field viewer of card view_data
go to card view_data
convert today to short date
put mildate(today) into card field update
put card field update into card field update of card cardname
else
  go to card draw
repeat with cardnum = 1 to 4
  go to next card
repeat with j = 1 to the number of lines in field 1
  if item 6 of line j of field 1 is empty then
    next repeat
  else
    put item 6 of line j of field 1 into baddate
    convert baddate to seconds
    if baddate > today then
      next repeat
    else
      if char 4 of item 1 of line j of field 1 <> "5" then
        put empty into item 6 of line j of field 1
        put "4" into item 4 of line j of field 1
        find item 1 of line j of field 1 in field 2
        put word 2 of the foundLine into linenum
        put item 1 of line j of field 1 & space into templine
      end if
    end if
  end if
end if
put item 2 of line j of field 1 into temp
repeat until the length of temp = 13
  put space after last char of temp
end repeat
put temp after last char of templine
put item 3 of line j of field 1 into temp
repeat until the length of temp = 20
  put space after last char of temp
end repeat
put temp after last char of templine
put "M-" & item 4 of line j of field 1 & space & space
after last char of templine
if item 5 of line j of field 1 is empty then
  put space & space & space & space & space & space & space & space & space after last char of templine
else
  put mildate(item 5 of line j of field 1) & space & space after last char of templine
end if
if item 6 of line j of field 1 is empty then
  put space & space & space & space & space & space & space & space after last char of templine
else
  put mildate(item 6 of line j of field 1) & space & space after last char of templine
end if
put item 7 of line j of field 1 & space & space
item 8 of line j of field 1 & space & space
item 9 of line j of field 1 after last char of templine
put templine & return into linenumber
else
  put expDate(item 5 of line j of field 1, item 12 of line j of field 1) into temp3
  convert temp3 to seconds
  if temp3 < today then
    put empty into item 6 of line j of field 1
    put "4" into item 4 of line j of field 1
  else
    put expDate(item 5 of line j of field 1, item 11 of line j of field 1) into temp2
    convert temp2 to seconds
    if temp2 < today then
      convert temp3 to short date
      put temp3 into item 6 of line j of field 1
      put "3" into item 4 of line j of field 1
    else
      put expDate(item 5 of line j of field 1, item 10 of line j of field 1) into temp1
      convert temp1 to seconds
      if temp1 < today then
        convert temp2 to short date
        put temp2 into item 6 of line j of field 1
put "2" into item 4 of line j of field 1
      end if
    end if
end if
find item 1 of line j of field 1 in field 2
put word 2 of the foundLine into linenum
put item 1 of line j of field 1 & space into templine
put item 2 of line j of field 1 into temp
repeat until the length of temp = 15
  put space after last char of temp
end repeat
put temp after last char of templine
put item 3 of line j of field 1 into temp
repeat until the length of temp = 20
  put space after last char of temp
end repeat
put temp after last char of templine
put "M-" & item 4 of line j of field 1 & space & space after last char of templine
if item 5 of line j of field 1 is empty then
  put space & space & space & space & space & space & space & space & space & space after last char of templine
else
  put mildate(item 5 of line j of field 1) & space after last char of templine
end if
if item 6 of line j of field 1 is empty then
  put space & space & space & space & space & space & space & space after last char of templine
else
  put mildate(item 6 of line j of field 1) & space & space after last char of templine
end if
put item 7 of line j of field 1 & space & space & space & space & space after last char of templine
item 8 of line j of field 1 & space & space after last char of templine
item 9 of line j of field 1 after last char of templine
put templine into line linenum of field 2
end if
end if
end repeat
end repeat
go to card view_data
put empty into field viewer
go to card draw
repeat with cardnum = 1 to 4
  go to next card
  put field 2 after last char of field viewer of card view_data
end repeat
go to card view_data
convert today to short date
put milDate(today) into card field update
put card field update into card field mob_update of card mob_n
end if
end mouseUp
APPENDIX E. SCHEDULES STACK SCRIPTS

SCRIPTS FOR STACK: schedules

** STACK SCRIPT ***********************************************

function validDate date
put date into tempdate
if the length of tempdate < 6 or the length of tempdate > 8 then
  return false
else
  if the length of tempdate = 6 then
    if char 1 of tempdate is not in "123456789" then
      return false
    end if
    if char 2 of tempdate <> "/" then
      return false
    end if
    if char 3 of tempdate is not in "123456789" then
      return false
    end if
    if char 4 of tempdate <> "/" then
      return false
    end if
    if char 5 of tempdate is not in "1234567890" then
      return false
    end if
    if char 6 of tempdate is not in "1234567890" then
      return false
    end if
  end if
  if the length of tempdate = 7 then
    if char 3 of tempdate = "/" then
      if char 1 of tempdate is not in "12" then
        return false
      end if
      if char 2 of tempdate is not in "012" then
        return false
      end if
      if char 4 of tempdate is not in "123456789" then
        return false
      end if
      if char 5 of tempdate <> "/" then
        return false
      end if
      if char 6 of tempdate is not in "1234567890" then
        return false
      end if
    end if
  end if
end if
end if
if char 7 of tempdate is not in "1234567890" then
  return false
end if
else
  if char 2 of tempdate = "/" then
    if char 1 of tempdate is not in "123456789" then
      return false
    end if
    if char 3 of tempdate is not in "123" then
      return false
    end if
    if char 4 of tempdate is not in "1234567890" then
      return false
    end if
    if char 5 of tempdate <> "/" then
      return false
    end if
    if char 6 of tempdate is not in "1234567890" then
      return false
    end if
    if char 7 of tempdate is not in "1234567890" then
      return false
    end if
    if char 1 of tempdate = 2 then
      if char 3 of tempdate = 3 then
        return false
      end if
    else
      if char 3 of tempdate = 2 then
        if char 4 of tempdate = 9 then
          put char 6 to 7 of tempdate into year
          if year mod 4 <> 0 then
            return false
          end if
        end if
      end if
    end if
  end if
end if
if char 3 of tempdate = 3 then
  if char 4 of tempdate <> 0 then
    return false
  end if
end if
end if
if char 3 of tempdate = 3 then
  if char 4 of tempdate > 1 then
    return false
  end if
end if
end if
end if
if the length of tempdate = 8 then
  if char 3 of tempdate <> "/" then
    return false
  end if
  if char 6 of tempdate <>="/" then
    return false
  end if
  if char 1 of tempdate <> 1 then
    return false
  end if
  if char 2 of tempdate is not in "012" then
    return false
  end if
  if char 4 of tempdate is not in "123" then
    return false
  end if
  if char 5 of tempdate is not in "1234567890" then
    return false
  end if
  if char 7 of tempdate is not in "1234567890" then
    return false
  end if
  if char 8 of tempdate is not in "1234567890" then
    return false
  end if
  if char 2 of tempdate = 1 then
    if char 4 of tempdate = 3 then
      if char 5 of tempdate <> 0 then
        return false
      end if
      end if
    end if
  end if
end if
end if
return true
end validDate
function validQtr qtr
  if the length of qtr <> 4 then
    return false
  end if
  if char 1 of qtr is not in "1234" then
    return false
  end if
  if char 2 of qtr is not in "/-" then
    return false
  end if
if char 3 of qtr is not in "1234567890" then
    return false
end if
if char 4 of qtr is not in "1234567890" then
    return false
end if
return true
end validQr

function goodDate start, date
    convert date to seconds
    convert start to seconds
    if date < start then
        return false
    end if
    convert date to dateItems
    convert start to dateItems
    if item 2 of date > item 2 of start + 2 then
        return false
    end if
    if item 1 of date > item 1 of start then
        return false
    end if
    return true
end goodDate

function convertQtr qtr
    put char 1 of qtr * 3 - 2 into tempdate
    put "/" & 1 after last char of tempdate
    put "/" & char 3 to 4 of qtr after last char of tempdate
    return tempdate
end convertQtr

function lastDay date
    if the length of date = 7 or char 1 of date = 7 then
        return 93
    end if
    if char 1 of date = 4 then
        return 92
    end if
    put char 5 to 6 of date into yr
    if yr mod 4 = 0 and yr <> "00" then
        return 92
    else
        return 91
    end if
end lastDay

function fiscal date
    if the length of date = 7 then
        put char 6 to 7 of date into yr
        if yr < 80 then
            return 2000 + yr + 1
        else
            return 1900 + yr + 1
        end if
    end if
else
    put char 5 to 6 of date into yr
    if yr < 80 then
        return 2000 + yr
    else
        return 1900 + yr
    end if
end if
end fiscal

function qNum date
    if the length of date = 7 then
        return "4th"
    end if
    if char 1 of date = 1 then
        return "1st"
    end if
    if char 1 of date = 4 then
        return "2nd"
    end if
    if char 1 of date = 7 then
        return "3rd"
    end if
end qNum

function dayNum start, date
    put start into tempstart
    put date into tempdate
    convert tempstart to seconds
    convert tempdate to seconds
    put tempdate + 86400 into tempdate
    return (tempdate - tempstart)/86400
end dayNum

function milDate date
    if date is empty then
        return empty
    end if
    convert date to abbr date
    if the length of item 2 of date = 6 then
        put "0" & char 6 of item 2 of date into temp
    else
        put char 6 to 7 of item 2 of date into temp
    end if
    put space & char 2 of item 2 of date after last char of temp
    put numToChar(charToNum(char 3 of item 2 of date) - 32) —
    after last char of temp
    put numToChar(charToNum(char 4 of item 2 of date) - 32) —
    after last char of temp
    put space & char 4 to 5 of item 3 of date after last char of temp
    return temp
end milDate

function dayDate start, day
    put start into tempstart
    convert tempstart to seconds
put (day - 1) * 86400 + tempstart into dateday
convert dateday to short date
return mfDate(dateday)
end dayDate

** BACKGROUND #1: Operations
on openStack
hide message box
show menuBar
pass openStack
end openStack

** CARD #1: schedules
on openCard
hide field 1
hide field 2
end openCard

** CARD #1, BUTTON #1: return
on mouseUp
go to operations
end mouseUp

** CARD #1, BUTTON #2: exit
on mouseUp
go argos
end mouseUp

** CARD #1, BUTTON #3: schedules
on mouseDown
put "EmpSkeds,New Schedule,Modify Schedule,Delete Schedule,Draw Chart" into menu 1
get HPopupMenu(menu 1, 0, 80, 65)
if it is not zero then
  Put Item 1 of it into TheLine
  put Item 2 of it into TheItem
  If TheLine = 1 and TheItem = 2 then
    go to card newsked
  end if
  If TheLine = 1 and TheItem = 3 then
    go to card modify
  end if
  If TheLine = 1 and TheItem = 4 then
    go to card skedfile
  end if
  if TheLine = 1 and TheItem = 5 then
    go to card draw
  end if
end if
end mouseDown

** CARD #2: newsked
on openCard
put card field start into oldstart
put empty into card field title
put empty into card field start
put empty into card field sked_box
put empty into card field days
ask "Title of New schedule"
if it is not empty then
    repeat until offset(",",it) = 0
        ask "No commas allowed in title, try again"
        if it is empty then
            send mouseUp to card button "return"
            exit openCard
        end if
    end repeat
    put it into card field title
    repeat until validQtr(it)
        ask "Enter calendar quarter of schedule"
        if it is empty then
            send mouseUp to card button "return"
            exit openCard
        end if
        if validQtr(it) then
            put convertQtr(it) into card field start
            select after last char of card field enter_box
            exit repeat
        else
            ask "Enter calendar quarter of schedule"
            if validQtr(it) then
                if it is empty then
                    send mouseUp to card button "return"
                    exit openCard
                end if
                put convertQtr(it) into card field start
                select after last char of card field enter_box
                exit repeat
            end if
        end if
    end repeat
    if oldstart <> card field start then
        send mouseUp to card button draw
    end if
    select after last char of card field enter_box
else
    send mouseUp to card button "return"
end if
end openCard
** CARD #2, FIELD #4: enter_box ********************************
on tabKey
    send mouseUp to card button "enter_info"
end tabKey
** CARD #2, BUTTON #1: Save Sked ******************************
on mouseUp
if card field sked_box is empty then
    answer "No schedule to save" with "return"
    exit mouseUp
else
    set cursor to 4
    put lastDay(card field start) - 1 into numdays
    repeat with j = 1 to numdays
        if line j of card field days is empty then
            answer "No major employment scheduled for " & dayDate(card field start,j) with "Return"
            exit mouseUp
        end if
    end repeat
    set cursor to 4
    set lockMessages to true
    set lockScreen to true
    put card field start into startdate
    put card field title into skedtitle
    put card field title & " " into skedname
    put the length of card field start into len
    if len = 6 then
        put (char 1 of card field start + 2)/3 after last char of skedname
        put char len - 2 to len of card field start after last char of skedname
        go to card skedname
    else
        put (char 1 to 2 of card field start + 2)/3 after last char of skedname
        put char len - 2 to len of card field start after last char of skedname
        go to card skedname
    end if
    if the result is empty then
        beep
        go to card newsked
        answer "That schedule title is saved" with "Cancel" or "Change title" or "Replace Sked"
        if it is "cancel" then
            exit mouseUp
        else
            if it is "Change title" then
                send mouseUp to card button "change title"
                exit mouseUp
            else
                go to card skedname
                put card field sked_box of card newsked into card field sked
                put skedtitle into card field title
                put startdate into card field start
                go to card newsked
                put empty into card field title
                put empty into card field sked_box
                exit mouseUp
            end if
        end if
    end if
else
set cursor to 4
set lockMessages to true
if line j of card field days is empty then
    answer "No major employment scheduled for " & dayDate(card field start,j) with "Return"
    exit mouseUp
end if
end if
end if
end if
set lockscreen to true
doMenu "new card"
set the name of this card to skedname
goto card skedname
doMenu "new field"
set name of card field 1 to "sked"
set style of card field 1 to scrolling
set rect of card field 1 to 1,26,510,280
set textfont of card field 1 to courier
set textsize of card field 1 to 12

doMenu "new field"
set name of card field 2 to "title"
set style of card field 2 to transparent
set rect of card field 2 to 0,6,220,23
set textfont of card field 2 to courier
set textsize of card field 2 to 12

doMenu "new field"
set name of card field 3 to "start"
set style of card field 3 to transparent
set rect of card field 3 to 427,6,509,23
set textfont of card field 3 to courier
set textsize of card field 3 to 12

doMenu "new field"
set name of card field 4 to "days"
set style of card field 4 to transparent
set rect of card field 4 to 1,285,30,342
set textfont of card field 4 to courier
set textsize of card field 4 to 12

card field skedbox of card newsked into card field sked
put skedtitle into card field title
put stardate into card field start
put card field days of card newsked into card field days
choose browse tool
put empty into card field title of card newsked
put empty into card field sked_box of card newsked
put empty into card field days of card newsked
goto card skedfile
put skedname & return after last char of card field listing
set lockMessages to false
goto card newsked
set lockscreen to false
end if
end mouseUp

** CARD #2, BUTTON #2: Delete Sked **********************************

on mouseUp
if card field sked_box is empty then
   answer "No schedule to delete" with "return"
else

148
put empty into card field enter_box
put empty into card field sked_box
put empty into card field title
put empty into card field start
send openCard to card newsked
end if
end mouseUp

** CARD #2, BUTTON #3: enter_info *************************************************************
on mouseUp
put empty into tempname
if card field title is empty then
  answer "Schedule must have a title" with "Cancel" or "Title"
  if it is "cancel" then
    exit mouseUp
  else
    send mouseUp to card button "change title"
  end if
end if
if card field start is empty then
  answer "No schedule quarter entered" with "Cancel" or "Enter"
  if it is "cancel" then
    exit mouseUp
  else
    send mouseUp to card button "change qtr"
    send mouseUp to card button "draw"
  end if
end if
if line 1 of card field enter_box is empty then
  ask "No event name, Enter event name"
  if it is empty then
    put empty into card field enter_box
    select after last char of card field enter_box
    exit mouseUp
  end if
find string it in field 1
if the result is not empty then
  repeat until the result is empty
    ask "Invalid event name, Enter event name"
    if it is empty then
      put empty into card field enter_box
      select after last char of card field enter_box
      exit mouseUp
    end if
    find string it in field 1
  end repeat
  put it into line 1 of card field enter_box
end if
put word 2 of the foundLine into nameloc.
if item 3 of line nameloc of field 1 is empty then
  put "A" into uwcode
else

149
put item 3 of line nameloc of field 1 into uwcode
end if
put it into line 1 of card field enter_box
else
find string line 1 of card field enter_box in field 1
if the result is not empty then
repeat until the result is empty
ask "Invalid event name, Enter event name"
if it is empty then
put empty into card field enter_box
select after last char of card field enter_box
exit mouseUp
end if
find string it in field 1
end repeat
put it into line 1 of card field enter_box
end if
put word 2 of the foundLine into nameloc
if item 3 of line nameloc of field 1 is empty then
put "A" into uwcode
else
put item 3 of line nameloc of field 1 into uwcode
end if
end if
put line 1 of card field enter_box & "," after last char of tempname
put line 2 of card field enter_box into it
repeat until validDate(it) is true and 

goodDate(card field start,it) is true
if validDate(it) is false then
ask it & & "Not a valid start date, Enter correct date"
end if
if it is empty then
put empty into card field enter_box
select after last char of card field enter_box
exit mouseUp
end if
if goodDate(card field start,it) is false then
ask "Date must be in sked qt-ter, enter date"
end if
if it is empty then
put empty into card field enter_box
select after last char of card field enter_box
exit mouseUp
end if
end repeat
put it into line 2 of card field enter_box
put line 2 of card field enter_box & "," after last char of tempname
put line 3 of card field enter_box into it
repeat until validDate(it) is true and 
goodDate(card field start,it) is true
if validDate(it) is false then
   ask it & "Not a valid end date, Enter correct date"
end if
if it is empty then
   put empty into card field enter_box
   select after last char of card field enter_box
   exit mouseUp
end if
if goodDate(card field startit) is false then
   ask "Date must be in sked quarter, enter date"
end if
if it is empty then
   put empty into card field enter_box
   select after last char of card field enter_box
   exit mouseUp
end if
end repeat
put it into line 3 of card field enter_box
put line 3 of card field enter_box & "," after last char of tempname
put uwcode & "," after last char of tempname
if the number of lines in card field sked_box >= 30 then
   answer "Schedule is full (30 Items max)" with 'Cancel' or 'Save sked'
   if it is 'cancel' then
      exit mouseUp
   else
      send mouseUp to card button "save sked"
      exit mouseUp
   end if
end if
answer "What type of employment?" with 'Major' or 'Concurrent'
if it is 'major' then
   put "M," after last char of tempname
   put dayNum(card field start item 2 of tempname) into startday
   put dayNum(card field start item 3 of tempname) into stopday
   repeat with j = startday to stopday
      if line j of card field days is not empty then
         answer "Two major employments on " & & "
         dayDate(card field start, j) with "Return"
      if j <> 1 then
         put j - 1 into erase
         repeat with k = startday to erase
            put empty into line k of card field days
         end repeat
      end if
   put empty into card field enter_box
   select after last char of card field enter_box
   exit mouseUp
end if
put "X" into line j of card field days
end repeat
else
   151
put "C," after last char of tempname
end if
if item 4 of line nameloc of field is not empty then
  ask "Enter location name"
  if it is empty then
    put "," after last char of tempname
  else
    put it & "," after last char of tempname
  end if
else
  put "," after last char of tempname
end if
if item 5 of line nameloc of field is not empty then
  ask "Enter unit name"
  if it is empty then
    put "," after last char of tempname
  else
    put it & "," after last char of tempname
  end if
else
  put "," after last char of tempname
end if
if item 6 of line nameloc of field is not empty then
  ask "Enter operation/exercise number"
  if it is empty then
    put "," after last char of tempname
  else
    put it & "," after last char of tempname
  end if
else
  put "," after last char of tempname
end if
put tempname & return after last char of card field sked_box
put empty into card field enterbox
select after last char of card field enterbox
end mouseUp

** CARD #2, BUTTON #4: Delete Item ****************************
on mouseUp
  if card field sked_box is empty then
    answer "No events to delete" with "return"
    exit mouseUp
  end if
  ask "Enter event name"
  if it is empty then
    select after last char of card field enter_box
    exit mouseUp
  else
    set cursor to 4
    put it into tgt
    repeat with j = 1 to the number of lines in card field sked_box
      if tgt = item 1 of line j of card field sked_box then

put dayNum(card field start item 2 of line j of card field sked_box) into startdate
put dayNum(card field start item 3 of line j of card field sked_box) into stopdate
repeat with x = startdate to stopdate
  put empty into line x of card field days
end repeat
delete line j of card field sked_box
select after last char of card field enter_box
exit mouseUp
end if
if j = the number of lines in card field sked_box then
  answer "Event not found" with "Return"
  select after last char of card field enter_box
end if
end repeat
end if
end mouseUp

** CARD #2, BUTTON #5: Change Title *************************************************************
on mouseUp
ask "Enter new title"
if it is empty then
  select after last char of card field enter_box
  exit mouseUp
else
  put it into card field title
  select after last char of card field enter_box
end if
end mouseUp

** CARD #2, BUTTON #6: Change Qtr *************************************************************
on mouseUp
ask "Enter new Quarter (Q/YY)"
if it is empty then
  select after last char of card field enter_box
  exit mouseUp
else
  if validQtr(it) then
    put char 1 of it * 3 - 2 into tempdate
    put "/" & 1 after last char of tempdate
    put "/" & char 3 to 4 of it after last char of tempdate
    put tempdate into card field start
    select after last char of card field enter_box
    exit mouseUp
  else
    repeat until validQtr(it)
    ask "Invalid, Enter calendar quarter of schedule"
    if it is empty then
      select after last char of card field enter_box
      exit mouseUp
    end if
  end if
end if
if validQtr(it) then
  put char 1 of it * 3 - 2 into tempdate
  put "/" & 1 after last char of tempdate
  put "/" & char 3 to 4 of it after last char of tempdate
  put tempdate into card field start
  select after last char of card field enter_box
  exit mouseUp
end if
end repeat
end if
end if
end mouseUp

** CARD #2, BUTTON #7: DRAW ***********************************************************
on mouseUp
  if card field start is empty then
    exit mouseUp
  end if
  set cursor to 4
  show card field working
  set lockScreen to true
  choose pencil tool
  click at 50,110
  click at 400,100
  choose select tool
  drag from 0,80 to 190,290
  type "x" with commandKey
  drag from 360,80 to 513,200
  type "x" with commandKey
  choose text tool
  set textSize to 9
  set textFont to geneva
  set textAlign to left
  set textStyle to plain
  set textHeight to 9
  click at 19,103
  type "S"
  click at 33,103
  type "M"
  click at 49,103
  type "T"
  click at 64,103
  type "W"
  click at 79,103
  type "T"
  click at 96,103
  type "F"
  click at 110,103
  type "S"
  put card field start into temp
  convert temp to long date
  click at 18,91
type word 1 of item 2 of temp & item 3 of temp
correct temp to dateltems
put item 2 of temp into mon
put 115 into lineloc
repeat while item 2 of temp = mon
put item 7 of temp into loc
click at 1 + 15 * loc, lineloc
if item 3 of temp > 9 then
  type item 3 of temp
else
  type " " & item 3 of temp
end if
if loc = 7 then
  put lineloc + 13 into lineloc
end if
correct temp to seconds
put 86400 + temp into temp
correct temp to dateltems
end repeat
correct temp to long date
click at 18,204
type word 1 of item 2 of temp & item 3 of temp
correct temp to dateltems
put item 2 of temp into mon
put 216 into lineloc
repeat while item 2 of temp = mon
put item 7 of temp into loc
click at 1 + 15 * loc, lineloc
if item 3 of temp > 9 then
  type item 3 of temp
else
  type " " & item 3 of temp
end if
if loc = 7 then
  put lineloc + 13 into lineloc
end if
correct temp to seconds
put 86400 + temp into temp
correct temp to dateltems
end repeat
click at 407,103
type "S"
click at 421,103
type "M"
click at 437,103
type "T"
click at 452,103
type "W"
click at 467,103
type "T"
click at 484,103
type "F"
click at 498,103
type "S"
convert temp to long date
click at 406,91
type word 1 of item 2 of temp & item 3 of temp
convert temp to dateItems
put item 2 of temp into mon
put 115 into lineloc
repeat while item 2 of temp = mon
  put item 7 of temp into loc
click at 389 + 15 * loc, lineloc
if item 3 of temp > 9 then
  type item 3 of temp
else
  type " & item 3 of temp
end if
if loc = 7 then
  put lineloc + 13 into lineloc
end if
convert temp to seconds
put 86400 + temp into temp
convert temp to dateItems
end repeat
choose browse tool
hide card field working
end mouseUp

** CARD #2, BUTTON #8: Return ****************************
on mouseUp
if card field sked_box is empty then
  go to card schedules
  exit mouseUp
end if
answer "Unsaved schedule will be lost" with "OK" or "Return"
if it is "return" then
  select after last char of card field enter_box
  exit mouseUp
end if
put empty into card field title
put empty into card field sked_box
put empty into card field enter_box
  go to card schedules
end mouseUp

** CARD #2, BUTTON #9: View Terms ****************************
on mouseUp
  push card
go to card terms
end mouseUp

** CARD #5, BUTTON #1: Return ****************************
on mouseUp
go to card schedules
end mouseUp

** CARD #5, BUTTON #2: Delete Schedule ****************************
on mouseUp
  set lockScreen to true
  set lockMessages to true
  if card field listing is empty then
    answer "No schedules on file" with "Return"
    exit mouseUp
  end if
  ask "Enter name of schedule to delete"
  if it is empty then
    send mouseUp to card button "return"
    exit mouseUp
  else
    repeat with j = 1 to the number of lines in card field listing
      if it = line j of card field listing then
        exit repeat
      end if
    end repeat
    if j = the number of lines in card field listing then
      answer "Schedule not found" with "Return"
      send mouseUp to card button "delete schedule"
      exit mouseUp
    end if
  end if
  put "card" & & quote & it & quote into tempname
  go to tempname
doMenu "delete card"
go to card skedfile
  repeat with j = 1 to the number of lines in card field listing
    if it = line j of card field listing then
      delete line j of card field listing
    end if
  end repeat
end mouseUp

** CARD #6: modify ****************************
on openCard
  put card field start into oldstart
  put card field listing of card skedfile into card field listing
  show card field listing
  hide card field title
  set lockMessages to true
  set lockScreen to true
  ask "Enter schedule name"
  if it is empty then
    send mouseUp to card button "return"
    exit openCard
  else

put "card" & quote & it & quote into tempname
go to tempname
end if
if the result is not empty then
repeat until the result is empty
  go to card "modify"
  ask "Schedule not found, Enter schedule name"
  put "card" & quote & it & quote into tempname
  put tempname into card field skedname
  put "card" & quote & it & quote into tempname
  go to tempname
end repeat
end if
put the short id of this card into card field card_id of card modify
put card field sked into card field sked_box of card modify
put card field title into card field title of card modify
put card field start into card field start of card modify
go to card "modify"
set lockscreen to false
if oldstart <> card field start then
  send mouseUp to card button draw
end if
hide card field listing
show card field title
select after last char of card field enter_box
end openCard
** CARD #6, FIELD #1: enter_box ******************************************************************************
on tabKey
  send mouseUp to card button "enter_info"
end tabKey
** CARD #6, BUTTON #1: Return ******************************************************************************
on mouseUp
  if card field sked_box is empty then
    go to card schedules
    exit mouseUp
  end if
  answer "Unsaved changes will be lost" with "OK" or "Return"
  if it is "return" then
    exit mouseUp
  end if
  put empty into card field sked_box
  put empty into card field sked
  put empty into card field title
  go to card schedules
end mouseUp
** CARD #6, BUTTON #2: Delete Event ******************************************************************************
on mouseUp
  if visible of card field listing is true then
    answer "Get schedule first" with "Cancel" or "Get Sked"
if it is "cancel" then
  exit mouseUp
else
  send mouseUp to card button "get sked"
  exit mouseUp
end if

ask "Enter Event name"
if it is empty then
  exit mouseUp
else
  put it into tgt
  repeat with j = I to the number of lines in card field sked_box
    if tgt = item 1 of line j of card field sked_box then
      delete line j of card field sked_box
      exit mouseUp
    end if
  if j = the number of lines in card field sked_box then
    answer "Event not found" with "Return"
  end if
  end repeat
end if
end mouseUp

** CARD #6, BUTTON #3: Change Date **************************************************
on mouseUp
  if visible of card field listing is true then
    answer "Get schedule first" with "Cancel" or "Get Sked"
    if it is "cancel" then
      exit mouseUp
    else
      send mouseUp to card button "get sked"
      exit mouseUp
    end if
  end if
  ask "Enter Event name"
  if it is empty then
    exit mouseUp
  else
    put it into tgt
    repeat with j = I to the number of lines in card field sked_box
      if tgt = item 1 of line j of card field sked_box then
        put j into linenum
        exit repeat
      end if
    if j = the number of lines in card field sked_box then
      answer "Event not found" with "Return"
    end if
    end repeat
    repeat with j = 1 to linenum
      ask "Enter new start date"
...

159
if it is empty then
exit repeat
else
repeat until validDate(it) is true and —
goodDate(card field start,it) is true
if validDate(it) is false then
ask it && "Not a valid start date, Enter correct date"
end if
if it is empty then
exit repeat
end if
if goodDate(card field start,it) is false then
ask "Date must be in sked quarter, enter date"
end if
if it is empty then
exit repeat
end if
end repeat
put it into item 2 of line linenum of card field sked_box
end if
end repeat
ask "Enter new end date"
if it is empty then
exit mouseUp
else
repeat until validDate(it) is true and —
goodDate(card field start,it) is true
if validDate(it) is false then
ask it && "Not a valid end date, Enter correct date"
end if
if it is empty then
select after last char of card field enter_box
exit mouseUp
end if
if goodDate(card field start,it) is false then
ask "Date must be in sked quarter, enter date"
end if
if it is empty then
select after last char of card field enter_box
exit mouseUp
end if
end repeat
put it into item 3 of line linenum of card field sked_box
end if
end if
end mouseUp

** CARD #6, BUTTON #4: Add Event

on mouseUp
if visible of card field listing is true then
answer "Get schedule first" with "Cancel" or "Get Sked"
if it is "cancel" then
exit mouseUp
else
    send mouseUp to card button "get sked"
    exit mouseUp
end if
end if
put empty into card field enter_box
select after last char of card field enter_box
end mouseUp

** CARD #6, BUTTON #5: Save Changes

on mouseUp
if visible of card field listing is true then
    answer "Get schedule first" with "Cancel" or "Get Sked"
    if it is "cancel" then
        exit mouseUp
    else
        send mouseUp to card button "get sked"
        exit mouseUp
    end if
end if
put card field card_id into card_id
put card field sked_box into card field sked of card id card_id
put empty into card field skedname
put empty into card field sked_box
put empty into card field title
put empty into card field enter_box
send openCard to card modify
end mouseUp

** CARD #6, BUTTON #6: Cancel Changes

on mouseUp
if visible of card field listing is true then
    answer "Get schedule first" with "Cancel" or "Get Sked"
    if it is "cancel" then
        exit mouseUp
    else
        send mouseUp to card button "get sked"
        exit mouseUp
    end if
end if
put card field card_id into card_id
put card field sked of card id card_id into card field sked_box
select after last char of card field enter_box
end mouseUp

** CARD #6, BUTTON #7: DRAW

on mouseUp
if card field start is empty then
    exit mouseUp
end if
set cursor to 4
show card field working
set lockScreen to true
choose pencil tool
click at 50,110
click at 400,100
choose select tool
drag from 0,80 to 160,290
type "x" with commandKey
drag from 360,80 to 513,200
type "x" with commandKey
choose text tool
set textSize to 9
set textFont to geneva
set textAlign to left
set textStyle to plain
set textHeight to 9
click at 49,103
type "S"
click at 63,103
type "M"
click at 79,103
type "T"
click at 94,103
type "W"
click at 109,103
type "T"
click at 126,103
type "F"
click at 140,103
type "S"
put card field start into temp
convert temp to long date
click at 48,91
type word 1 of item 2 of temp && item 3 of temp
convert temp to dateItems
put item 2 of temp into mon
put 115 into lineloc
repeat while item 2 of temp = mon
  put item 7 of temp into loc
click at 31 + 15 * loc,lineloc
  if item 3 of temp > 9 then
    type item 3 of temp
  else
    type " " & item 3 of temp
  end if
  if loc = 7 then
    put lineloc + 13 into lineloc
  end if
convert temp to seconds
put 86400 + temp into temp
convert temp to dateItems
end repeat
convert temp to long date
click at 48,204
type word 1 of item 2 of temp & item 3 of temp
click at 386,91
type word 1 of item 2 of temp & item 3 of temp
click at 386,91
type word 1 of item 2 of temp & item 3 of temp
convert temp to dateitems
put item 2 of temp into mon
put 216 into lineloc
repeat while item 2 of temp = mon
  put item 7 of temp into loc
  click at 31 + 15 * loc,lineloc
  if item 3 of temp > 9 then
    type item 3 of temp
    if loc = 7 then
      put lineloc + 13 into lineloc
      end if
  else
    type " " & item 3 of temp
  end if
if loc = 7 then
  put lineloc + 13 into lineloc
  end if
convert temp to seconds
put 86400 + temp into temp
convert temp to dateitems
end repeat

163
convert temp to seconds
put 86400 + temp into temp
convert temp to dateitems
end repeat
choose browse tool
hide card field working
end mouseUp

** CARD #6, BUTTON #8: enter into *******************************************
on mouseUp
put empty into tempname
if card field title is empty then
answer "Schedule must have a title" with "Cancel" or "Title"
if it is "cancel" then
exit mouseUp
else
send mouseUp to card button "change title"
end if
end if
if card field start is empty then
answer "No schedule quarter entered" with "Cancel" or "Enter"
if it is "cancel" then
exit mouseUp
else
send mouseUp to card button "change qtr"
send mouseUp to card button "draw"
end if
end if
if line 1 of card field enter_box is empty then
ask "No event name, Enter event name"
if it is empty then
put empty into card field enter_box
select after last char of card field enter_box
exit mouseUp
end if
put it into line 1 of card field enter_box
end if
put line 1 of card field enter_box & "," after last char of tempname
put line 2 of card field enter_box into it
repeat until validDate(it) is true and ~
goodDate(card field start,it) is true
if validDate(it) is false then
ask it & & "Not a valid start date, Enter correct date"
end if
if it is empty then
select after last char of card field enter_box
exit mouseUp
end if
if goodDate(card field start,it) is false then
ask "Date must be in sked quarter, enter date"
end if
if it is empty then
    select after last char of card field enter_box
    exit mouseUp
end if
end repeat

put it into line 2 of card field enter_box
put line 2 of card field enter_box & "," after last char
of tempname
put line 3 of card field enter_box into it
repeat until validDate(it) is true and ~
goodDate(card field start,it) is true
    if validDate(it) is false then
        ask it & & "Not a valid end date, Enter correct date"
    end if
    if it is empty then
        select after last char of card field enter_box
        exit mouseUp
    end if
    if goodDate(card field start,it) is false then
        ask "Date must be in sked quarter, enter date"
    end if
    if it is empty then
        select after last char of card field enter_box
        exit mouseUp
    end if
end repeat

put it into line 3 of card field enter_box
put line 3 of card field enter_box & return after last char of ~
tempname
if the number of lines in card field sked_box >= 15 then
    answer "Schedule is full (15 Items max)" with "Cancel" or ~
    "Save sked"
    if it is "cancel" then
        exit mouseUp
    else
        send mouseUp to card button "save sked"
        exit mouseUp
    end if
end if

put tempname after last char of card field sked_box
put empty into card field enter_box
select after last char of card field enter_box
end mouseUp

** CARD #6, BUTTON #9: Get Sked **************************************
on mouseUp
    put card field start into oldstart
    put card field listing of card skedfile into card field listing
    show card field listing
    hide card field title
    set lockMessages to true
    set lockScreen to true
ask "Enter schedule name"
if it is empty then
    send mouseUp to card button "return"
    exit mouseUp
else
    repeat with j = 1 to the number of lines in card field listing
    if it = line j of card field listing then
        exit repeat
    end if
    if j = the number of lines in card field listing then
        answer "Schedule not found" with "Return"
        send mouseUp to card button "get sked"
        exit mouseUp
    end if
end repeat
put quote & it & quote into tempname
put tempname into card field skedname
put "card" & quote & it & quote into tempname
go to tempname
put the short id of this card into card field card_id of card modify
put card field sked into card field sked_box of card modify
put card field title into card field title of card modify
put card field start into card field start of card modify
go to card "modify"
set lockscreen to false
if oldstart <> card field start then
    send mouseUp to card button draw
end if
hide card field listing
show card field title
select after last char of card field enter_box
end if
end mouseUp

** CARD #6, BUTTON #10: View Terms

on mouseUp
    push card
    go to card terms
end mouseUp

** CARD #7: terms

on openCard
    show field 2
end openCard

** CARD #7, BUTTON #1: New Button

on mouseUp
    set lockMessages to true
    hide field 2
    pop card
end mouseUp

** CARD #8: draw
on openCard
  put card field listing of card skedfile into card field listing
  show card field header
  show card field list_head
  show card field listing
end openCard
on idle
  hide card field shield
end idle
** CARD #8, BUTTON #1: Get Sked *********************************************
on mouseUp
  put card field start into oldstart
  put card field listing of card skedfile into card field listing
  show card field listing
  hide card field title
  set lockMessages to true
  set lockScreen to true
  ask "Enter schedule name"
  if it is empty then
    exit mouseUp
  else
    repeat with j = 1 to the number of lines in card field listing
      if it = line j of card field listing then
        exit repeat
      end if
    end repeat
    if j = the number of lines in card field listing then
      answer "Schedule not found" with "Return"
      send mouseUp to card button "get sked"
      exit mouseUp
    end if
  end if
  put quote & it & quote into tempname
  put "card" & & quote & it & quote into tempname
  go to tempname
  put card field sked into card field sked_box of card draw
  put card field title into card field title of card draw
  put card field start into card field start of card draw
  go to card "draw"
  hide card field list_head
  hide card field listing
  set lockscreen to false
end if
end mouseUp

** CARD #8, BUTTON #2: Return *********************************************
on mouseUp
  send mouseUp to card button erase
  put empty into card field sked_box
  put empty into card field start
  put empty into card field title
  put empty into card field listing
  go to card schedules
end mouseUp

** CARD #8, BUTTON #3: Print ******************************************
on mouseUp
  set cursor to 4
  show card field shield
  open printing
  print this card
  close printing
end mouseUp

** CARD #8, BUTTON #4: Erase ******************************************
on mouseUp
  set cursor to 4
  set lockScreen to true
  choose pencil tool
  drag from 0,0 to 0,1
  choose select tool
  drag from 0,0 to 512,342
  type "x" with commandKey
  show card field header
  choose browse tool
end mouseUp

** CARD #8, BUTTON #5: Draw ******************************************
on mouseUp
  set cursor to 4
  send mouseUp to card button "get sked"
  if card field start is empty then
    send mouseUp to card button "return"
    exit mouseUp
  end if
  hide card field header
  hide card field listing
  hide card field list_head
  put 0 into underway
  put 0 into column
  put 0 into noteline
  put 1 into notenum
  repeat with j = 1 to 10
    put j + 8 into emptyfield
    put empty into card field emptyfield
  end repeat
  choose bucket tool
  set pattern to 1
  click at 100,100
  choose browse tool
  put card field start into begin
  convert begin to dateitems
  get last char of begin
  put it into temp
  if temp > 2 then
put temp - 3 into temp
else
  if temp = 2 then
    put 6 into temp
  else
    put 5 into temp
  end if
end if
put 55 - 5 * temp into location
put begin into tempdate
put last char of begin into temp
put (9 - temp) mod 7 into temp
convert tempdate to seconds
put tempdate + 86400 * temp into tempdate
set the textFont to courier
set the textSize to 10
set the textStyle to plain
set the textHeight to 10
set textAlign to left
choose line tool
put card field start into now
put lastDay(now) into ending
drag from 25,61 to ending * 5 + 20,61
drag from 25,68 to ending * 5 + 20,68
put 20 into Ioc
repeat with j = I to ending
  drag from loc + 5 * j,62 to loc + 5 * j,67
end repeat
repeat until location > 465
  choose line tool
  drag from location,53 to location,61
  choose text tool
  convert tempdate to abbr date
  put the length of word 3 of tempdate into len
  put char 1 to len - 1 of word 3 of tempdate into day
  click at location + 2,58
type day
  if day < 8 then
    put char 2 of item 2 of tempdate into mon
    put numToChar(charToNum(char 3 of item 2 of tempdate) - 32) after-
      last char of mon
    put numToChar(charToNum(char 4 of item 2 of tempdate) - 32) after-
      last char of mon
type mon
  end if
  convert tempdate to seconds
  put 604800 + tempdate into tempdate
  put location + 35 into location
end repeat
click at 256,25
set textAlign to center
set textHeight to 12
set textSize to 12

set the textFont to courier
set the textStyle to plain
set the textHeight to 10
set textAlign to left

put card field start into startdate
convert startdate to seconds
put empty into card field notes

repeat with j = 1 to the number of lines in card field sked_box

choose line tool
put item 2 of line j of card field sked_box into tempstart
convert tempstart to seconds
put ((tempstart - startdate)/86400) * 5 + 25 into startloc
put item 3 of line j of card field sked_box into tempstop
convert tempstop to seconds
put ((tempstop - startdate)/86400) * 5 + 30 into stoploc

if item 5 of line j of card field sked_box = "m" then

choose text tool
click at startloc + 1,77
put stoploc - startloc into interval
put the length of item 1 of line j of card field sked_box * 6 into len

if interval < len + 1 then

 type notenum
put notenum + 1 into notenum
put item 1 of line j of card field sked_box & return ; after last char of card field notes
else

 type item 1 of line j of card field sked_box
end if

choose line tool
drag from startloc,70 to startloc,80
drag from stoploc,70 to stoploc,80
drag from startloc,78 to stoploc,78

set textFont to courier

if item 4 of line j of card field sked_box = "b" then

choose bucket tool
set pattern to 22
put startloc + 1 into loc
repeat until loc > stoploc
put underway + 1 into underway
click at loc,62
put loc + 5 into loc
end repeat
end if

choose text tool
else
repeat with x = 1 to 10
put x + 8 into fieldnum
if line 1 of card field fieldnum > stoploc or
line 2 of card field fieldnum < startloc then
put 80 + 15 * x into lineloc
choose text tool
click at startloc + 1, lineloc - 3
put stoploc - startloc into interval
put the length of item 1 of line j of card field sked_box * 6 -
into len
if interval < len + 1 then
type notenum
put notenum + 1 into notenum
put item 1 of line j of card field sked_box & return after
last char of card field notes
else
type item 1 of line j of card field sked_box
end if
choose line tool
drag from startloc,lineloc - 10 to startloc,lineloc
drag from stoploc,lineloc - 10 to stoploc,lineloc
drag from startloc,lineloc - 2 to stoploc,lineloc - 2
set textFont to courier
if card field fieldnum is empty then
put startloc into line 1 of card field fieldnum
put stoploc into line 2 of card field fieldnum
exit repeat
end if
if line 1 of card field fieldnum > startloc then
put startloc into line 1 of card field fieldnum
exit repeat
else
if line 2 of card field fieldnum < stoploc then
put stoploc into line 2 of card field fieldnum
exit repeat
end if
end if
end repeat
end if
end repeat
choose text tool
set numberFormat to "00"
put underway/(ending - 1) * 100 into optempo
click at 25,45
type "DAYS UNDERWAY:" && underway
click at 400,45
type "OPTEMPO:" && optempo & 
set numberFormat to "0.######"
choose line tool
drag from 30,260 to 483,260
choose text tool
click at 30,257
type "NOTES:"
repeat with j = 1 to the number of lines in card field notes
  click at 30 + column * 140,270 + noteline
  type j && line j of card field notes
  put noteline + 10 into noteline
  if j mod 4 = 0 then
    put 0 into noteline
    put column + 1 into column
  end if
end repeat
choose browse tool
reset paint
end mouseUp
APPENDIX F. PUBLICATIONS STACK SCRIPTS

SCRIPTS FOR STACK: pubs

** STACK SCRIPT ***********************************************
on openStack
  put field 1 of card oporders into field 1 of card dummy
end openStack
function goodDate date
  if the length of date <> 5 then
    return false
  end if
  if char 1 of date is not in "01" then
    return false
  end if
  if char 1 of date = "0" then
    if char 2 of date is not in "123456789" then
      return false
    end if
  else
    if char 1 of date = "1" then
      if char 2 of date is not in "012" then
        return false
      end if
    end if
  end if
  if char 4 of date is not in "0123456789" then
    return false
  end if
  if char 5 of date is not in "0123456789" then
    return false
  end if
  if char 3 of date <> "/" then
    return false
  end if
  return true
end goodDate
function goodNum Num
  if the length of num > 2 then
    return false
  end if
  if the length of num = 2 then
    if char 1 of num is not in "123456789" then
      return false
    else
      if char 2 of num is not in "123456789" then
        return false
      end if
    end if
  end if
end goodNum
end if
end if
else
if char 1 of num is not in "123456789" then
return false
end if
end if
return true
goodNum

** BACKGROUND #1: Operations *************************************************************
on openStack
  hide message box
  show menuBar
  pass openStack
end openStack

** CARD #1, BUTTON #1: RETURN *************************************************************
on mouseUp
  go to operations
end mouseUp

** CARD #1, BUTTON #2: exit ***************************************************************
on mouseUp
  go argos
end mouseUp

** C * ?D #1, BUTTON #3: pubs *************************************************************
on mouseDown
  put "OPORDERS & OPLANS" into menu1
  put return & "INSTRUCTIONS" after menu1
  put return & "NWP" after menu1
  get HPopupMenu(menu1,0,88,102)
  if it is not zero then
    Put Item 1 of it into TheLine
    put Item 2 of it into Theltem
    if TheLine= 1 and Theltem=1 then
      GO NEXT
    end if
  end if
end mouseDown

** CARD #2, BUTTON #1: Find ***************************************************************
on mouseUp
  ask "Enter pub title"
  if it is empty then
    exit mouseUp
  end if
  put it & "," into findstring
  ask "Enter annex" with "None"

if it is empty then
  exit mouseUp
end if
if it <> "none" then
  if it is not in "abcdefghijklmnopqrstuvwxyz" then
    repeat until it is in "abcdefghijklmnopqrstuvwxyz"
    ask "Invalid entry (Must be a letter)" with "NONE"
  end repeat
  end if
  put it &, "," after last char of findstring
else
  put "0," after last char of findstring
end if
ask "Enter appendix" with "None"
if it is empty then
  exit mouseUp
end if
if it <> "none" then
  if goodNum(it) is false then
    repeat until goodNum(it) is true
    ask "Invalid entry (Must be a number)" with "NONE"
    if it is empty then
      exit mouseUp
    end if
  end repeat
  end if
  put it &, "," after last char of findstring
else
  put "0," after last char of findstring
end if
ask "Enter tab" with "None"
if it is empty then
  exit mouseUp
end if
if it <> "none" then
  if it is not in "abcdefghijklmnopqrstuvwxyz" then
    repeat until it is in "abcdefghijklmnopqrstuvwxyz"
    ask "Invalid entry (Must be a letter)" with "NONE"
  end repeat
  end if
  put it &, "," after last char of findstring
else
  put "0," after last char of findstring
end if
find string findstring in field
if the result is not empty then
  answer "Publication is not on record" with "return"
  exit mouseUp
end if
put word 2 of the foundLine into here
put line here of field 2 into card field view
end mouseUp
** CARD #2, BUTTON #2: Enter New Pub ****************************

on mouseUp
  ask "Enter pub title"
  if it is empty then
    exit mouseUp
  end if
  put it & "," into templine
  ask "Enter annex" with "None"
  if it is empty then
    exit mouseUp
  end if
  if it <> "none" then
    if it is not in "abcdefghijklmnopqrstuvwxyz" then
      repeat until it is in "abcdefghijklmnopqrstuvwxyz"
        ask "Invalid entry (Must be a letter)" with "NONE"
        end repeat
    end if
    put it & "," after last char of templine
  else
    put "0," after last char of templine
  end if
  ask "Enter appendix" with "None"
  if it is empty then
    exit mouseUp
  end if
  if it <> "none" then
    if goodNum(it) is false then
      repeat until goodNum(it) is true
        ask "Invalid entry (Must be a number)" with "NONE"
        if it is empty then
          exit mouseUp
        end if
      end repeat
    end if
    put it & "," after last char of templine
  else
    put "0," after last char of templine
  end if
  ask "Enter tab" with "None"
  if it is empty then
    exit mouseUp
  end if
  if it <> "none" then
    if it is not in "abcdefghijklmnopqrstuvwxyz" then
      repeat until it is in "abcdefghijklmnopqrstuvwxyz"
        ask "Invalid entry (Must be a letter)" with "NONE"
        end repeat
    end if
    put it & "," after last char of templine
  else
    put "0," after last char of templine
  end if
end if

ask "Enter Effective date (MM/YY)"
if it is empty then
  exit mouseUp
end if

if goodDate(it) is false then
  repeat until goodDate(it) is true
    ask "Invalid entry (Use leading zero)"
    if it is empty then
      exit mouseUp
    end if
  end repeat
end if

put it & "," after last char of templine

ask "Enter classification (T,S,C, or U)"
if it is empty then
  exit mouseUp
end if

if it is not in "tscu" then
  repeat until it is in "tscu"
    ask "Invalid entry (Must be T,S,C, or U)"
    if it is empty then
      exit mouseUp
    end if
  end repeat
end if

put it & "," after last char of templine

ask "Enter latest change number" with "NONE"
if it is empty then
  exit mouseUp
end if

if it <> "none" then
  if goodNum(it) is false then
    repeat until goodNum(it) is true or it = "NONE"
      ask "Invalid entry (Must be a number)" with "NONE"
      if it is empty then
        exit mouseUp
      end if
    end repeat
  end if
else
  put "0" into it
end if

put it & "," after last char of templine

ask "Enter Location/Custodian"
if it is empty then
  exit mouseUp
end if

put it after last char of templine

put item 1 of templine & "," & item 2 of templine & "," & item 3 of templine & "," & item 4 of templine into sortstring
repeat with j = 1 to the number of lines in field 1
put item 1 of line j of field 1 & "," & item 2 of line j of field 1 & "," & item 3 of line j of field 1 & "," & item 4 of line j of field 1 into findstring
if sortstring > findstring then
  if j = the number of lines in field 1 then
    put return & templine after line j of field 1
    put false into firstline
    put true into insert
    put j into here
    exit repeat
  end if
next repeat
else
  if sortstring = findstring then
    put templine into line j of field 1
    put false into insert
    put false into firstline
    put j into here
    exit repeat
  else
    if j = 1 then
      put templine & return before line 1 of field 1
      put true into firstline
    exit repeat
    else
      put return & templine after line j - 1 of field 1
      put false into firstline
      put true into insert
      put j - 1 into here
      exit repeat
    end if
  end if
end if
end repeat
put item 1 of templine into tempadd
repeat until the length of tempadd = 37
  put space after last char of tempadd
end repeat
if "0" is in item 2 of templine then
  put space & space & space after last char of tempadd
else
  put space & item 2 of templine & space after last char of tempadd
end if
if "0" is in item 3 of templine then
  put space & space & space after last char of tempadd
else
  if the length of item 3 of templine = 2 then
    put item 3 of templine & space after last char of tempadd
  else
    put space & item 3 of templine & space after last char of tempadd
  end if
end if
if "0" is in item 4 of templine then
  put space & space after last char of tempadd
else
  put item 4 of templine & space after last char of tempadd
end if

if item 7 of templine = "NONE" then
  put "0" & space after last char of tempadd
else
  put " & item 7 of templine & space after last char of tempadd
end if

if firstline is true then
  put tempadd & return before line 1 of field 2
  exit mouseUp
end if

if insert is true then
  put return & tempadd after line here of field 2
else
  put tempadd into line here of field 2
end if

put field 1 into field 1 of card dummy

** CARD #2, BUTTON #3: Delete ******************************************
on mouseUp
  ask "Enter pub title"
  if it is empty then
    exit mouseUp
  end if
  put it &, " into findstring
  ask "Enter annex" with "None"
  if it is empty then
    exit mouseUp
  end if
  if it <> "none" then
    if it is not in "abcdefghijklmnopqrstuvwxyz" then
      repeat until it is in "abcdefghijklmnopqrstuvwxyz"
      ask "Invalid entry (Must be a letter)" with "NONE"
      end repeat
    end if
    put it &, " after last char of findstring
  else
    put "0," after last char of findstring
  end if
  ask "Enter appendix" with "None"
  if it is empty then
    exit mouseUp
  end if
  if it <> "none" then
if goodNum(it) is false then
    repeat until goodNum(it) is true
        ask "Invalid entry (Must be a number)" with "NONE"
        if it is empty then
            exit mouseUp
        end if
    end repeat
    put it &, "," after last char of findstring
else
    put "0," after last char of findstring
end if
ask "Enter tab" with "None"
if it is empty then
    exit mouseUp
end if
if it <> "none" then
    if it is not in "abcdefghijklmnopqrstuvwxyz" then
        repeat until it is in "abcdefghijklmnopqrstuvwxyz"
            ask "Invalid entry (Must be a letter)" with "NONE"
        end repeat
    end if
    put it &, "," after last char of findstring
else
    put "0," after last char of findstring
end if
find string findstring in field 1
if the result is not empty then
    answer "Publication is not on record" with "return"
    exit mouseUp
end if
put word 2 of the foundLine into trash
delete line trash of field 1
delete line trash of field 2
put field 1 into field 1 of card dummy
end mouseUp

** CARD #2, BUTTON #4: Selective List **********************
on mouseUp
    put empty into card field view
    answer "Choose list criterion" with "Classification" or "Originator"-
or "Custodian"
    if it is "Classification" then
        ask "Enter classification (T,S,C, or U)"
        if it is empty then
            exit mouseUp
        end if
        if it <> "t" and it <> "s" and it <> "c" and it <> "u" then
            repeat until it = "t" or it = "s" or it = "c" or it = "u"
                ask "Invalid classification (T,S,C, or U)"
            end repeat
        end if
    end if
end if
end repeat
end if
put it into class
repeat with \( j = 1 \) to the number of lines in field 1
  if item 6 of line \( j \) of field 1 = class then
    put line \( j \) of field 2 & return after last char of card field view
  end if
end repeat
end if
if it is "originator" then
  ask "Enter originator name"
  if it is empty then
    exit mouseUp
  end if
  put it into orig
  set cursor to 4
  set lockScreen to true
  repeat
    find whole orig in field 1
    if the result <> empty then
      if card field view is empty then
        answer "No pubs originated by" && orig && "on file" with "Return"
        exit mouseUp
      end if
      else
        if the short id of this card <> 4762 then
          go to card oporders
        end if
        put word 2 of the foundLine into linenum
        put line linenum of field 2 & return after last char of card field view
      end if
    end if
  end repeat
end if
if it is "custodian" then
  ask "Enter custodian/location"
  if it is empty then
    exit mouseUp
  else
    put it into thename
    set cursor to 4
    set lockScreen to true
    repeat
      find whole thename in field 1
      if the result <> empty then
        if card field view is empty then
          answer "No pubs held by" && thename && "on file" with "Return"
        end if
      end if
    end repeat
end if
exit mouseUp
end if
else
if the short id of this card <> 4762 then
go to card oporders
exit mouseUp
end if
put word 2 of the foundLine into linenum
put line linenum of field 2 & return after last char of →
card field view
end if
end repeat
end if
end if
end mouseUp

** CARD #2, BUTTON #5: List *******************************
on mouseUp
put field 2 into card field view
end mouseUp

** CARD #2, BUTTON #6: RETURN *******************************
on mouseUp
go first
end mouseUp

** CARD #2, BUTTON #7: Change Data *******************************
on mouseUp
ask "Enter pub fitle"
if it is empty then
exit mouseUp
end if
put it &, " into findstring
ask "Enter annex" with "None"
if it is empty then
exit mouseUp
end if
if it <> "none" then
if it is not in "abcdefghijklmnopqrstuvwxyz" then
repeat until it is in "abcdefghijklmnopqrstuvwxyz"
ask "Invalid entry (Must be a letter)" with "NONE"
end repeat
end if
put it &, " after last char of findstring
else
put "0," after last char of findstring
end if
ask "Enter appendix" with "None"
if it is empty then
exit mouseUp
end if
if it <> "none" then
if goodNum(it) is false then  
   repeat until goodNum(it) is true  
      ask "Invalid entry (Must be a number)" with "NONE"  
   end if  
end repeat  
else  
   put it after last char of findstring  
end if  

ask "Enter tab" with "None"  
if it is empty then  
   exit mouseUp  
end if  
if it <> "none" then  
   if it is not in "abcdefghijlmonoprstuwxzy" then  
      repeat until it is in "abcdefghijlmonoprstuwxzy"  
         ask "Invalid entry (Must be a letter)" with "NONE"  
      end repeat  
   end if  
   put it after last char of findstring  
else  
   put "0," after last char of findstring  
end if  
find string findstring in field 1  
if the result is not empty then  
   answer "Publication is not on record" with "return"  
   exit mouseUp  
end if  
put word 2 of the foundLine into change  
answer "Choose data item to change" with "Change #" or "Custodian"  
if it is "change #" then  
   ask "Enter latest change number"  
   if it is empty then  
      exit mouseUp  
   else  
      if goodNum(it) is false then  
         repeat until goodNum(it) is true  
            ask "Invalid entry (Must be a number)"  
         end if  
      end if  
      put it into line change of field 1  
      put it into char 59 of line change of field 2  
   else  
      ask "Enter new custodian"  
      if it is empty then  
         exit mouseUp  
      end if  
   end if  
else  
   ask "Enter new custodian"  
   if it is empty then  
      exit mouseUp  
   end if  
end if  

exit mouseUp
else
  put char 1 to 20 of it into item 8 of line change of field 1
  put char 1 to 20 of it into char 62 to 81 of line change of field 2
end if
end if
end mouseUp
BIBLIOGRAPHY


<table>
<thead>
<tr>
<th>No.</th>
<th>Copies</th>
<th>Initial Distribution List</th>
</tr>
</thead>
</table>
| 1.  | 2      | Defense Technical Information Center  
Cameron Station  
Alexandria, VA 22304-6145 |
| 2.  | 2      | Library, Code 0142  
Naval Postgraduate School  
Monterey, CA 93943-5002 |
| 3.  | 1      | Office of Naval Research  
Office of the Chief of Naval Research  
Attn: CDR Michael Gehl, Code 1224  
800 N. Quincy St.  
Arlington, VA 22217-5000 |
| 4.  | 1      | Space and Naval Warfare Systems Command  
Attn: LCDR Topoeroff  
Nation Center 1, Room 11N08  
2511 Jefferson Davis Hwy  
Washington, DC 20363-5100 |
| 5.  | 1      | Office of the Chief of Naval Operations  
Attn: CAPT Don Rhodes  
Code OP-403  
Washington, DC 20350-2000 |
| 6.  | 1      | Department of the Navy  
Naval Sea System Command  
Attn: Mr. Clifford Geiger  
Code: Cheng L  
Washington, DC 20362-5101 |
| 7.  | 1      | Office of the Secretary of Defense  
Attn: CDR Barber  
STARS Program Office  
Washington, DC 20301 |
| 8.  | 1      | Office of the Secretary of Defense  
Attn: Mr. Joel Trimble  
STARS Program Office  
Washington, DC 20301 |
| 9.  | 1      | Commanding Officer  
Naval Research Laboratory  
Code 5150  
Attn: Dr. Elizabeth Wald  
Washington, DC 20375-5000 |
<table>
<thead>
<tr>
<th>#</th>
<th>Organization</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Naval Ocean Systems Center</td>
<td>Attn: Linwood Sutton, Code 423 San Diego, CA 92152-5000</td>
</tr>
<tr>
<td>11</td>
<td>National Science Foundation</td>
<td>Division of Computer and Computation Research Washington, DC 20550</td>
</tr>
<tr>
<td>12</td>
<td>Department of the Navy</td>
<td>Naval Sea Systems Command Attn: Mr. Phil Styles Code: CEL-TD1 Washington, DC 20362-5101</td>
</tr>
<tr>
<td>13</td>
<td>Department of the Navy</td>
<td>Naval Sea Systems Command Attn: Mr. Mike Mehalic Code: CEL-PAB Washington, DC 20362-5101</td>
</tr>
<tr>
<td>14</td>
<td>Office of Naval Research</td>
<td>Computer Science Division, Code: 1133 Attn: Dr. Van Tilburg 800 N. Quincy St. Arlington, VA 22217-5000</td>
</tr>
<tr>
<td>15</td>
<td>David W. Taylor Naval Ship R&amp;D Center</td>
<td>Attn: Mr. J. Hawkins Code:1740.2 Bethesda, MD 20084-5000</td>
</tr>
<tr>
<td>16</td>
<td>Navy Management Systems Support Office</td>
<td>Detachment Pacific Attn: Mr. Lyle Rich Code: 311 Naval Station, Box 217 San Diego, CA 92136-5217</td>
</tr>
<tr>
<td>17</td>
<td>Prof. C. T. Wu</td>
<td>Code: 52Wq Naval Postgraduate School Monterey, CA 93943</td>
</tr>
<tr>
<td>18</td>
<td>Prof. D. K. Hsiao</td>
<td>Code: 52Hq Naval Postgraduate School Monterey, CA 93943</td>
</tr>
<tr>
<td>19</td>
<td>LT William R. Ault</td>
<td>Class 109 Surface Warfare Officer School Command Newport, RI 02841</td>
</tr>
</tbody>
</table>