

MICROCOPY

10121

AD-A169 183

NAVAL POSTGRADUATE SCHOOL Monterey, California



DTIC
ELECTE
JUL 02 1986
S D D

THESIS

THE DESIGN AND DEVELOPMENT OF A MANAGEMENT
INFORMATION SYSTEM FOR THE MONTEREY NAVY FLYING CLUB

by

Derek R. George

March 1986

Co-Advisors: Jack Lamatra
Barry Frew

DTIC FILE COPY

Approved for public release; distribution is unlimited.

NO ALC 100

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION		1b RESTRICTIVE MARKINGS	
2a SECURITY CLASSIFICATION AUTHORITY		3 DISTRIBUTION/AVAILABILITY OF REPORT	
2b DECLASSIFICATION/DOWNGRADING SCHEDULE		Approved for public release; distribution is unlimited.	
4 PERFORMING ORGANIZATION REPORT NUMBER(S)		5 MONITORING ORGANIZATION REPORT NUMBER(S)	
6a NAME OF PERFORMING ORGANIZATION	6b OFFICE SYMBOL (if applicable)	7a. NAME OF MONITORING ORGANIZATION	
Naval Postgraduate School	54	Naval Postgraduate School	
6c ADDRESS (City, State, and ZIP Code)		7b ADDRESS (City, State, and ZIP Code)	
Monterey, CA 93943		Monterey, CA 93943	
8a NAME OF FUNDING/SPONSORING ORGANIZATION	8b OFFICE SYMBOL (if applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
8c ADDRESS (City, State, and ZIP Code)		10 SOURCE OF FUNDING NUMBERS	
		PROGRAM ELEMENT NO	PROJECT NO
		TASK NO	WORK UNIT ACCESSION NO
11 TITLE (Include Security Classification)			
THE DESIGN AND DEVELOPMENT OF A MANAGEMENT INFORMATION SYSTEM FOR THE MONTEREY NAVY FLYING CLUB			
12 PERSONAL AUTHOR(S)			
GEORGE, Derek R.,			
13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Master's Thesis	FROM _____ TO _____	27 March 1986	123
16 SUPPLEMENTARY NOTATION			
17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	PROTOTYPE, MIS
19 ABSTRACT (Continue on reverse if necessary and identify by block number)			
This thesis provides a Management Information System for the Monterey Navy Flying Club. It supplies the tools necessary to enable the club manager to maintain all club records and generate required administrative and financial reports.			
20 DISTRIBUTION/AVAILABILITY OF ABSTRACT		21 ABSTRACT SECURITY CLASSIFICATION	
<input checked="" type="checkbox"/> UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS RPT <input type="checkbox"/> DTIC USERS		UNCLASSIFIED	
22a NAME OF RESPONSIBLE INDIVIDUAL	22b TELEPHONE (Include Area Code)	22c OFFICE SYMBOL	
Norman F. Lyons	408 646-2666	341b	

Approved for public release; distribution is unlimited.

The Design and Development of a
Management Information System
for the Monterey Navy Flying Club

by

Derek R. George
Commander, United States Navy
B. S., Southern Illinois University, 1967

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

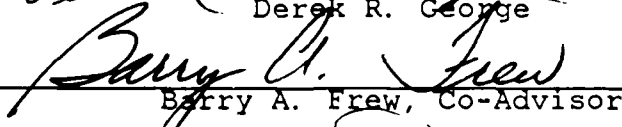
from the

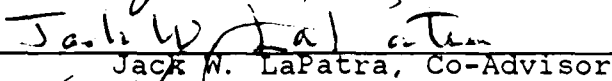
NAVAL POSTGRADUATE SCHOOL
March 1986

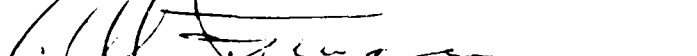
Author:



Derek R. George

Approved by:


Barry A. Frew, Co-Advisor


Jack W. LaPatra, Co-Advisor


Willis R. Greer, Jr., Chairman,
Department of Administrative Sciences


Kneale T. Marshall,
Dean of Information and Policy Sciences

ABSTRACT

This thesis provides a Management Information System for the Monterey Navy Flying Club. It supplies the tools necessary to enable the club manager to maintain all club records and generate required administrative and financial reports.



Accession For	
NTIS CRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

THESIS DISCLAIMER

The reader is cautioned that computer programs developed in this research may not have been exercised for all cases of interest. While every effort has been made, within the time available, to ensure that the programs are free of computational and logic errors, they cannot be considered validated. Any application of these programs without additional verification is at the risk of the user.

TABLE OF CONTENTS

I.	INTRODUCTION	8
	A. ORGANIZATION	8
	B. THESIS BACKGROUND	8
	C. GOALS AND OBJECTIVES	9
	D. METHODOLOGY	10
	E. SYSTEM REQUIREMENTS	11
	F. SYSTEM OVERVIEW	11
II.	DATA FLOW DIAGRAMS	16
III.	USER'S GUIDE	21
	A. INTRODUCTION	21
	B. STARTING THE SYSTEM	21
	C. MENU GUIDE	21
	1. MNFC Main Menu (Figure 1.2)	21
	2. Membership Transactions (Figure 1.3)	22
	3. Financial Transactions (Figure 1.4)	22
	4. Enter A New Transaction (Figure 1.5)	22
	5. Edit/Delete An Existing Transaction (Figure 1.6).	22
	6. Print Reports / Statements (Figure 1.7)	23
	7. Print Administrative Reports (Figure 1.8)	23
	8. Print Financial Statements (Figure 1.9)	23
	9. Print Annual Report Inputs (Figure 1.10).	23
	D. DATE FORMAT INFORMATION	24
	E. LOADING THE DATABASE INITIALLY	24
	F. SYSTEM BACKUP	25
	G. EXITING RBASE 5000	25

IV.	SYSTEM REPORTS AND STATEMENTS	26
V.	SOURCE CODE	33
VI.	TEST DATA	34
VII.	CONCLUSIONS / RECOMMENDATIONS	35
	A. SUMMARY	35
	B. OUTPUT REVIEW	35
	1. Member Statement	35
	2. Instructor Statement	35
	3. Member Roster	35
	4. Dispatcher Report	35
	5. Lessor Statement	35
	6. Aircraft Inventory and Status Report . . .	36
	C. RECOMMENDATION	36
APPENDIX A:	MONTEREY NAVY FLYING CLUB FUNCTIONAL DESCRIPTIONS	37
	1. FIRST LEVEL FUNCTIONAL BREAKDOWN	37
	2. MANAGE CLUB RECORDS FUNCTIONAL BREAKDOWN . . .	40
	3. GENERATE REQUIRED REPORTS FUNCTIONAL BREAKDOWN	47
	4. PROCESS END OF MONTH TRANSACTIONS FUNCTIONAL BREAKDOWN.	55
APPENDIX B:	MNFC DATA DICTIONARY	57
APPENDIX C:	SOURCE CODE	83
	BIBLIOGRAPHY	122
	INITIAL DISTRIBUTION LIST	123

LIST OF FIGURES

1.1	Menu Structure Chart	12
1.2	MNEC Main Menu	13
1.3	Membership Transactions	13
1.4	Financial Transactions	13
1.5	Enter a New Transaction	13
1.6	Edit / Delete An Existing Transaction	14
1.7	Print Reports / Statements	14
1.8	Print Administrative Reports	14
1.9	Print Financial Statements	14
1.10	Print Annual Reports	15
2.1	Manage Navy Flying Club DFD	16
2.2	Manage Navy Flying Club - 1st Level DFD	17
2.3	Manage Club Records DFD	18
2.4	Generate Required Reports DFD	19
2.5	Process End of Month Transactions	20
4.1	Aircraft Inventory and Status Report	27
4.2	Monterey Navy Flying Club Roster	28
4.3	Monterey Navy Flying Club Dispatcher Report	29
4.4	Instructor Statement	30
4.5	Lessor Statement	31
4.6	Member Statement	32
A.1	MNEC Structure Chart	38
A.2	First Level Functional Breakdown	39
A.3	Manage Club Records Functional Breakdown	40
A.4	Generate Required Reports Functional Breakdown	47
A.5	Process End Of Month Transactions Functional Breakdown	55
B.1	Data Relationship Chart	58

I. INTRODUCTION

A. ORGANIZATION

The Monterey Navy Flying Club is a non-profit organization sponsored by the Naval Post Graduate School, Monterey, Ca. It exists to provide low cost, safe, light aircraft operations for its members. The Superintendent, Naval Postgraduate School, is directly responsible for ensuring strict adherence to FAA regulations and flight procedures. An elected Board of Directors, through the Club Manager, is responsible to the Commanding Officer for the implementation of his directives.

The Club Manager conducts routine club business within the guidelines of applicable DOD, USN, FAA, and NTSB directives and regulations. A major responsibility is maintaining accurate financial records of club activities. This data is used to prepare the annual financial report for submission to NMPC at the completion of each fiscal year. An annual audit is conducted each year for inclusion in the annual report. Specific areas of the audit include:

- Organization, management and administration
- Aircraft, equipment and supplies
- Operations, training and standardization
- Safety and aircraft maintenance
- Financial management

B. THESIS BACKGROUND

The Monterey Navy Flying Club has one of the largest memberships of the Navy sponsored flying clubs. As a result of this large membership and the amount of manual paperwork required to properly maintain club records, the Manager's ability to provide necessary services and reports is severely hampered. The implementation of

an efficient Management Information System will provide the resources necessary to track the statistics and financial data mandated by current directives and regulations.

C. GOALS AND OBJECTIVES

The primary goal of this system is to provide a functioning Management Information System for the MNFC Manager's use. It should provide substantial time savings and significantly reduce the manual duplication presently required. The structure of this system is designed to accomodate full financial and administrative support. Of the final output listed below, The first five items have been included in this thesis. Ultimate output will include:

- Member Roster
- Dispatcher Report
- Instructor Statement
- Member Statement
- Lessor Statement
- Income Statement
- Balance Sheet
- Input Items For The Budget Proposal
- Manager's Report (monthly)
- A/C Inventory and Status Report (monthly)
- Aged Accounts Receivables
- Input Items For Appendix A of Annual Report
- Statement of Net Worth
- Membership Breakdown

All necessary data stores (tables) and logic to produce the final system have been incorporated in the thesis. The Monterey Navy Flying Club does not presently maintain individual product inventories. However, an appropriate table containing product name, company, part number, unit price, number on hand, reorder point, date of last update, etc, with the necessary command file could be added in the future.

D. METHODOLOGY

Function descriptions (Appendix A) and data descriptions (Appendix B) will be combined to develop the data flow diagrams of Chapter IV.

The resulting logical system specification is converted into a physical system implementation which may be carried out in a phased manner.

Management of the Monterey Navy Flying Club can be functionally divided into two areas:

- MANAGE THE CLUB'S RECORDS
- GENERATE THE REQUIRED REPORTS

Hierarchical function charts for the two major functions are shown in Chapter II. Additionally, each of the functions utilized by the system are described in detail. This description includes a structure chart of the function and its next level of detail (abstraction), a unique numerical identifier, and a detailed description of the function. The lowest level function includes a listing of the data used, and their source and destination.

The data descriptions in Appendix A consist of a data dictionary containing the entity name, a detailed description of the entity and a list of the elements that make up that entity. Each element is further defined by type, field size, and a brief description if necessary. The following data entities are included:

- AIRCRAFT HOURS TABLE
- AIRCRAFT MAINTENANCE TABLE
- AIRCRAFT RECORD TABLE
- ACCOUNTS PAYABLE TABLE
- ACCOUNTS RECEIVABLE TABLE
- BALANCE SHEET TABLE
- BILLTEMP TABLE
- CAPITAL ASSET TABLE
- CASH ACCOUNTS TABLE
- CHARGE HISTORY TABLE

- FLIGHT RECORD TABLE
- HOUR HISTORY TABLE
- INSTRUCTOR HISTORY TABLE
- INSTRUCTOR RECORD TABLE
- LESSOR ACCOUNTS TABLE
- MAINTENANCE HISTORY TABLE
- MEMBER CHARGE TABLE
- MEMBER FLIGHT TABLE
- MEMBER PAYMENT TABLE
- MEMBER RECORD TABLE
- PAYMENT HISTORY TABLE
- TEMPORARY HOURS TABLE

E. SYSTEM REQUIREMENTS

A prototype Management Information System for the Monterey Navy Flying Club was developed using the R:base 5000 Database Management System. The R:base 5000 program is a product of Microrim, Inc., Bellview, Washington and requires the following to operate properly:

- IBM PC, PC-XT, PC-AT, or 100% compatible microcomputer
- DOS, version 2.0 or higher
- 237 K of main memory after system configuration (a minimum of 320 K is recommended)
- Either a color or monochrome monitor
- Either a hard disk drive and one double-sided, double-density 5.25 inch floppy disk drive, or 2 double-sided, double-density 5.25 inch floppy disk drives.

This system is designed to utilize a hard disk. Processing speed can be increased considerably if 640 K of main memory is available.

F. SYSTEM OVERVIEW

The basic user interface for this system is via a series of menus set up to allow the user to select the type of system interaction desired. The initial menu will allow the user to select membership or financial transactions,

print certain reports and records, or generate end of month transactions. Each selection will produce another menu. The menu choices are shown below in Figures 1.1 - 1.10. Chapter V provides detailed instructions for and explanations of the choices available for the various functions.

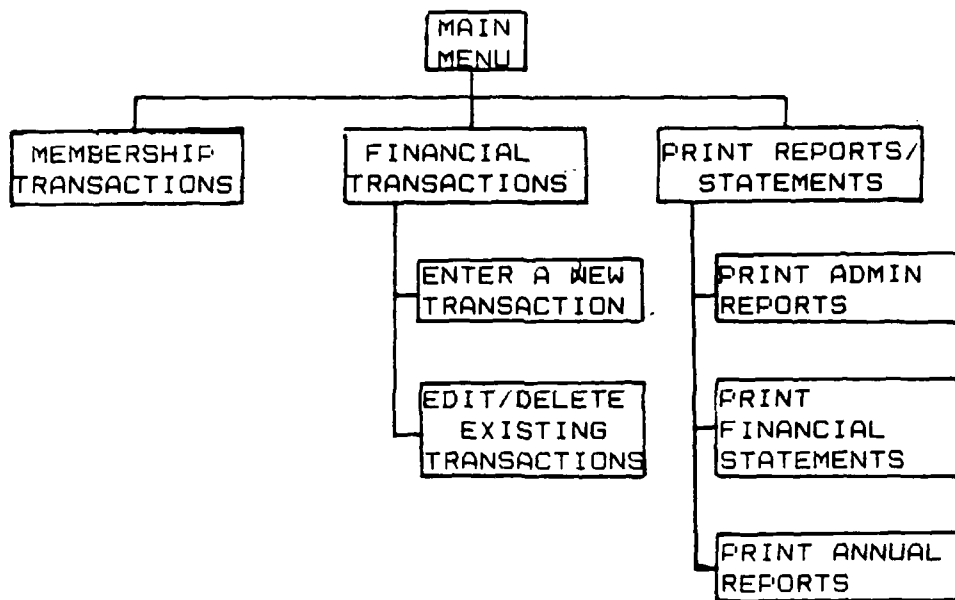


Figure 1.1 Menu Structure Chart.

Additional input/query interaction with the system is available via the R:base 5000 language.

The system will automatically credit or debit the applicable accounts, and update appropriate fields (transparent to the user).

- MNFC MAIN MENU
- (1) MEMBERSHIP TRANSACTIONS
 - (2) FINANCIAL TRANSACTIONS
 - (3) PRINT REPORTS / STATEMENTS
 - (4) GENERATE END OF MONTH TRANSACTIONS
 - (5) BACK UP AND PACK THE DATABASE
 - (6) RETURN TO SYSTEM DOS

Figure 1.2 MNFC Main Menu.

- MEMBERSHIP TRANSACTIONS
- (1) ENTER A NEW MEMBER RECORD
 - (2) EDIT / DELETE EXISTING MEMBER RECORD
 - (3) RETURN TO MAIN MENU

Figure 1.3 Membership Transactions.

- FINANCIAL TRANSACTIONS
- (1) ENTER A NEW TRANSACTION
 - (2) EDIT / DELETE AN EXISTING TRANSACTION
 - (3) RETURN TO MAIN MENU

Figure 1.4 Financial Transactions.

- ENTER A NEW TRANSACTION
- (1) ACCOUNTS RECEIVABLE
 - (2) ACCOUNTS PAYABLE
 - (3) ENTER A MEMBER'S CHARGES
 - (4) ENTER A MEMBER'S PAYMENTS
 - (5) ENTER A MAINTENANCE TRANSACTION
 - (6) RETURN TO FINANCIAL TRANSACTIONS MENU

Figure 1.5 Enter a New Transaction.

EDIT / DELETE AN EXISTING TRANSACTION

- (1) ACCOUNTS RECEIVABLE
- (2) ACCOUNTS PAYABLE
- (3) MEMBER CHARGE TRANSACTIONS
- (4) MEMBER PAYMENT TRANSACTION
- (5) RETURN TO FINANCIAL TRANSACTIONS MENU

Figure 1.6 Edit / Delete An Existing Transaction.

PRINT REPORTS / STATEMENTS

- (1) PRINT ADMINISTRATIVE REPORTS
- (2) PRINT FINANCIAL STATEMENTS
- (3) PRINT ANNUAL REPORTS
- (4) RETURN TO MAIN MENU

Figure 1.7 Print Reports / Statements.

PRINT ADMINISTRATIVE REPORTS

- (1) PRINT MEMBER ROSTER
- (2) PRINT DISPATCHER REPORT
- (3) PRINT MANAGER'S REPORT
- (4) RETURN TO PRINT REPORTS/STATEMENTS MENU

Figure 1.8 Print Administrative Reports.

PRINT FINANCIAL STATEMENTS

- (1) PRINT MEMBER STATEMENTS
- (2) PRINT INSTRUCTOR STATEMENTS
- (3) PRINT LESSOR STATEMENTS AND AIRCRAFT INVENTORY/STATUS
- (4) PRINT AGED ACCOUNTS RECEIVABLE
- (5) PRINT BALANCE SHEET INPUTS
- (6) PRINT INCOME STATEMENT INPUTS
- (7) RETURN TO PRINT REPORTS/STATEMENTS MENU

Figure 1.9 Print Financial Statements.

PRINT ANNUAL REPORTS

- (1) PRINT MEMBERSHIP BREAKDOWN
- (2) PRINT STATEMENT OF NET WORTH DATA
- (3) PRINT DATA FOR APPENDIX A OF ANNUAL REPORT
- (4) RETURN TO PRINT REPORTS/STATEMENTS MENU

Figure 1.10 Print Annual Reports.

II. DATA FLOW DIAGRAMS

The data flow depicted in Figure 2.1 through Figure 2.5 is the result of combining the function descriptions from Appendix A and the data descriptions from Appendix B.

As shown, the flow through the system is initiated and controlled by a series of menu choices. These were discussed in Chapter I and depicted in Figures 1.1 through 1.10.

To ensure full availability of DBMS advantages, the user also has the option of addressing the system in a direct query/command mode for ad hoc query requirements.

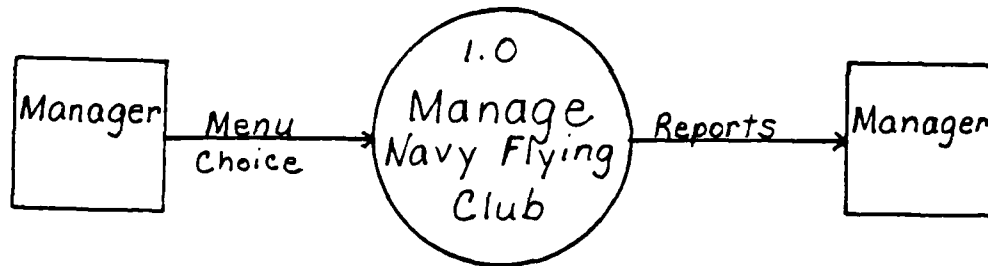


Figure 2.1 Manage Navy Flying Club DFD.

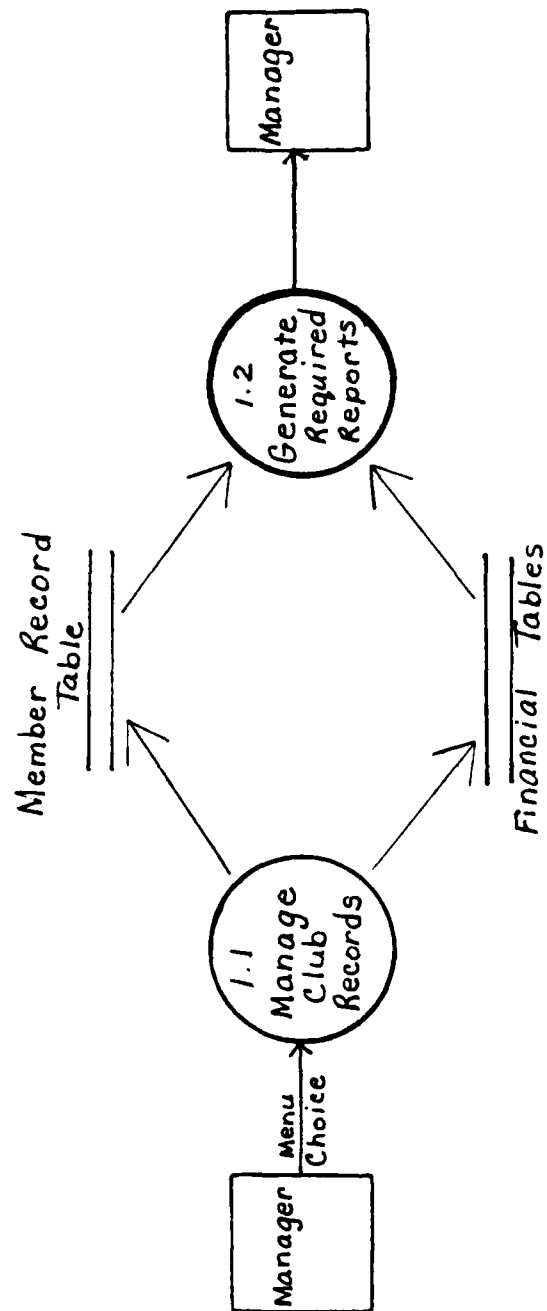


Figure 2.2 Manage Navy Flying Club - 1st Level DFD.

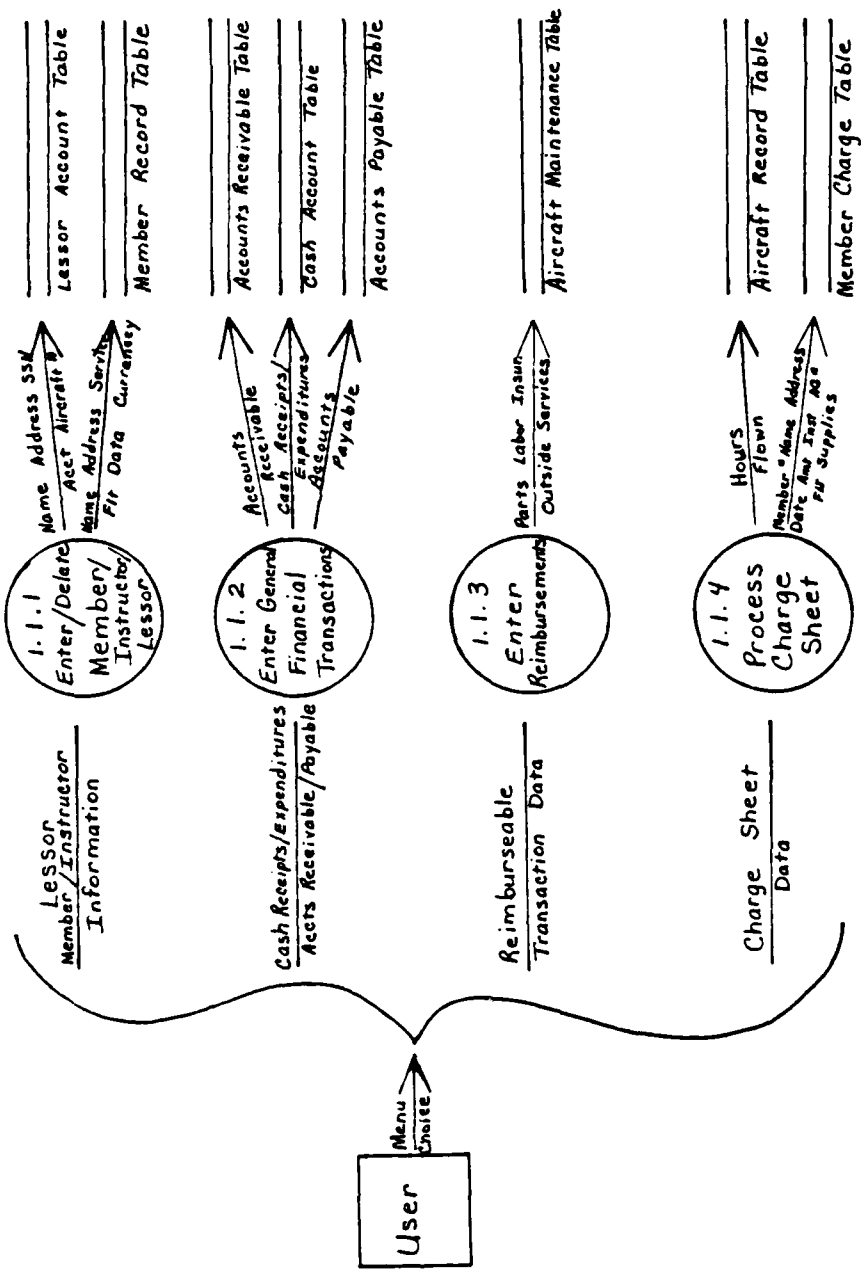


Figure 2.3 Manage Club Records DFD.

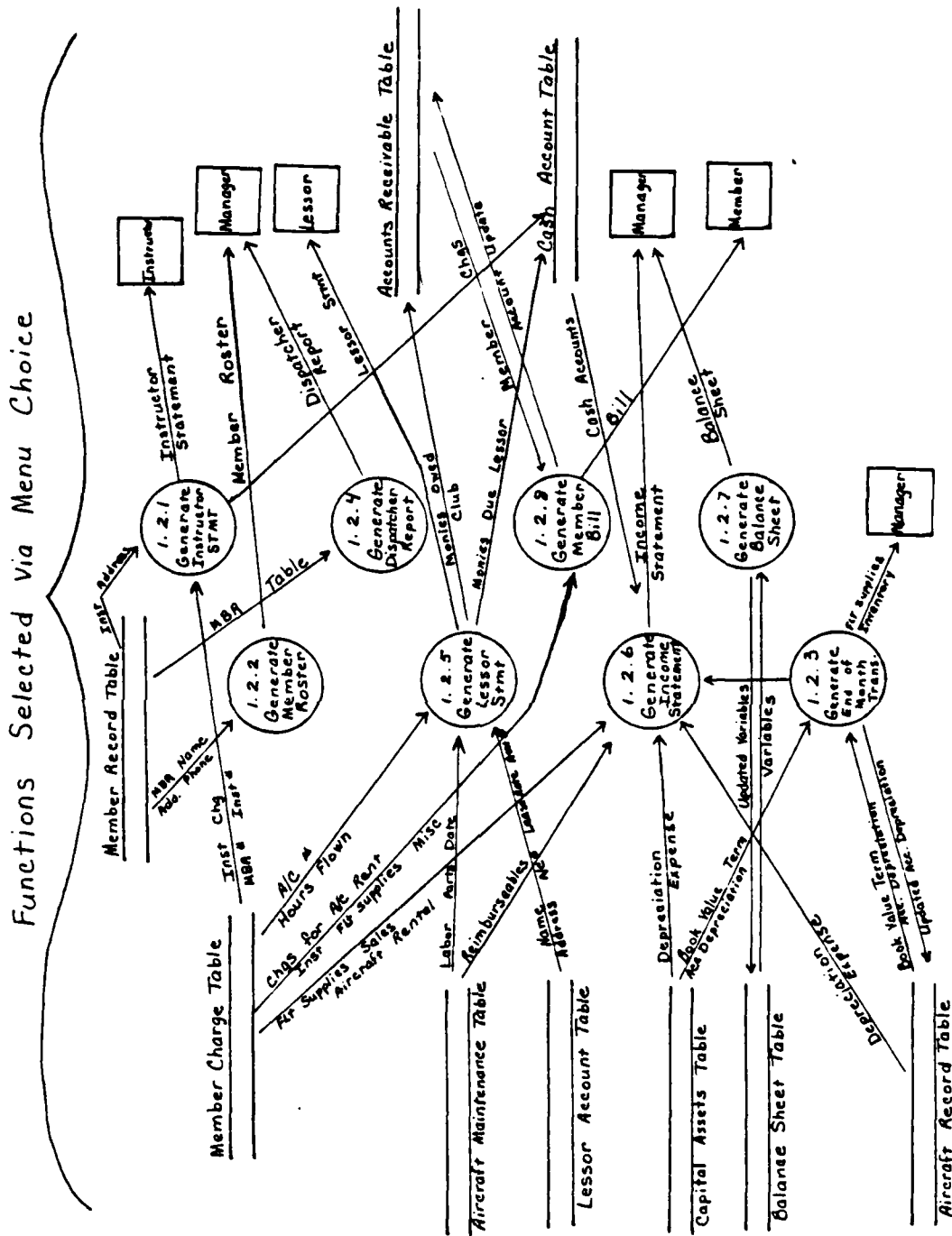


Figure 2.4 Generate Required Reports DFD.

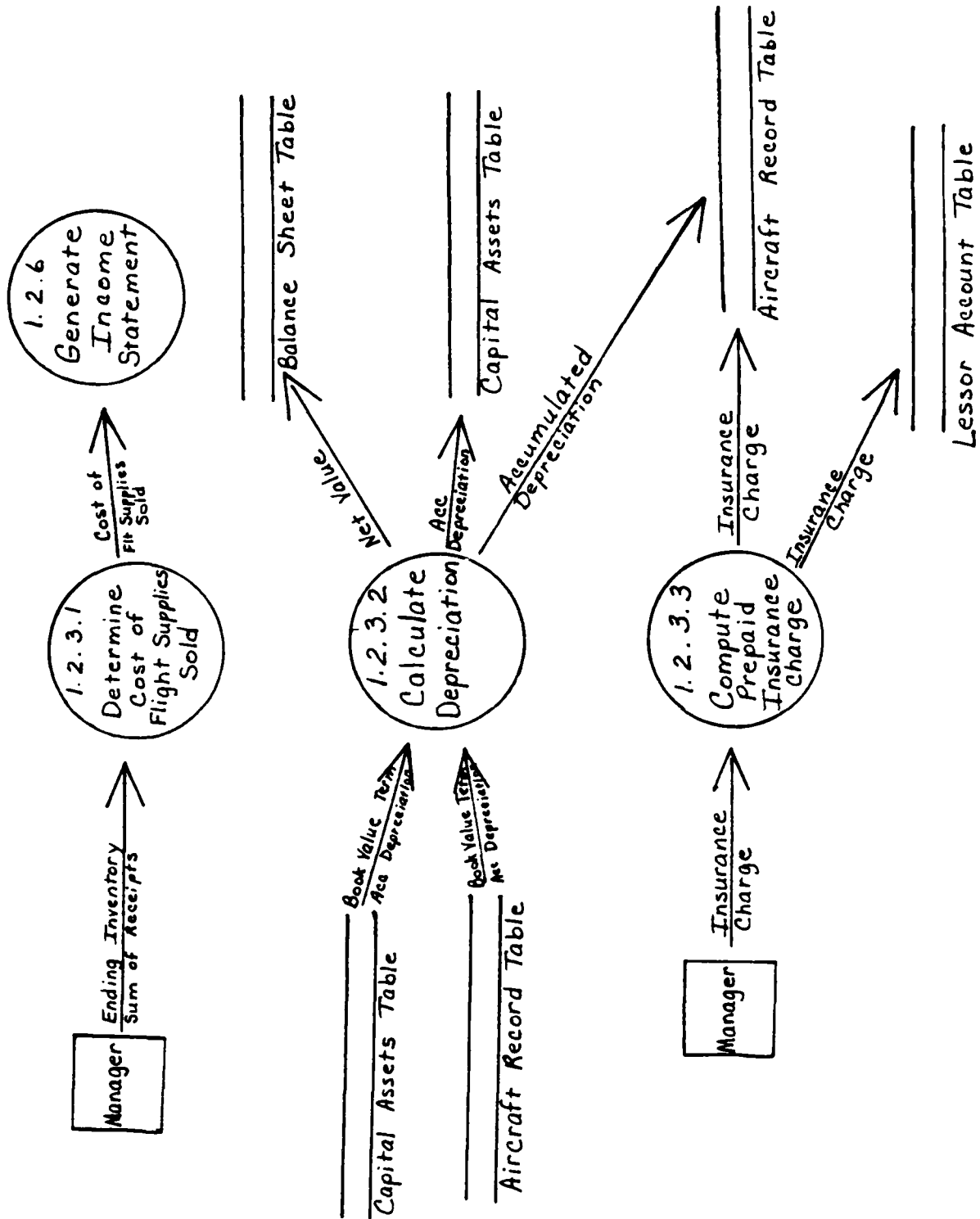


Figure 2.5 Process End of Month Transactions.

III. USER'S GUIDE

A. INTRODUCTION

The Monterey Navy Flying Club Management Information System is an integrated MIS designed to provide the club manager with an efficient and effective tool with which to manage both the administrative and financial records of the club. The following special topics are presented to facilitate introduction to this system:

- STARTING THE SYSTEM
- MENU GUIDE
- DATE FORMAT
- LOADING THE DATABASE INITIALLY
- SYSTEM BACKUP
- EXITING R:base 5000

B. STARTING THE SYSTEM

Once the system is on line, the proper subdirectory selected, and the C> is displayed, type RB5000 and press [ENTER]. Move the cursor to RBASE and press [ENTER]. when the RBASE menu appears, select #1 for the R> prompt. At this time, type RUN CLUB_MIS IN CLUB_MIS.APX and press [ENTER]. This selects the Monterey Navy Flying Club Management information system and displays the MNFC MAIN MENU (Figure 1.2).

C. MENU GUIDE

1. MNEC Main Menu (Figure 1.2)

Selecting option 1, MEMBERSHIP TRANSACTIONS, from the main menu displays the MEMBERSHIP TRANSACTIONS menu (Figure 1.3). Option 2 presents the FINANCIAL TRANSACTIONS menu (Figure 1.4). Option 3 displays the PRINT REPORTS / STATEMENTS menu (Figure 1.8). Option 4 computes four end

of month transactions for the BALANCE SHEET and the INCOME STATEMENT. Option 5 backs up and repacks the database to recover unusable disk space. Option 6 returns the user to system DOS.

2. Membership Transactions (Figure 1.3)

This menu provides the user with 3 options; ENTER A NEW MEMBER RECORD, EDIT/DELETE AN EXISTING MEMBER RECORD, and RETURN TO MAIN MENU. Both of the first two options display an interactive form and either requests new member information or a member number for the record to be edited / deleted. The third option returns the user to the MNFC MAIN MENU.

3. Financial Transactions (Figure 1.4)

If option 1 is selected, the user is provided with the ENTER A NEW TRANSACTION menu (Figure 1.5). Option 2 provides the EDIT/DELETE AN EXISTING TRANSACTION menu (Figure 1.6). Option 3 will return the user to the MNFC MAIN MENU.

4. Enter A New Transaction (Figure 1.5)

This menu allows the user to select what kind of transaction to enter via 4 options with a fifth option to return to the FINANCIAL TRANSACTIONS MENU. Option 1, ACCOUNTS RECEIVABLE, enters data into the ACCOUNTS RECEIVABLE TABLE. Option 2, ACCOUNTS PAYABLE, enters data into the ACCOUNTS PAYABLE TABLE. Option 3, MEMBER CHARGE TRANSACTIONS, processes members' charge sheets as input by the club manager. Option 4, MEMBER PAYMENT TRANSACTIONS, processes member payments as input by the club manager.

5. Edit/Delete An Existing Transaction (Figure 1.6).

Option 1, ACCOUNTS RECEIVABLE, edits/deletes data into the ACCOUNTS RECEIVABLE TABLE. Option 2, ACCOUNTS PAYABLE, edits/deletes data into the ACCOUNTS PAYABLE TABLE. Option 3, MEMBER CHARGE TRANSACTIONS, edits/deletes

members' charge sheets. Option 4, MEMBER PAYMENT TRANSACTIONS, edits/deletes member payments. All options display an interactive form. To enter new transaction information or member information, the number for the record/account plus the invoice number for the particular transaction desired is requested. Option 5 returns the user to the FINANCIAL TRANSACTIONS menu.

6. Print Reports / Statements (Figure 1.7)

Option 1 displays the PRINT ADMINISTRATIVE REPORTS menu (Figure 1.8). Option 2 displays the PRINT FINANCIAL STATEMENTS menu (Figure 1.9). Option 3 displays the PRINT ANNUAL REPORTS INPUT menu (Figure 1.10). Option 4 returns the user to the MNFC MAIN MENU.

7. Print Administrative Reports (Figure 1.8)

This menu allows the user to print 4 reports. Option 1 produces the Member Roster. Option 2 produces the Dispatcher Report, and option 3 prints the monthly Manager's Report. Option 4 returns the user to the PRINT REPORTS/STATEMENTS menu.

8. Print Financial Statements (Figure 1.9)

This menu contains 7 options. The first 6 produce Member Statements, Instructor Statements, Lessor Statements/Aircraft Inventory and Status Report, Aged Accounts Receivable Report, Balance Sheet Inputs, and Income Statement Inputs. The last option returns the user to the PRINT REPORTS/STATEMENTS menu.

9. Print Annual Report Inputs (Figure 1.10)

Option 1 produces the required annual Membership Breakdown. Option 2 prints data to be used in developing the Statement of Net Worth. Option 3 produces data for inclusion in the Annual Report. Option 4 returns the user to the PRINT REPORTS/ STATEMENTS menu.

As a relational database language, RBASE 5000 is efficient and powerful. The menus address the routine

and repetitive tasks involved in managing the MNEC. Additional items of information are readily available through the extensive system of tables provided and the normal RBASE 5000 ommands. For example, to print an alphabetized list of members qualified to operate the C-150 aircraft, the user would enter the following two commands:

```
OUTPUT PRINTER
```

```
SELECT L:NAME F:NAME MEM_NUM FROM MEM_REC SORTED  
BY + L:NAME WHERE CUR_MODS CONTAINS C_150
```

D. DATE FORMAT INFORMATION

The date function can be set to several basic formats: i.e., day-month-year, day-year-month, month-day-year, month-year-day, year-month-day, or year-day-month. The day must be expressed as two digits. The month can be expressed as either two digits or a three letter abbreviation. The year can be expressed as either two or four digits. The RBASE default format is MM/DD/YY. Once a format is set, RBASE only accepts dates entered in that format. If a month or day is only one digit, it is not necessary to enter a zero before the digit.

E. LOADING THE DATABASE INITIALLY

The system is designed to input data via the menus in most instances. However, before using the menu system, certain of the tables in the database must be loaded directly, i.e., initial information about each aircraft and lessor data (name, address, etc.). To enter aircraft information, the user enters the following command at the " R> " prompt: OPEN FLYCLUB followed by: ENTER A/C_FORM The user is then shown a form that requests the necessary information about the aircraft.

To enter lessor data, the user enters the following command at the " R> " prompt: ENTER LES_FORM Once

again he is prompted to enter the necessary information about the individual lessors and their aircraft identification.

All remaining data is entered via menus within the normal MNFC MIS. To start the system running, enter the following command at the " R> " prompt: RUN CLUB_MIS IN CLUB_MIS.APX

F. SYSTEM BACKUP

System backup and "packing" the database is accomplished via option 5 on the Main Menu, BACK UP AND PACK THE DATABASE. As data is entered and removed from various tables, unusable disk space accumulates. To alleviate this situation, the user "repacks" the database. Menu prompts direct the user through the necessary steps to obtain a floppy disk backup copy of the database and complete a "repack." It is recommended that this be completed at least once weekly.

G. EXITING RBASE 5000

To exit Rbase, the user selects the final option of the current menu to return to the Main Menu. From there he may exit to DOS.

IV. SYSTEM REPORTS AND STATEMENTS

The following monthly reports and statements are included in the system and are selected by the menu choices indicated:

REPORT / STATEMENT	MENU SELECTION / FIGURE #
AIRCRAFT INVENTORY/STATUS	SELECTION 3 / FIGURE 1.9
MEMBER ROSTER	SELECTION 1 / FIGURE 1.8
DISPATCHER REPORT	SELECTION 2 / FIGURE 1.8
INSTRUCTOR STATEMENT	SELECTION 2 / FIGURE 1.9
LESSOR STATEMENTS	SELECTION 3 / FIGURE 1.9
MEMBER STATEMENTS	SELECTION 1 / FIGURE 1.9

Figure 4.1 - Figure 4.6 are examples of the above output.

MONTEREY NAVY FLYING CLUB
 MONTHLY AIRCRAFT STATUS AND FLIGHT TIME REPORT
 FOR PERIOD ENDING: 03/05/86

A/C NUM	TYPE	ENDTIME	START	GROSS	MAINT/R	BILLED	RENT	RENT TOT	LEASE	LEASE TOT	MAINT CST	INSUR COST	OTHER	NET TO OWNER	FUEL CST	NET TO CLUB
N15631	P-100	140.00	100.00	40.00	5.000	35.00	30.00	1,050.00	15.00	525.00	420.00	60.00				
N7322A	C-150	100.00	70.000	30.00	0.000	30.00	23.00	690.00	11.00	330.00	55.00	60.00				
N0047Y	AZTEC	120.00	100.00	20.00	0.000	20.00	65.00	1,300.00		0.00		75.00				

Figure 4.1 Aircraft Inventory and Status Report.

MONTEREY NAVY FLYING CLUB ROSTER

ROSTER DATE: 04/16/86 TOTAL MEMBERS: 4 INST NO. PHONE PAGE: 1

NAME ADDRESS	MEMBER NO.	INST NO.	PHONE
CLIFT, MICHAEL R. 4538 BROWNELL	3		(408) 646-0732
GEORGE, DEREK R. 1284 OAKGLEN WAY	1	1	(408) 649-0992
RODECK, BRIAN D. 830 SUNSET DR. #J	4	4	(408) 375-7442
RODECK, RENEE R. 830 SUNSET DR. #J	2	2	(408) 375-7442

Figure 4.2 Monterey Navy Flying Club Roster.

MONTEREY NAVY FLYING CLUB
DISPATCHERS REPORT

DATE: 04/16/86

PAGE: 1

NAME MEM NO. INST NO. FLT HRS-BIENNIAL MED DATE/CLASS FAA CERT/CLASS COLLATERAL DUTY
LAST DUAL--LAST FLIGHT/AIRCRAFT-DAY LNDSG-NITE TIME/LAST-NITE LND-PREC APP-NON PRC-INST TIME-AIRCRAFT MODELS CURRENT IN

CLIFT, MICHAEL R. - 3 - - - - 09/20/85/III - STUD/SINGLE - TREASURER
- - - - - / - - - - 0 - - 0 - - C-152

GEORGE, DEREK R. - 1 - 1 - 8.0 - 08/12/85 - 09/30/85/I - COML/SINGLE, MULTI, INST, LAND - PRESIDENT
03/03/86 - 03/04/86 - N15631 - 0 - - 0 - - /03/04/86 - 0 - - 0 - - T34, P-3, SR-71

RODECK, BRIAN D. - 4 - 4 - - 05/13/84 - 01/01/84/II - PRIV/HELD, SINGLE, LAND - SUPPLY
- - - - - / - - - - 0 - - 0 - - UH-1, C-172

RODECK, RENEE R. - 2 - 2 - - 02/04/86 - 01/06/86/I - CFI /SINGLE, LAND, INST - SECRETARY
- - - - - / - - - - 0 - - 0 - - C-180, BEECH-95, C-177RG

Figure 4.3 Monterey Navy Flying Club Dispatcher Report.

MONTEREY NAVY FLYING CLUB
806 AIRPORT ROAD
MONTEREY, CA 93940
(408) 372-7033

STATEMENT OF INSTRUCTION PROVIDED

INSTRUCTOR NO.: 1 STATEMENT FOR PERIOD ENDING: 02/28/86

DEREK GEORGE
1284 OAKGLEN WAY
MONTEREY CA 93943

DATE	STUDENT	AIRCRAFT	HOURS FLOWN	AMOUNT
02/03/86	CLIFT	N8047Y	1.000	20.00
02/04/86	RODECK	N7322A	2.000	25.00

TOTAL DUE INSTRUCTOR: \$45.00

Figure 4.4 Instructor Statement.

MONTEREY NAVY FLYING CLUB
 806 AIRPORT ROAD
 MONTEREY, CA 93940
 (408) 372-7033

STATEMENT OF LEASED AIRCRAFT OPERATION

AIRCRAFT NO : N7322A	HOBBS METER READINGS:
FOR THE PERIOD ENDING: 03/05/86	ENDING: 100.00
	BEGINNING: 70.00
JONES, JAYNE J.	ELAPSED: 30.00
21 SOUTH ELM	LESS MAINT TIME: 0.00
SEASIDE CA 93953	NET HOURS FLOWN: 30.00
	LESS OWNER TIME: 1.00
	TOTAL LEASED TIME: 29.00

COMPUTED LEASE AMOUNT: 29.00 HRS x \$11.00 /HR. = \$319.00

MAINTENANCE PERFORMED

DATE	LABOR CHARGES	PARTS CHARGES
03/02/86	\$10.00	\$5.00
03/01/86	\$40.00	\$0.00
		TOTAL MAINT: \$55.00
		SUB TOTAL: \$264.00

OTHER CHARGES AND CREDITS

TOTAL OTHER: -----

TOTAL DUE OWNER: -----

MNFC CHECK # _____ ENCLOSED IN THE AMOUNT OF \$ _____ DUE TO OWNER.

THANK YOU,

MNFC MANAGER

Figure 4.5 Lessor Statement.

V. SOURCE CODE

Appendix C contains the program code generated by both this thesis writer and the application generator within the Rbase 5000 language in the process of developing the system menus, tables, etc. The portions of the code generated through the Application Express do not contain comments. As any changes input to the current application cause a complete rewrite of this code, it is not considered cost effective to attempt to include comments. However, one is able to discern how the other programs, reports, forms, etc. are incorporated and how to go about changing / updating them by reviewing the system through the Express program.

VI. TEST DATA

The following eight data stores (tables) were loaded with data to test the system and to generate example reports and statements:

- Aircraft Hours
- Aircraft Maintenance
- Aircraft Record
- Accounts Receivable
- Lessor Accounts
- Member Charge
- Member Payment
- Member Record

A listing of the data that was input is contained in the database files.

VII. CONCLUSIONS / RECOMMENDATIONS

A. SUMMARY

This thesis has developed a prototype Management Information System for the Monterey Navy Flying Club. Rbase:5000, by Microrim of Bellevue, Washington, was used as the database language for this product.

The system is interactive and menu-driven and consists of 22 data stores (tables), 161 separate columns and currently 8 command programs.

B. OUTPUT REVIEW

Present output consists of the following administrative and financial reports / statements.

1. Member Statement

This statement is a compilation of all charges and payments that have transpired during the current billing period. In conjunction with posting appropriate monetary charges, additional data is recorded to produce additional output.

2. Instructor Statement

Includes all flights for each instructor during the billing period and lists the date, aircraft, flight time, student, charge, etc.

3. Member Roster

Includes a listing of all club members for a quick crosscheck of member number, address, etc.

4. Dispatcher Report

List each member and his/her currency in numerous areas.

5. Lessor Statement

Produces a listing of what the club owes / is owed in connection with leased aircraft.

6. Aircraft Inventory and Status Report

Includes costs and flight time comparisons of all club aircraft, both owned and leased.

C. RECOMMENDATION

The structure of this system was based from the beginning on the expectations of a final product that included full financial and administrative support for the MNFC manager. Although the final product was beyond the scope of this thesis, every effort has been made to create a program that will readily accept follow on work to provide for expansion and implementation of additional financial functions.

It is recommended that a phased implementation be utilized and that follow on thesis work be assigned to completed the remaining financial portions i.e. Balance Sheet and Income Statement. Additional areas to be developed that have been provided for in the initial structure include the appendices for the Annual Report.

APPENDIX A

MONTEREY NAVY FLYING CLUB FUNCTIONAL DESCRIPTIONS

Figure A.1 - Figure A.5 depicting the functional breakdowns of the Monterey Navy Flying Club, as well as individual function descriptions are contained in this appendix.

1. FIRST LEVEL FUNCTIONAL BREAKDOWN

ENTITY NAME : Manage Monterey Navy Flying Club
NUMERICAL IDENTIFIER : 1.0
DESCRIPTION : This module manages all club records, assimilates external inputs and generates reports required to manage financial accounts, maintenance functions, membership data, aircraft data, internal reports to the Board of Directors and the Superintendent, as well as reports required by external activities.

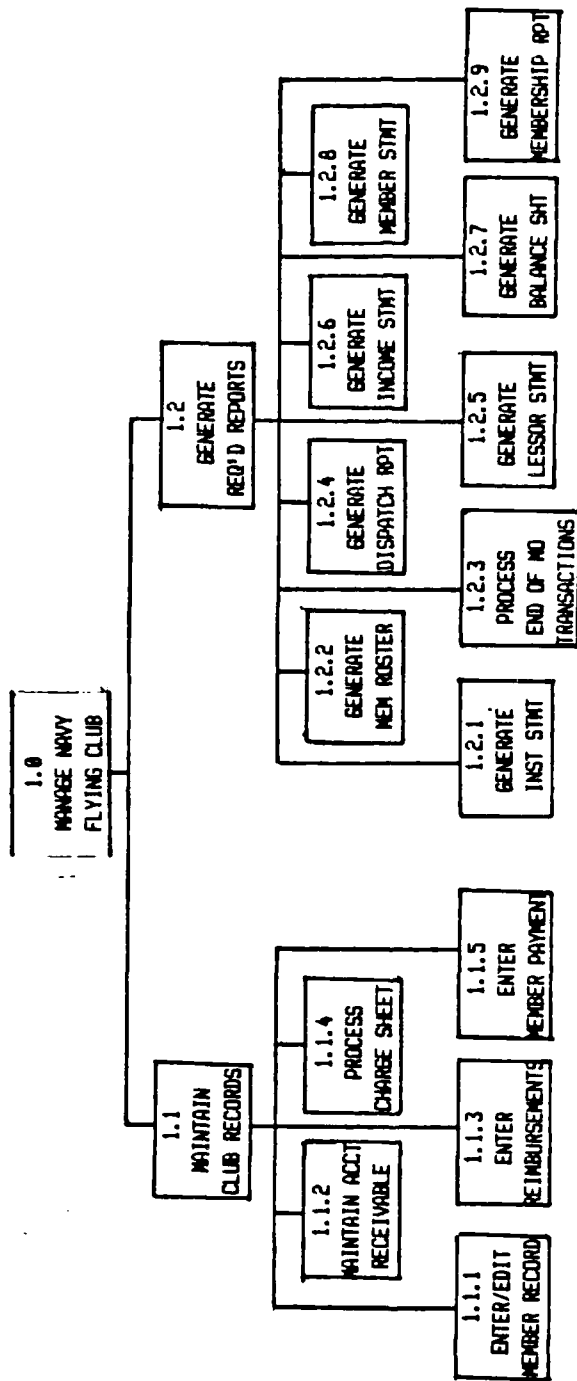


Figure A.1 MNEC Structure Chart.

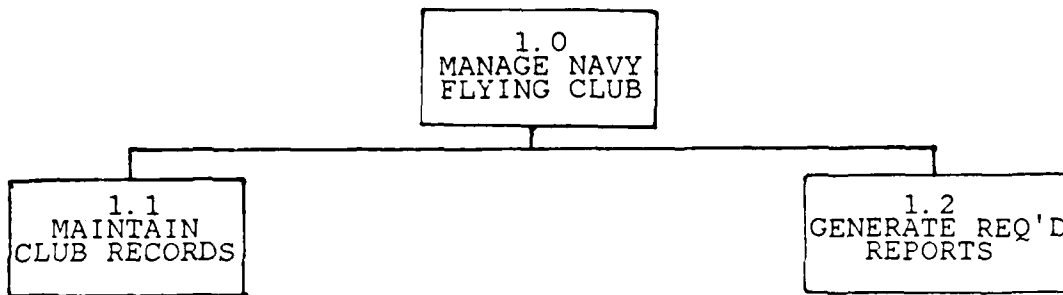


Figure A.2 First Level Functional Breakdown.

FUNCTION NAME : Manage Club Records
 NUMERICAL IDENTIFIER : 1.1
 DESCRIPTION : This module processes charge sheets, maintains member and aircraft records, accounts receivable, and aircraft transaction data, and maintains flight supplies data. Output is provided to various destinations within the system.

FUNCTION NAME : Generate Required Reports
 NUMERICAL IDENTIFIER : 1.2
 DESCRIPTION : This module retrieves data from the supporting data stores and tables within the system to produce reports requested by the manager. Desired reports are selected via menu options.

2. MANAGE CLUB RECORDS FUNCTIONAL BREAKDOWN

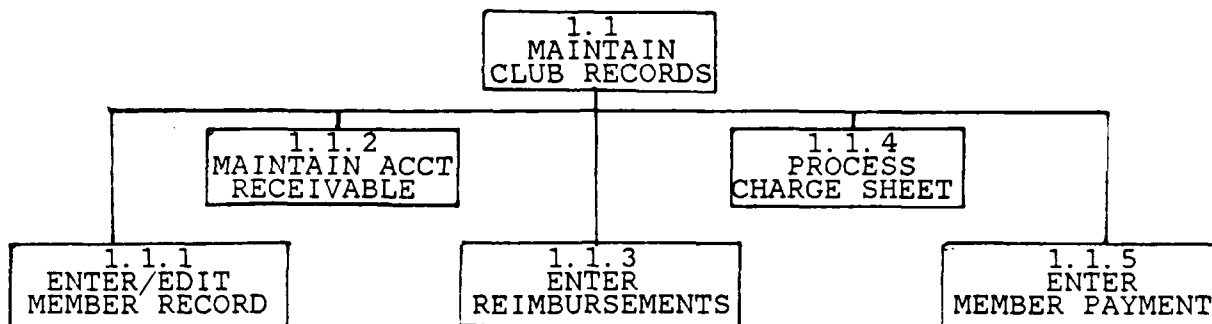


Figure A.3 Manage Club Records Functional Breakdown.

FUNCTION NAME : ENTER/DELETE NEW MEMBER/INSTRUCTOR

NUMERICAL IDENTIFIER : 1.1.1

DESCRIPTION : This module retrieves data about new members or instructors (entered by the club manager) and enters them in the MEMBER RECORD TABLE. If there is no source specified below, information is entered/updated from the MEMBER FORM, addressed via the Edit /delete Member Record option (figure 1.3). The last date flown and the last aircraft model flown, as well as all landings, approaches, and instrument information, are updated by the PROCESS CHARGE SHEET module. The destination for all data listed below is the MEMBER RECORD TABLE unless listed differently under SOURCE/DESTINATION.

DATA SOURCE (s) / DESTINATION (d)

- member number
- instructor number
- last name
- first name
- middle initial
- SSN
- street
- city
- state
- zipcode
- phone
- category (MO,NO,ME,etc)
- initiation fee (Y/N)
- key deposit (Y/N)
- current models (152,172,T34,etc.)
- collateral duty
- biennial flight check date (MM/DD/YY)
- medical cert. date (MM/DD/YY)
- medical classification
- FAA cert (Stud,CFI,COML,etc.)
- FAA classification
(single,multi,inst,etc.)
- record of previous mishap (Y/N)

FUNCTION NAME : Maintain Accounts Receivable

NUMERICAL IDENTIFIER : 1.1.2

DESCRIPTION : This module receives external input as well as input from other modules within the system to keep ACCOUNTS RECEIVABLE TABLE current. Destination is the ACCOUNTS RECEIVABLE TABLE; sources are listed below:

DATA	SOURCE (s) / DESTINATION (d)
- who/what	(s) club manager input
- transaction date	(s) club manager input
- amount	(s) club manager input
- account number	(s) club manager input
- member number	(s) Generate Member Stmt module
- current balance	(s) Generate Member Stmt module
- balance past due 30 days	(s) Generate Member Stmt module
- balance past due 60 days	(s) Generate Member Stmt module
- balance past due 90 days	(s) Generate Member Stmt module
- bal brought fwd - 30	(s) Generate Member Stmt module
- bal brought fwd - 60	(s) Generate Member Stmt module
- bal brought fwd - 90	(s) Generate Member Stmt module

FUNCTION NAME : Enter Reimbursable Transactions

NUMERICAL IDENTIFIER : 1.1.3

DESCRIPTION : This module receives data (input by the club manager) for leased aircraft and updates both the AIRCRAFT MAINTENANCE TABLE and the AIRCRAFT RECORD TABLE.

DATA	SOURCE (s) / DESTINATION (d)
- prepaid insurance	(s) club manager input (d) AIRCRAFT RECORD TABLE
- aircraft number	(s) club manager input (d) AIRCRAFT MAINTENANCE TABLE
- invoice number	(s) club manager input (d) AIRCRAFT MAINTENANCE TABLE
- labor hours	(s) club manager input (d) AIRCRAFT MAINTENANCE TABLE
- labor charge	(s) club manager input (d) AIRCRAFT MAINTENANCE TABLE
- parts charge	(s) club manager input (d) AIRCRAFT MAINTENANCE TABLE
- date of maintenance	(s) club manager input (d) AIRCRAFT MAINTENANCE TABLE
- maintenance source	(s) club manager input (d) AIRCRAFT MAINTENANCE TABLE

FUNCTION NAME : Process Charge Sheet

NUMERICAL IDENTIFIER : 1.1.4

DESCRIPTION : This module receives external input i.e. data from individual charge sheets, and routes the data to the appropriate module/data store for processing and/or storage. The source for the data listed below is the Charge Sheet and the destination is temporary storage for use later to generate the instructor statements, and member statements.

DATA	SOURCE (s) / DESTINATION (d)
- member number	
- member name	(s) MEMBER RECORD TABLE (d) MEMBER CHARGE TABLE
- aircraft number	(d) FLIGHT RECORD TABLE (last aircraft flown)
- invoice number	
- date charged	(d) MEMBER FLIGHT TABLE (d) FLIGHT RECORD TABLE
- hours flown	(d) GENERATE LESSOR STATEMENT module
- aircraft charge	(s) AIRCRAFT RECORD TABLE
- instructor number	
- instructor name	(s) MEMBER RECORD TABLE (d) MEMBER CHARGE FORM
- instruction charge	

- flight supplies charge
- misc. charge type
- misc. charge amount
- misc. credit type
- misc. credit amount
- cumulative flight hours
- flight date
- aircraft flown
- dual flight
- night flight
- number of night landings
- number of day landings
- number of night hours
- number of precision approaches
- number of non-precision approaches
- number of instrument hours

FUNCTION NAME : Enter Member Payments

NUMERICAL IDENTIFIER : 1.1.5

DESCRIPTION : This module receives external input i.e. data from member payments, and routes the data to the appropriate module for processing and/or storage. The source for the data listed below is the individual members' payments and the destination is storage for use later by the GENERATE MEMBER STATEMENT module. Data with different sources or different / additional destinations will be listed below under SOURCE / DESTINATION.

DATA	SOURCE (s) / DESTINATION (d)
- member number	
- member name	(s) MEMBER RECORD TABLE (d) MEMBER CHARGE FORM
- check number	
- date received	(d) MEMBER RECORD TABLE
- amount received	

3. GENERATE REQUIRED REPORTS FUNCTIONAL BREAKDOWN

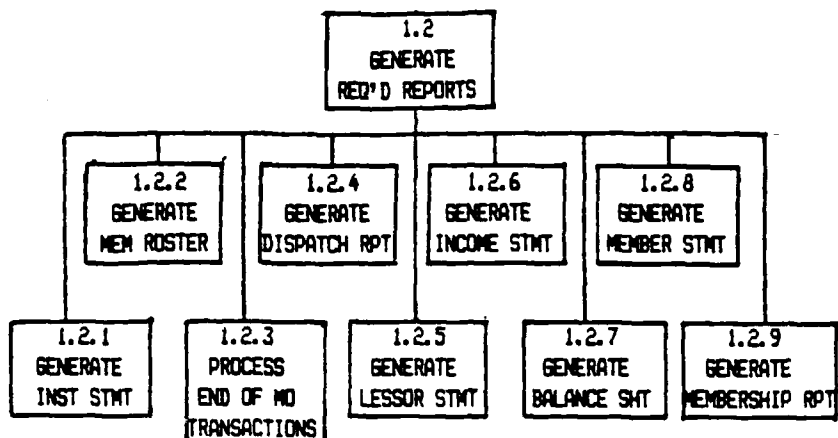


Figure A.4 Generate Required Reports Functional Breakdown.

FUNCTION NAME : Generate Instructor Statement

NUMERICAL IDENTIFIER : 1.2.1

DESCRIPTION : This module generates individual instructor statements.

DATA	SOURCE (s) / DESTINATION (d)
- instructor member #	(s) MEMBER CHARGE DATA
- instructor name	(s) MEMBER RECORD DATA (d) GENERATE INSTRUCTOR STATEMENT
- instructor address	(s) MEMBER RECORD DATA (d) GENERATE INSTRUCTOR STATEMENT
- date	(s) MEMBER CHARGE DATA (d) GENERATE INSTRUCTOR STATEMENT

- member # (student) (s) MEMBER CHARGE DATA
- member name (student) (s) MEMBER RECORD DATA
(d) GENERATE INSTRUCTOR STATEMENT
- instruction charge (s) MEMBER CHARGE DATA
(d) GENERATE INSTRUCTOR STATEMENT

FUNCTION NAME : Generate Member Roster

NUMERICAL IDENTIFIER : 1.2.2

DESCRIPTION : This module retrieves data from
MEMBER RECORD DATA and generates a member roster.

DATA	SOURCE (s) / DESTINATION (d)
- member name	(s) MEMBER RECORD DATA (d) MEMBER ROSTER
- member number	(s) MEMBER RECORD DATA (d) MEMBER ROSTER
- member street	(s) MEMBER RECORD DATA (d) MEMBER ROSTER
- member city	(s) MEMBER RECORD DATA (d) MEMBER ROSTER
- member state	(s) MEMBER RECORD DATA (d) MEMBER ROSTER
- member zipcode	(s) MEMBER RECORD DATA (d) MEMBER ROSTER
- member phone	(s) MEMBER RECORD DATA (d) MEMBER ROSTER

FUNCTION NAME : Process End of Month Transactions

NUMERICAL IDENTIFIER : 1.2.3

DESCRIPTION : This module processes transactions relating to accumulating data from user input. These computations are used by the system to produce the income statement and the balance sheet.

FUNCTION NAME : Generate Dispatcher Report

NUMERICAL IDENTIFIER : 1.2.4

DESCRIPTION : This module retrieves data from the MEMBER RECORD TABLE and produces the DISPATCHER'S REPORT. The source for data used in this module is the MEMBER FLIGHT table and the MEMBER RECORD table. The destination is the DISPATCHER'S REPORT.

DATA	SOURCE (s) / DESTINATION (d)
- MEMBER NAME	(s) MEMBER RECORD TABLE (d) DISPATCHER'S REPORT
- MEMBER NUMBER	(s) MEMBER RECORD TABLE (d) DISPATCHER'S REPORT
- INSTRUCTOR NUMBER	(s) MEMBER RECORD TABLE (d) DISPATCHER'S REPORT
- FAA certificate	(s) MEMBER RECORD TABLE (d) DISPATCHER'S REPORT
- FAA classification	(s) MEMBER RECORD TABLE (d) DISPATCHER'S REPORT
- cum flight hours	(s) FLIGHT RECORD TABLE (d) DISPATCHER'S REPORT
- biennial flight check date	(s) MEMBER RECORD TABLE (d) DISPATCHER'S REPORT
- medical cert. date	(s) MEMBER RECORD TABLE (d) DISPATCHER'S REPORT
- medical cert. classification	(s) MEMBER RECORD TABLE (d) DISPATCHER'S REPORT

- current aircraft models	(s) MEMBER RECORD TABLE
	(d) DISPATCHER'S REPORT
- date last dual	(s) FLIGHT RECORD TABLE
	(d) DISPATCHER'S REPORT
- date last flight	(s) FLIGHT RECORD TABLE
	(d) DISPATCHER'S REPORT
- last aircraft flown	(s) FLIGHT RECORD TABLE
	(d) DISPATCHER'S REPORT
- number of day landings (last 90 days)	(s) MEMBER FLIGHT TABLE
	(d) DISPATCHER'S REPORT
- number of night landings (last 90 days)	(s) MEMBER FLIGHT TABLE
	(d) DISPATCHER'S REPORT
- number of night hours (last 90 days)	(s) MEMBER FLIGHT TABLE
	(d) DISPATCHER'S REPORT
- date last night time	(s) FLIGHT RECORD TABLE
	(d) DISPATCHER'S REPORT
- number precision approaches (last 6 months)	(s) MEMBER FLIGHT TABLE
	(d) DISPATCHER'S REPORT
- number non-precision app. (last 6 months)	(s) MEMBER FLIGHT TABLE
	(d) DISPATCHER'S REPORT
- number instrument hours (last 6 months)	(s) MEMBER FLIGHT TABLE
	(d) DISPATCHER'S REPORT

FUNCTION NAME : Generate Lessor Statement

NUMERICAL IDENTIFIER : 1.2.5

DESCRIPTION : This module receives data from various data stores within the system to produce individual lessor statements.

DATA	SOURCE (s) / DESTINATION (d)
- aircraft number	(s) LESSOR ACCOUNTS TABLE (d) LESSOR STATEMENT
- lessor name	(s) LESSOR ACCOUNTS TABLE (d) LESSOR STATEMENT
- lessor address	(s) LESSOR ACCOUNTS TABLE (d) LESSOR STATEMENT
- hourly lease cost	(s) LESSOR ACCOUNTS TABLE (d) LESSOR STATEMENT
- insurance charge	(s) AIRCRAFT RECORD TABLE (d) LESSOR STATEMENT
- maintenance parts	(s) AIRCRAFT MAINTENANCE TABLE (d) LESSOR STATEMENT
- maintenance labor	(s) AIRCRAFT MAINTENANCE TABLE (d) LESSOR STATEMENT
- date (of maint. action)	(s) AIRCRAFT MAINTENANCE TABLE (d) LESSOR STATEMENT
- hours flown	(s) AIRCRAFT HOURS TABLE (d) LESSOR STATEMENT

FUNCTION NAME : Generate Income Statement
NUMERICAL IDENTIFIER : 1.2.6
DESCRIPTION : This module will scan transaction tables and sum revenue and expense accounts to produce the monthly INCOME STATEMENT. Although not incorporated at this time, the structure and data tables are included in the system for future implementation.

FUNCTION NAME : Generate Balance Sheet
NUMERICAL IDENTIFIER : 1.2.7
DESCRIPTION : This module will take Balance Brought Forward from the BALANCE SHEET VARIABLES, scan the transaction tables for transactions affecting these variables and produce new BALANCE SHEET VARIABLES for use in the current BALANCE SHEET. All necessary structure and data tables are included in the present system to facilitate future implementation.

FUNCTION NAME : Generate Membership Report
NUMERICAL IDENTIFIER : 1.2.8
DESCRIPTION : This module will sort and talley the category column in the MEMBER RECORD table to produce the

membership report required in the Annual Report. Structure and database tables are in place to facilitate future implementation.

FUNCTION NAME : Generate Member Statement

NUMERICAL IDENTIFIER : 1.2.9

DESCRIPTION : This module retrieves data from various modules within the system and generates individual member statements. The destination is the individual member statement unless specified otherwise.

DATA	SOURCE (s) / DESTINATION (d)
- member name	(s) MEMBER RECORD TABLE
- member address	(s) MEMBER RECORD TABLE
- statement date	(s) current system date
- transaction date	(s) BILLTEMP TABLE
- invoice number	(s) BILLTEMP TABLE
- transaction description	(s) BILLTEMP TABLE
- transaction charge	(s) BILLTEMP TABLE
- transaction payment	(S) BILLTEMP TABLE
- balance brought forward	(S) ACCOUNTS RECEIVABLE TABLE
- balance 30 days	(S) ACCOUNTS RECEIVABLE TABLE
- balance 60 days	(S) ACCOUNTS RECEIVABLE TABLE
- balance 90 days	(S) ACCOUNTS RECEIVABLE TABLE

4. PROCESS END OF MONTH TRANSACTIONS FUNCTIONAL BREAKDOWN.

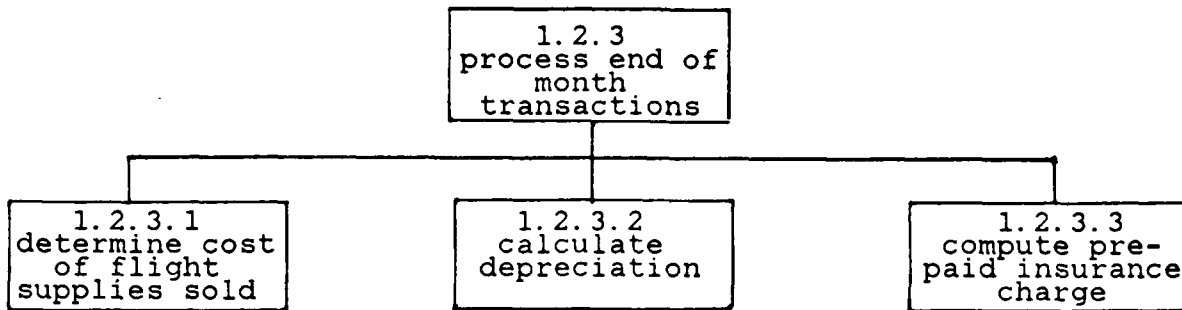


Figure A.5 Process End Of Month Transactions
Functional Breakdown.

FUNCTION NAME : Determine Cost of Flight Supplies Sold

NUMERICAL IDENTIFIER : 1.2.3.1

DESCRIPTION : This module will compute the cost of flight supplies sold by using the beginning balance, and subtracting cost of purchases. This will result in the ending balance which will be sent to storage for generating the INCOME STATEMENT.

DATA	SOURCE (s) / DESTINATION (d)
- beginning inventory	(s) Club Manager
- cost of inventory purchased	(s) Club Manager
- ending inventory	(s) Club Manager
- cost of flight supplies sold	(d) BALANCE SHEET TABLE

FUNCTION NAME : Calculate Depreciation

NUMERICAL IDENTIFIER : 1.2.3.2

DESCRIPTION : This module will calculate the depreciation on capital assets and update the CAPITAL ASSETS TABLE.

DATA	SOURCE (s) / DESTINATION (d)
- book value	(s) CAPITAL ASSETS TABLE
- term	(s) CAPITAL ASSETS TABLE
- accumulated capital depreciation	(s) CAPITAL ASSETS TABLE
- book value	(s) AIRCRAFT RECORD TABLE
- term	(s) AIRCRAFT RECORD TABLE
- accumulated aircraft depreciation	(s) AIRCRAFT RECORD TABLE
- net value	(d) BALANCE SHEET TABLE

FUNCTION NAME : Compute Prepaid Insurance Charge

NUMERICAL IDENTIFIER : 1.2.3.3

DESCRIPTION : This module will compute and assign insurance charges to the appropriate aircraft accounts.

DATA	SOURCE (s) / DESTINATION (d)
- insurance charge	(s) MANAGER INPUT
	(d) AIRCRAFT RECORD TABLE
	(d) LESSOR ACCOUNT TABLE

APPENDIX B
MNEC DATA DICTIONARY

This data dictionary consists of the following items:

- Data Relationship Chart (Figure B.1)
- Listing of tables contained in the database Flyclub
- Listing of column definitions and locations
- A brief description of each data store (table)

Of the 22 data stores (tables) included in the database, 5 are not currently being utilized. They have been included in the database to facilitate future expansion and implementation of additional financial functions not within the scope of this thesis. These five tables are: Accounts Payable, Accounts Receivable, Balance Sheet, Capital Assets, and Cash Accounts.

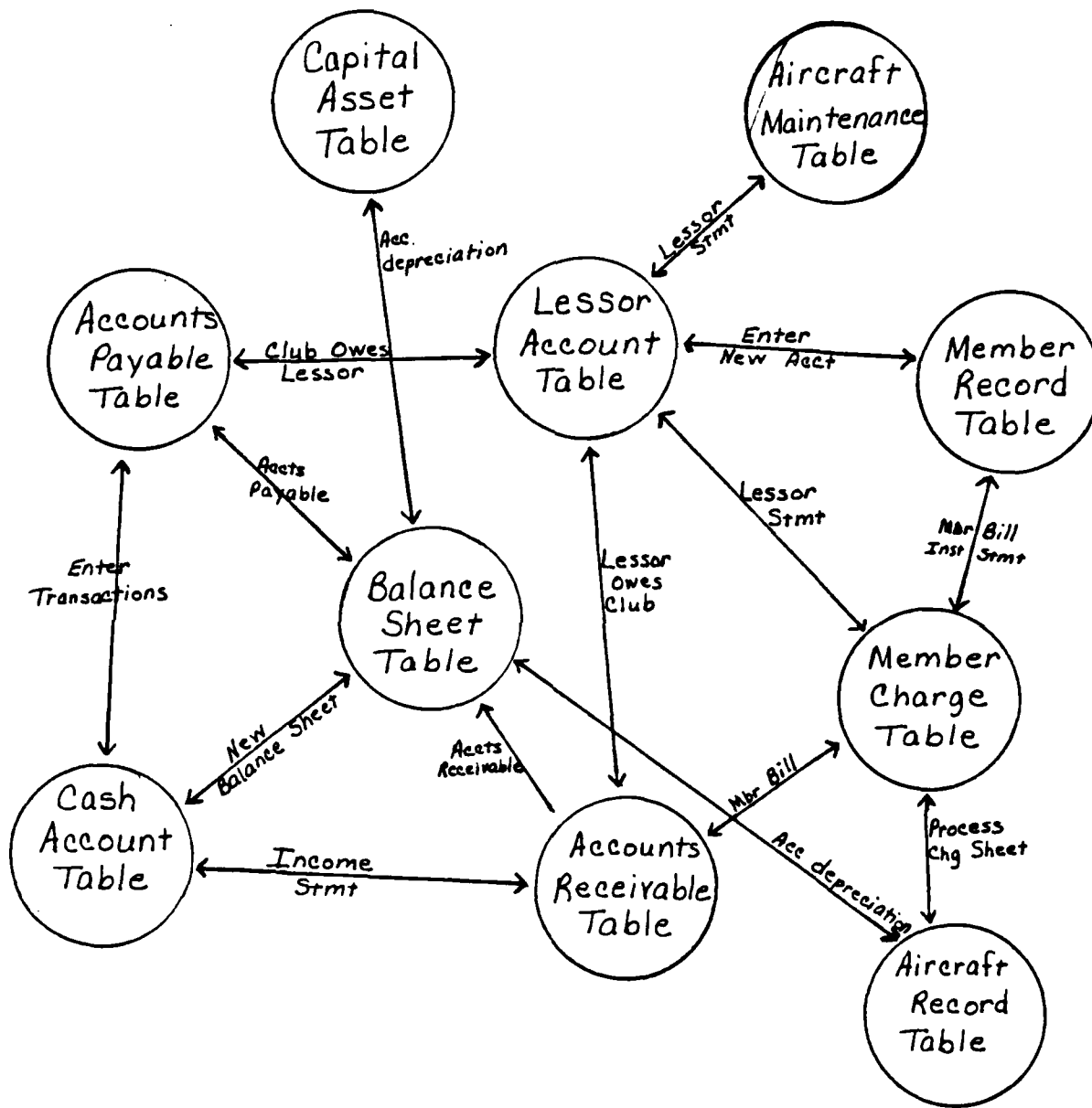


Figure B.1 Data Relationship Chart.

Tables in the Database FLYCLUB

Name	Columns	Rows	Name	Columns	Rows
FORMS	2	203	REPORTS	2	462
ACCT_REC	13	4	ACCT_PAY	4	0
MEM_REC	22	4	BAL_SHET	2	0
MEM_CHG	7	13	MEM_PAY	5	3
BILLTEMP	6	0	PAYHIST	5	0
LES_ACCT	10	2	CASH_ACT	10	0
CHGHIST	7	0	MAINHIST	7	44
TEMP_HRS	3	3	CAP_ASET	5	0
A/C_REC	10	3	HOURLHIST	10	45
A/C_HRS	11	3	INST_REC	6	6
INSTHIST	6	3	FLT_REC	6	1
MEM_FLT	9	2	A/CMAINT	7	4

Report	Table
-----	-----
1MEM_REC	MEM_REC
A/C_STAT	A/C_HRS
DISPATCH	MEM_REC
INSTSTMT	INST_REC
LES_STMT	LES_ACCT
MAINT	A/CMAINT
MEM_BILL	BILLTEMP
ROSTER	MEM_REC

Form	Table
-----	-----
A/C_FORM	A/C_REC
CHGFORM	
HOB_FORM	
LES_FORM	LES_ACCT

MAINTFRM A/CMAINT
MEM_FORM MEM_REC
PAYFORM

Table: A/C_HRS

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	A/C_NUM	TEXT	6 characters	yes
2	CUM_HRS	REAL	1 value(s)	
3	HOB_END	REAL	1 value(s)	
4	HOBSTART	REAL	1 value(s)	
5	HOB_USED	REAL	1 value(s)	
6	HOBMAINT	REAL	1 value(s)	
7	HOB_NET	REAL	1 value(s)	
8	HOBOWNER	REAL	1 value(s)	
9	HOBLEASE	REAL	1 value(s)	
10	LEASEAMT	DOLLAR	1 value(s)	
11	TOTMAINT	DOLLAR	1 value(s)	

Current number of rows: 3

Table: A/CMAINT

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	A/C_NUM	TEXT	6 characters	yes
2	INVO_NUM	TEXT	4 characters	
3	LABOR_HR	REAL	1 value(s)	
4	LABR_CHG	DOLLAR	1 value(s)	
5	PART_CHG	DOLLAR	1 value(s)	
6	A/C_DATE	DATE	1 value(s)	

7 SOURCE TEXT 50 characters

Current number of rows: 4

Table: A/C_REC

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	A/C_NUM	TEXT	6 characters	yes
2	TYPE	TEXT	6 characters	
3	YEAR	TEXT	4 characters	
4	LEAS_RAT	DOLLAR	1 value(s)	
5	RENT_RAT	DOLLAR	1 value(s)	
6	PREP_INS	DOLLAR	1 value(s)	
7	BOOK_VAL	DOLLAR	1 value(s)	
8	ACC_DEP	DOLLAR	1 value(s)	
9	DEP_TERM	INTEGER	1 value(s)	
10	DATE_GOT	DATE	1 value(s)	

Current number of rows: 3

Table: ACCT_PAY

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	WHO/WHAT	TEXT	20 characters	
2	AMOUNT	DOLLAR	1 value(s)	
3	AP_DATE	DATE	1 value(s)	
4	ACCT_NUM	TEXT	4 characters	yes

Current number of rows: 0

Table: ACCT_REC

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	WHO/WHAT	TEXT	20 characters	
2	AR_DATE	DATE	1 value(s)	
3	AMOUNT	DOLLAR	1 value(s)	
4	ACCT_NUM	TEXT	4 characters	yes
5	MEM_NUM	TEXT	4 characters	yes
6	CURR_BAL	DOLLAR	1 value(s)	
7	BAL: 30	DOLLAR	1 value(s)	
8	BAL: 60	DOLLAR	1 value(s)	
9	BAL: 90	DOLLAR	1 value(s)	
10	BALFWD30	DOLLAR	1 value(s)	
11	BALFWD60	DOLLAR	1 value(s)	
12	BALFWD90	DOLLAR	1 value(s)	
13	SUM60+90	DOLLAR	1 value(s)	

Current number of rows: 4

Table: BAL_SHET

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	ACCT_NUM	TEXT	4 characters	yes
2	BALANCE	DOLLAR	1 value(s)	

Current number of rows: 0

Table: BILLTEMP

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	MEM_NUM	TEXT	4 characters	yes
2	INVO_NUM	TEXT	4 characters	
3	TRANDATE	DATE	1 value(s)	
4	CHARGES	DOLLAR	1 value(s)	
5	PAYMENTS	DOLLAR	1 value(s)	
6	DESCRIPT	TEXT	20 characters	

Current number of rows: 0

Table: CAP_ASET

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	IDENT	TEXT	50 characters	yes
2	INIT_CST	DOLLAR	1 value(s)	
3	DATE_ACQ	DATE	1 value(s)	
4	DEP_TERM	INTEGER	1 value(s)	
5	ACC_DEP	DOLLAR	1 value(s)	

Current number of rows: 0

Table: CASH_ACT

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	MEM_NUM	TEXT	4 characters	yes
2	ACCT_NUM	TEXT	4 characters	yes
3	A/C_NUM	TEXT	6 characters	
4	LESOR_NU	TEXT	4 characters	
5	WHO/WHAT	TEXT	20 characters	

6	AMOUNT	DOLLAR	1 value(s)
7	CASHDATE	DATE	1 value(s)
8	CASH_I/O	TEXT	3 characters
9	OTH_ACCT	TEXT	4 characters
10	CHK_SAVE	TEXT	1 characters

Current number of rows: 0

Table: CHGHIST

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	MEM_NUM	TEXT	4 characters	yes
2	INVO_NUM	TEXT	4 characters	
3	TRANDATE	DATE	1 value(s)	
4	CHARGES	DOLLAR	1 value(s)	
5	DESCRIPT	TEXT	20 characters	
6	HRS_FLON	REAL	1 value(s)	
7	A/C_NUM	TEXT	6 characters	

Current number of rows: 0

Table: FLT_REC

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	MEM_NUM	TEXT	4 characters	yes
2	FLT_HRS	REAL	1 value(s)	
3	LAST_FLT	DATE	1 value(s)	
4	LAST_A/C	TEXT	6 characters	
5	LASTDUAL	DATE	1 value(s)	
6	LASTNITE	DATE	1 value(s)	

Current number of rows: 1

Table: HOURHIST

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	A/C_NUM	TEXT	6 characters	yes
2	CUM_HRS	REAL	1 value(s)	
3	HOB_END	REAL	1 value(s)	
4	HOBSTART	REAL	1 value(s)	
5	HOB_USED	REAL	1 value(s)	
6	HOBMAINT	REAL	1 value(s)	
7	HOB_NET	REAL	1 value(s)	
8	HOBOWNER	REAL	1 value(s)	
9	HOBLEASE	REAL	1 value(s)	
10	LEASEAMT	DOLLAR	1 value(s)	

Current number of rows: 45

Table: INSTHIST

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	INST_NUM	TEXT	4 characters	yes
2	TRANDATE	DATE	1 value(s)	
3	MEM_NUM	TEXT	4 characters	
4	HRS_FLON	REAL	1 value(s)	
5	A/C_NUM	TEXT	6 characters	
6	AMOUNT	DOLLAR	1 value(s)	

Current number of rows: 3

Table: INST_REC

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	INST_NUM	TEXT	4 characters	yes
2	TRANDATE	DATE	1 value(s)	
3	MEM_NUM	TEXT	4 characters	
4	HRS_FLON	REAL	1 value(s)	
5	A/C_NUM	TEXT	6 characters	
6	AMOUNT	DOLLAR	1 value(s)	

Current number of rows: 6

Table: LES_ACCT

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	L:NAME	TEXT	15 characters	
2	F:NAME	TEXT	15 characters	
3	MID_INIT	TEXT	2 characters	
4	SSN	TEXT	11 characters	
5	STREET	TEXT	20 characters	
6	CITY	TEXT	15 characters	
7	STATE	TEXT	2 characters	
8	ZIPCODE	TEXT	5 characters	
9	PHONE	TEXT	13 characters	
10	A/C_NUM	TEXT	6 characters	yes

Current number of rows: 2

Table: MAINHIST

Read Password: NO
Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	A/C_NUM	TEXT	6 characters	yes
2	INVO_NUM	TEXT	4 characters	
3	LABOR_HR	REAL	1 value(s)	
4	LABR_CHG	DOLLAR	1 value(s)	
5	PART_CHG	DOLLAR	1 value(s)	
6	A/C_DATE	DATE	1 value(s)	
7	SOURCE	TEXT	50 characters	

Current number of rows: 44

Table: MEM_CHG

Read Password: NO
Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	MEM_NUM	TEXT	4 characters	yes
2	INVO_NUM	TEXT	4 characters	
3	TRANDATE	DATE	1 value(s)	
4	CHARGES	DOLLAR	1 value(s)	
5	DESCRIPT	TEXT	20 characters	
6	HRS_FLON	REAL	1 value(s)	
7	A/C_NUM	TEXT	6 characters	

Current number of rows: 13

Table: MEM_FLT

Read Password: NO
Modify Password: NO

Column definitions

#	Name	Type	Length	Key
---	------	------	--------	-----

1	MEM_NUM	TEXT	4 characters	yes
2	DAILYFLT	REAL	1 value(s)	
3	FLT_DATE	DATE	1 value(s)	
4	DAYLNDG	INTEGER	1 value(s)	
5	EACHNITE	REAL	1 value(s)	
6	NITELNDG	INTEGER	1 value(s)	
7	EACHINST	REAL	1 value(s)	
8	PREC_APP	INTEGER	1 value(s)	
9	NONPREC	INTEGER	1 value(s)	

Current number of rows: 2

Table: MEM_PAY

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	MEM_NUM	TEXT	4 characters	yes
2	CHK_NUM	TEXT	10 characters	
3	TRANDATE	DATE	1 value(s)	
4	PAYMENTS	DOLLAR	1 value(s)	
5	DESCRIPT	TEXT	20 characters	

Current number of rows: 3

Table: MEM_REC

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	MEM_NUM	TEXT	4 characters	yes
2	INST_NUM	TEXT	4 characters	
3	L: NAME	TEXT	15 characters	
4	F: NAME	TEXT	15 characters	

5	MID_INIT	TEXT	2 characters
6	SSN	TEXT	11 characters
7	STREET	TEXT	20 characters
8	CITY	TEXT	15 characters
9	STATE	TEXT	2 characters
10	ZIPCODE	TEXT	5 characters
11	PHONE	TEXT	13 characters
12	CATEGORY	TEXT	2 characters
13	INIT_FEE	TEXT	1 characters
14	KEY_DEP	TEXT	1 characters
15	CUR_MODS	TEXT	30 characters
16	COLLDUTY	TEXT	30 characters

Column definitions

#	Name	Type	Length	Key
17	BIENNIAL	DATE	1 value(s)	
18	MED_CERT	DATE	1 value(s)	
19	MEDCLASS	TEXT	3 characters	
20	FAA_CERT	TEXT	4 characters	
21	FAA_CLAS	TEXT	25 characters	
22	MISHAP	TEXT	1 characters	

Current number of rows: 4

Table: PAYHIST

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	MEM_NUM	TEXT	4 characters	yes
2	CHK_NUM	TEXT	10 characters	
3	TRANDATE	DATE	1 value(s)	
4	PAYMENTS	DOLLAR	1 value(s)	
5	DESCRIPT	TEXT	20 characters	

Current number of rows: 0

Table: TEMP_HRS

Read Password: NO

Modify Password: NO

Column definitions

#	Name	Type	Length	Key
1	CUM_HRS	REAL	1 value(s)	
2	HOB_END	REAL	1 value(s)	
3	A/C_NUM	TEXT	6 characters	yes

Current number of rows: 3

Column definitions and locations:

Name	Type	Length	Table	Key
A/C_DATE	DATE	1 value(s)	A/CMAINT MAINHIST	
A/C_NUM	TEXT	6 characters	CHGHIST MAINHIST A/C_REC INSTHIST A/C_HRS MEM_CHG A/CMAINT CASH_ACT HOURHIST INST_REC LES_ACCT TEMP_HRS	yes yes yes yes
ACCT_NUM	TEXT	4 characters	BAL_SHET CASH_ACT	yes
ACCT_NUM	TEXT	4 characters	ACCT_PAY ACCT_REC	yes yes
ACC_DEP	DOLLAR	1 value(s)	A/C_REC CAP_ASET	
AMOUNT	DOLLAR	1 value(s)	INSTHIST ACCT_REC INST_REC ACCT_PAY CASH_ACT	
AP_DATE	DATE	1 value(s)	ACCT_PAY	
AR_DATE	DATE	1 value(s)	ACCT_REC	
BAL: 30	DOLLAR	1 value(s)	ACCT_REC	
BAL: 60	DOLLAR	1 value(s)	ACCT_REC	
BAL: 90	DOLLAR	1 value(s)	ACCT_REC	
BALANCE	DOLLAR	1 value(s)	BAL_SHET	
BALFWD30	DOLLAR	1 value(s)	ACCT_REC	

BALFWD60	DOLLAR	1 value(s)	ACCT_REC
BALFWD90	DOLLAR	1 value(s)	ACCT_REC
BIENNIAL	DATE	1 value(s)	MEM_REC
BOOK_VAL	DOLLAR	1 value(s)	A/C_REC
CASHDATE	DATE	1 value(s)	CASH_ACT
CASH_I/O	TEXT	3 characters	CASH_ACT
CATEGORY	TEXT	2 characters	MEM_REC
CHARGES	DOLLAR	1 value(s)	BILLTEMP MEM_CHG CHGHIST
CHK_NUM	TEXT	10 characters	MEM_PAY PAYHIST
CHK_SAVE	TEXT	1 characters	CASH_ACT
CITY	TEXT	15 characters	LES_ACCT MEM_REC
COLLDUTY	TEXT	30 characters	MEM_REC

Column definitions and locations:

Name	Type	Length	Table	Key
CUM_HRS	REAL	1 value(s)	TEMP_HRS HOURHIST A/C_HRS	
CURR_BAL	DOLLAR	1 value(s)	ACCT_REC	
CUR_MODS	TEXT	30 characters	MEM_REC	
DAILYFLT	REAL	1 value(s)	MEM_FLT	
DATE_ACQ	DATE	1 value(s)	CAP_ASET	
DATE_GOT	DATE	1 value(s)	A/C_REC	
DAYLNDG	INTEGER	1 value(s)	MEM_FLT	
DEP_TERM	INTEGER	1 value(s)	CAP_ASET A/C_REC	
DESCRIPT	TEXT	20 characters	MEM_CHG MEM_PAY PAYHIST CHGHIST	

		BILLTEMP	
EACHINST	REAL	1 value(s)	MEM_FLT
EACHNITE	REAL	1 value(s)	MEM_FLT
F: NAME	TEXT	15 characters	LES_ACCT MEM_REC
FAA_CERT	TEXT	4 characters	MEM_REC
FAA_CLAS	TEXT	25 characters	MEM_REC
FDATA	TEXT	80 characters	FORMS
FLT_DATE	DATE	1 value(s)	MEM_FLT
FLT_HRS	REAL	1 value(s)	FLT_REC
FNAME	TEXT	8 characters	FORMS yes
HOBLEASE	REAL	1 value(s)	A/C_HRS HOURHIST
HOBMAINT	REAL	1 value(s)	HOURHIST A/C_HRS
HOBOWNER	REAL	1 value(s)	HOURHIST A/C_HRS
HOBSTART	REAL	1 value(s)	A/C_HRS HOURHIST
HOB_END	REAL	1 value(s)	A/C_HRS TEMP_HRS HOURHIST
HOB_NET	REAL	1 value(s)	HOURHIST A/C_HRS
HOB_USED	REAL	1 value(s)	HOURHIST A/C_HRS
HRS_FLON	REAL	1 value(s)	INST_REC INSTHIST CHGHIST MEM_CHG

Column definitions and locations:

Name	Type	Length	Table	Key
IDENT	TEXT	50 characters	CAP_ASET	

INIT_CST	DOLLAR	1 value(s)	CAP_ASET	
INIT_FEE	TEXT	1 characters	MEM_REC	
INST_NUM	TEXT	4 characters	INSTHIST	
			INST_REC	yes
			MEM_REC	
INVO_NUM	TEXT	4 characters	MAINHIST	
			MEM_CHG	
			BILLTEMP	
			A/CMAINT	
			CHGHIST	
KEY_DEP	TEXT	1 characters	MEM_REC	
L: NAME	TEXT	15 characters	MEM_REC	
			LES_ACCT	
LABOR_HR	REAL	1 value(s)	MAINHIST	
			A/CMAINT	
LABR_CHG	DOLLAR	1 value(s)	MAINHIST	
			A/CMAINT	
LASTDUAL	DATE	1 value(s)	FLT_REC	
LASTNITE	DATE	1 value(s)	FLT_REC	
LAST_A/C	TEXT	6 characters	FLT_REC	
LAST_FLT	DATE	1 value(s)	FLT_REC	
LEASEAMT	DOLLAR	1 value(s)	HOURLIST	
			A/C_HRS	
LEAS_RAT	DOLLAR	1 value(s)	A/C_REC	
LESOR_NU	TEXT	4 characters	CASH_ACT	
MEDCLASS	TEXT	3 characters	MEM_REC	
MED_CERT	DATE	1 value(s)	MEM_REC	
MEM_NUM	TEXT	4 characters	BILLTEMP	yes
			FLT_REC	
			INSTHIST	
			ACCT_REC	yes
			MEM_REC	yes
			MEM_CHG	yes

			MEM_PAY	yes
MEM_NUM	TEXT	4 characters	PAYHIST	yes
			CHGHIST	
			INST_REC	
			CASH_ACT	
			MEM_FLT	
MID_INIT	TEXT	2 characters	MEM_REC	
			LES_ACCT	
MISHAP	TEXT	1 characters	MEM_REC	

Column definitions

Name	Type	Length	Table	Key
NITELNDG	INTEGER	1 value(s)	MEM_FLT	
NONPREC	INTEGER	1 value(s)	MEM_FLT	
OTH_ACCT	TEXT	4 characters	CASH_ACT	
PART_CHG	DOLLAR	1 value(s)	A/CMaint	
			MAINHIST	
PAYMENTS	DOLLAR	1 value(s)	BILLTEMP	
			PAYHIST	
			MEM_PAY	
PHONE	TEXT	13 characters	LES_ACCT	
			MEM_REC	
PREC_APP	INTEGER	1 value(s)	MEM_FLT	
PREP_INS	DOLLAR	1 value(s)	A/C_REC	
RDATA	TEXT	132 characters	REPORTS	
RENT_RAT	DOLLAR	1 value(s)	A/C_REC	
RNAME	TEXT	8 characters	REPORTS	yes
SOURCE	TEXT	50 characters	A/CMaint	
			MAINHIST	
SSN	TEXT	11 characters	MEM_REC	
			LES_ACCT	

STATE	TEXT	2 characters	LES_ACCT MEM_REC
STREET	TEXT	20 characters	LES_ACCT MEM_REC
SUM60+90	DOLLAR	1 value(s)	ACCT_REC
TOTMAINT	DOLLAR	1 value(s)	A/C_HRS
TRANDATE	DATE	1 value(s)	INST_REC MEM_CHG CHGHIST PAYHIST BILLTEMP MEM_PAY INSTHIST
TYPE	TEXT	6 characters	A/C_REC
WHO/WHAT	TEXT	20 characters	ACCT_REC ACCT_PAY CASH_ACT
YEAR	TEXT	4 characters	A/C_REC
ZIPCODE	TEXT	5 characters	MEM_REC LES_ACCT

ENTITY NAME : AIRCRAFT HOURS TABLE (A/C_HRS)

DESCRIPTION : Database file containing information relative to aircraft hours flown each month.

ENTITY NAME : AIRCRAFT MAINTENANCE TABLE (A/CMAINT)

DESCRIPTION : Repository for data on maintenance actions completed each month on all aircraft.

ENTITY NAME : AIRCRAFT RECORDS TABLE (A/C_REC)

DESCRIPTION : Contains permanent information on each aircraft used, whether owned or leased, by the Monterey Navy Flying Club.

ENTITY NAME : ACCOUNTS PAYABLE TABLE (ACCT_PAY)

DESCRIPTION : Contains listing of club's accounts payable - not necessarily interactive with the user.

ENTITY NAME : ACCOUNTS RECEIVABLE TABLE (ACCT_REC)

DESCRIPTION : Maintains listing of club's accounts receivables - not necessarily interactive with the user.

ENTITY NAME : BALANCE SHEET TABLE (BAL_SHET)

DESCRIPTION : Contains a series of variables (account number and balance) to be used in constructing the Balance Sheet.

ENTITY NAME : BILLTEMP TABLE (BILLTEMP)

DESCRIPTION : Data store containing records of all charges and payments made during any given billing period. Data is appended from both MEM_CHG and MEM_PAY tables to produce the members' statements.

ENTITY NAME : CAPITAL ASSETS TABLE (CAP_ASET)

DESCRIPTION : Contains data pertaining to shop equipment and aircraft. Used to figure and maintain accumulated depreciation.

ENTITY NAME : CASH ACCOUNTS TABLE (CASH_ACT)

DESCRIPTION : Contains records of all cash transactions.

ENTITY NAME : CHARGE HISTORY TABLE (CHGHIST)

DESCRIPTION : An historical data store where records of monthly charges are archived following completion of the monthly member statements.

ENTITY NAME : FLIGHT RECORD TABLE (FLT_REC)

DESCRIPTION : One of two data stores utilized to maintain flight records on all club members. This store contains data pertaining to flight minimums as published in the dispatcher's report.

ENTITY NAME : HOUR HISTORY TABLE (HOURHIST)

DESCRIPTION : An historical data store where aircraft flight information from the A/C_HRS table is archived following monthly statements.

ENTITY NAME : INSTRUCTOR HISTORY TABLE (INSTHIST)

DESCRIPTION : An historical data store where flight instruction charge data is archived following completion of the monthly Instructor Statements.

ENTITY NAME : INSTRUCTOR RECORD TABLE (INST_REC)

DESCRIPTION : Contains flight instruction charge data.

Utilized to generate the monthly instructor statements.

ENTITY NAME : LESSOR ACCOUNTS TABLE (LES_ACCT)

DESCRIPTION : Data store containing data relating to individuals that own aircraft and have leased them to the club.

ENTITY NAME : MAINTENANCE HISTORY TABLE (MAINHIST)

DESCRIPTION : An historical data store where maintenance action records on each aircraft are archived following the completion of monthly statements.

ENTITY NAME : MEMBER CHARGE TABLE (MEM_CHG)

DESCRIPTION : Data store containing complete record of all member charges entered during any given billing period.

ENTITY NAME : MEMBER FLIGHT TABLE (MEM_FLT)

DESCRIPTION : One of two data stores utilized to maintain flight records on all club members. This store contains data on members' daily flights.

ENTITY NAME : MEMBER PAYMENT TABLE (MEM_PAY)

DESCRIPTION : Data store containing complete records of all payments made by members during any given billing period.

ENTITY NAME : MEMBER RECORD TABLE (MEM_REC)

DESCRIPTION : Data store containing primarily administrative data about each club member. One record per member.

ENTITY NAME : PAYMENT HISTORY TABLE (PAYHIST)

DESCRIPTION : An historical data store where records of monthly payments are archived following completion of the monthly member statements.

ENTITY NAME : TEMPORARY HOURS TABLE (TEMP_HRS)

DESCRIPTION : Store for the temporary storage of aircraft
flight hour data during monthly statement preparation.

APPENDIX C
SOURCE CODE

```
$COMMAND
CLUB_MIS
SET MESSAGE OFF
OPEN FLYCLUB
SET ERROR MESSAGE OFF
SET VAR PICK1 INT
LABEL STARTAPP
NEWPAGE
CHOOSE PICK1 FROM MAIN IN CLUB_MIS.APX
IF PICK1 EQ 0 THEN
  GOTO ENDAPP
ENDIF
IF PICK1 EQ 1 THEN
  SET VAR PICK2 INT
  SET VAR LEVEL2 INT
  SET VAR LEVEL2 TO 0
  WHILE LEVEL2 EQ 0 THEN
    NEWPAGE
    CHOOSE PICK2 FROM SUBMENIO IN CLUB_MIS.APX
    IF PICK2 EQ 0 THEN
      BREAK
```

```
ENDIE
IF PICK2 EQ 1 THEN
  RUN ADD_MEM IN CLUB_MIS.APX
ENDIE
IF PICK2 EQ 2 THEN
  FILLIN WHVAL USING "ENTER NUMBER FOR MEMBER TO BE CHANGED "
  EDIT USING MEM_FORM +
    SORTED BY MEM_NUM L:NAME +
    WHERE MEM_NUM EQ .WHVAL
  CLEAR WHVAL
ENDIE
IF PICK2 EQ 3 THEN
  BREAK
ENDIE
ENDWHILE
CLEAR LEVEL2
CLEAR PICK2
GOTO STARTAPP
ENDIE
IF PICK1 EQ 2 THEN
  SET VAR PICK2 INT
  SET VAR LEVEL2 INT
  SET VAR LEVEL2 TO 0
  WHILE LEVEL2 EQ 0 THEN
    NEWPAGE
```

```

CHOOSE PICK2 FROM SUBMEN20 IN CLUB_MIS.APX
IF PICK2 EQ 0 THEN
  BREAK
ENDIF
IF PICK2 EQ      1 THEN
  SET VAR PICK3 INT
  SET VAR LEVEL3 INT
  SET VAR LEVEL3 TO 0
  WHILE LEVEL3 EQ 0 THEN
    NEWPAGE
    CHOOSE PICK3 FROM SUBMEN21 IN CLUB_MIS.APX
    IF PICK3 EQ 0 THEN
      BREAK
    ENDIF
    IF PICK3 EQ      1 THEN
      ENDIF
    IF PICK3 EQ      2 THEN
      ENDIF
    IF PICK3 EQ      3 THEN
      ENDIF
    IF PICK3 EQ      4 THEN
      RUN PUTCHG IN CLUB_MIS.APX
    ENDIF
    IF PICK3 EQ      5 THEN
      RUN PUTPAY IN CLUB_MIS.APX
    ENDIF
    IF PICK3 EQ      6 THEN
      ENDIF
  
```



```

RUN ADDMAINT IN CLUB_MIS.APX
ENDIF
IF PICK3 EQ 6 THEN
    BREAK
ENDIF
ENDWHILE
CLEAR LEVEL3
CLEAR PICK3
ENDIF
IF PICK2 EQ 2 THEN
    SET VAR PICK3 INT
    SET VAR LEVEL3 INT
    SET VAR LEVEL3 TO 0
    WHILE LEVEL3 EQ 0 THEN
        NEWPAGE
        CHOOSE PICK3 FROM SUBMEN22 IN CLUB_MIS.APX
        IF PICK3 EQ 0 THEN
            BREAK
        ENDIF
        IF PICK3 EQ 1 THEN
            ENDIF
        IF PICK3 EQ 2 THEN
            ENDIF
        IF PICK3 EQ 3 THEN
            ENDIF
    
```

```
IF PICK3 EQ 4 THEN
ENDIF
IF PICK3 EQ 5 THEN
BREAK
ENDIF
ENDWHILE
CLEAR LEVEL3
CLEAR PICK3
ENDIF
IF PICK2 EQ 3 THEN
BREAK
ENDIF
ENDWHILE
CLEAR LEVEL2
CLEAR PICK2
GOTO STARTAPP
ENDIF
IF PICK1 EQ 3 THEN
SET VAR PICK2 INT
SET VAR LEVEL2 INT
SET VAR LEVEL2 TO 0
WHILE LEVEL2 EQ 0 THEN
NEWPAGE
CHOOSE PICK2 FROM SUBMEN30 IN CLUB_MIS.APX
IF PICK2 EQ 0 THEN
```

```
BREAK
ENDIF
IF PICK2 EQ 1 THEN
  SET VAR PICK3 INT
  SET VAR LEVEL3 INT
  SET VAR LEVEL3 TO 0
  WHILE LEVEL3 EQ 0 THEN
    NEWPAGE
    CHOOSE PICK3 FROM SUBMEN31 IN CLUB_MIS.APX
    IF PICK3 EQ 0 THEN
      BREAK
    ENDIF
    IF PICK3 EQ 1 THEN
      RUN ROSTER IN CLUB_MIS.APX
    ENDIF
    IF PICK3 EQ 2 THEN
      RUN DISPATCH IN CLUB_MIS.APX
    ENDIF
    IF PICK3 EQ 3 THEN
      BREAK
    ENDIF
    IF PICK3 EQ 4 THEN
      BREAK
    ENDIF
  ENDWHILE
  CLEAR LEVEL3

```

```
CLEAR PICK3
ENDIF
IF PICK2 EQ 2 THEN
  SET VAR PICK3 INT
  SET VAR LEVEL3 INT
  SET VAR LEVEL3 TO 0
  WHILE LEVEL3 EQ 0 THEN
    NEWPAGE
    CHOOSE PICK3 FROM SUBMEN32 IN CLUB_MIS.APX
    IF PICK3 EQ 0 THEN
      BREAK
    ENDIF
    IF PICK3 EQ 1 THEN
      RUN MEMSTMT IN CLUB_MIS.APX
    ENDIF
    IF PICK3 EQ 2 THEN
      RUN INSTSTMT IN CLUB_MIS.APX
    ENDIF
    IF PICK3 EQ 3 THEN
      RUN LES_STMT IN CLUB_MIS.APX
    ENDIF
    IF PICK3 EQ 4 THEN
      ENDIF
    ENDIF
    IF PICK3 EQ 5 THEN
      ENDIF
    ENDIF
  
```

```
IF PICK3 EQ 6 THEN
ENDIF
IF PICK3 EQ 7 THEN
  BREAK
ENDIF
ENDWHILE
CLEAR LEVEL3
CLEAR PICK3
ENDIF
IF PICK2 EQ 3 THEN
  SET VAR PICK3 INT
  SET VAR LEVEL3 INT
  SET VAR LEVEL3 TO 0
  WHILE LEVEL3 EQ 0 THEN
    NEWPAGE
    CHOOSE PICK3 FROM SUBMEN3 IN CLUB_MIS.APX
    IF PICK3 EQ 0 THEN
      BREAK
    ENDIF
    IF PICK3 EQ 1 THEN
      ENDIF
    IF PICK3 EQ 2 THEN
      ENDIF
    IF PICK3 EQ 3 THEN
      ENDIF
    
```

```
IF PICK3 EQ      4 THEN
  BREAK
  ENDIF
  ENDWHILE
  CLEAR LEVEL3
  CLEAR PICK3
  ENDIF
IF PICK2 EQ      4 THEN
  BREAK
  ENDIF
  ENDWHILE
  CLEAR LEVEL2
  CLEAR PICK2
  GOTO STARTAPP
  ENDIF
IF PICK1 EQ      4 THEN
  GOTO ENDAPP
  GOTO STARTAPP
  ENDIF
IF PICK1 EQ      5 THEN
  RUN BACKPACK IN CLUB_MIS.APX
  GOTO STARTAPP
  ENDIF
IF PICK1 EQ      6 THEN
  GOTO ENDAPP
```

ENDIE
GOTO STARTAPP
LABEL ENDAPP
CLEAR PICK1
RETURN
\$MENU
MAIN
COLUMN MNEC MAIN MENU
MEMBERSHIP TRANSACTIONS
FINANCIAL TRANSACTIONS
PRINT REPORTS / STATEMENTS
GENERATE END OF MONTH TRANSACTIONS
BACK UP AND PACK THE DATABASE
RETURN TO SYSTEM DOS
\$MENU
SUBMEN10
COLUMN MEMBERSHIP TRANSACTIONS
ENTER A NEW MEMBER RECORD
EDIT / DELETE EXISTING MEMBER RECORD
RETURN TO MAIN MENU
\$MENU
SUBMEN20
COLUMN FINANCIAL TRANSACTIONS
ENTER A NEW TRANSACTION
EDIT / DELETE AN EXISTING TRANSACTION

RETURN TO MAIN MENU
\$MENU
SUBMEN22
COLUMN EDIT / DELETE AN EXISTING TRANSACTION
ACCOUNTS RECEIVABLE
ACCOUNTS PAYABLE
MEMBER CHARGE TRANSACTIONS
MEMBER PAYMENT TRANSACTION
RETURN TO FINANCIAL TRANSACTIONS MENU
\$MENU
SUBMEN21
COLUMN ENTER A NEW TRANSACTION
ACCOUNTS RECEIVABLE
ACCOUNTS PAYABLE
ENTER A MEMBER'S CHARGES
ENTER A MEMBER'S PAYMENTS
ENTER A MAINTENANCE TRANSACTION
RETURN TO FINANCIAL TRANSACTIONS MENU
\$MENU
SUBMEN30
COLUMN PRINT REPORTS / STATEMENTS
ADMINISTRATIVE REPORTS
FINANCIAL STATEMENTS
ANNUAL REPORTS
RETURN TO MAIN MENU

\$MENU
SUBMEN31
COLUMN PRINT ADMINISTRATIVE REPORTS
PRINT MEMBER ROSTER
PRINT DISPATCHER REPORT
PRINT MANAGER'S REPORT
RETURN TO PRINT REPORTS/STATEMENTS MENU
\$MENU
SUBMEN32
COLUMN PRINT FINANCIAL STATEMENTS
PRINT MEMBER STATEMENTS
PRINT INSTRUCTOR STATEMENTS
PRINT LESSOR STATEMENTS AND AIRCRAFT INVENTORY/STATUS
PRINT AGED ACCOUNTS RECEIVABLE
PRINT BALANCE SHEET INPUTS
PRINT INCOME STATEMENT INPUTS
RETURN TO PRINT REPORTS/STATEMENTS MENU
\$MENU
SUBMEN33
COLUMN PRINT ANNUAL REPORTS
PRINT MEMBERSHIP BREAKDOWN
PRINT STATEMENT OF NET WORTH DATA
PRINT DATA FOR APPENDIX A OF ANNUAL REPORT
RETURN TO PRINT REPORTS/STATEMENTS MENU
\$COMMAND

BACKPACK

*(*****)

PROGRAM: BACKPACK.COMD

AUTHOR: D.R.GEORGE

DATE WRITTEN: MARCH, 1986

DESCRIPTION: CREATES A BACKUP OF THE DATABASE AND REPACKS IT

*****)

SET MESSAGES OFF

SET ERROR MESSAGES OFF

NEWPAGE

WRITE "INSERT A FORMATTED DISK IN DRIVE A" AT 5,10

WRITE "PRESS ANY KEY TO CONTINUE" AT 7,14

PAUSE

SET ERR VAR ERV

COPY FLYCLUB?.RBS A:

IF ERV = 0 THEN

PACK FLYCLUB

IF ERV = 0 THEN

COPY FLYCLUB?.RBS A:

ENDIF

ELSE

NEWPAGE

WRITE "TRY ANOTHER DISK, UNABLE TO BACKUP DATABASE" AT 15,10

WRITE "PRESS ANY KEY TO CONTINUE" AT 17,14

AD-A169 183

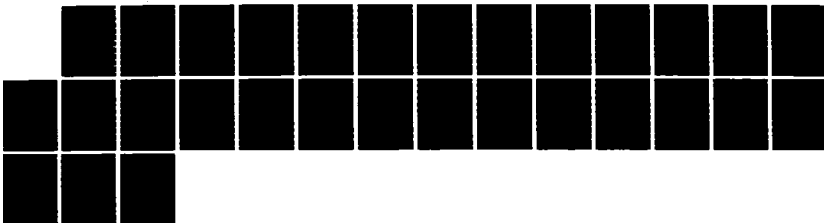
THE DESIGN AND DEVELOPMENT OF A MANAGEMENT INFORMATION
SYSTEM FOR THE MONTEREY NAVY FLYING CLUB(U) NAVAL
POSTGRADUATE SCHOOL MONTEREY CA D R GEORGE 27 MAR 86

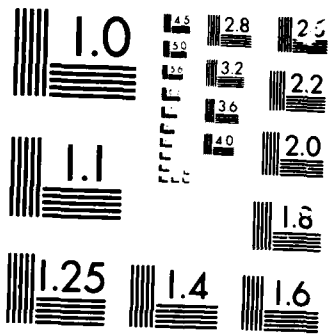
2/2

UNCLASSIFIED

F/G 5/1

ML





MICROCOPY

1010

```
PAUSE
BREAK
ENDIF
NEWPAGE
WRITE "THE PACK IS COMPLETE, AT 5,15
WRITE "THE DISK IN DRIVE 'A' CONTAINS A BACKUP OF THE DATABASE" AT 7,4
WRITE "PRESS ANY KEY TO CONTINUE" AT 9,12
PAUSE
RETURN
$COMMAND
ADD_MEM
ENTER MEM_FORM
RETURN
```

```
$COMMAND
PUTCHG
*( *****
```

```
PROGRAM: PUTCHG.CMD
AUTHOR: D. R. GEORGE
DATE WRITTEN: DECEMBER, 1985
DESCRIPTION: ENTERS MEMBER'S CHARGES ( INPUT BY MANAGER
```

FROM INDIVIDUAL CHARGE SHEETS)
TABLES USED: MEM_REC, MEM_CHG, MEM_FLT, INST_REC
FORM USED: CHGFORM

*****)

```
*(- Establish message parameters)
SET MESSAGES OFF
SET ERROR MESSAGES OFF
OPEN FLYCLUB
*(Establish loop for entering member number)
SET VARIABLE VMEMNUM TEXT
SET VARIABLE VMEMNUM TO 0
WHILE VMEMNUM EXISTS THEN
  NEWPAGE
  CLEAR VINVONUM
  CLEAR VTRANDAT
  CLEAR VACCHARG
  CLEAR VHRSEFLON
  CLEAR VACNUM
  CLEAR VINSTNAM
  CLEAR VINCHARG
  CLEAR VFSCHARG
  CLEAR VMSCHARG
  CLEAR VMSCREDT
  CLEAR VCRTYPE
```

```

CLEAR VCHGTYPE
CLEAR FELT_HRS
CLEAR VDAYLDG
CLEAR VNITELDG
CLEAR VNITEHRS
CLEAR VPREAPP
CLEAR VNONPRE
CLEAR VINSTIME
*(Get member number)
WRITE "Add New Charges (ESC to quit)" AT 2,1
FILLIN VMEMNUM USING "Enter member number - " AT 4,1
*(If no member number entered, exit)
IF VMEMNUM FAILS THEN
    SET MESSAGES ON
    SET ERROR MESSAGES ON
    BREAK
ENDIF *(VMEMNUM fails)
*(Look up member number in MEM_REC table)
SET POINTER #1 STATUS1 FOR MEM_REC WHERE MEM_NUM = .VMEMNUM
*(If member number not found, warn user)
IF STATUS1 <> 0 THEN
    WRITE "No such member number!" AT 18,1
    WRITE "Press any key to continue." AT 19,1
    PAUSE
    BREAK

```

```

*(Otherwise set up variables for screen)
ELSE
  SET VARIABLE NAME1 TO L:NAME IN #1
  SET VARIABLE NAME2 TO .NAME1 + ", "
  SET VARIABLE NAME3 TO F:NAME IN #1
  SET VARIABLE NAME4 TO .NAME2 & .NAME3
  SET VARIABLE NAME5 TO MID_INIT IN #1
  SET VARIABLE VFULLNAM TO .NAME4 & .NAME5
  SET VARIABLE VEXFUEL TO "EXCESS FUEL CHARGE"
  SET VARIABLE VINITFEE TO "INITIATION FEE"
  SET VARIABLE VKEYDEP TO "KEY DEPOSIT"
  SET VARIABLE VINITDUE TO "INITIAL DUES"
  SET VARIABLE VFUELCR TO "FUEL CREDIT"
  SET VARIABLE VKEYRET TO "KEY RETURN"
  SET VARIABLE VDUESCR TO "DUES CREDIT"
  SET VARIABLE VACRENT TO "AIRCRAFT RENTAL"
  SET VARIABLE VINCHG TO "INSTRUCTION CHARGE"
  SET VARIABLE VFSCHG TO "FLIGHT SUPPLIES"
  SET VARIABLE VFILT_HRS TO FLT_HRS IN FLT_REC WHERE MEM_NUM = .VMEMNUM

ENDIF *( Status1 <> 0)
NEWPAGE
DRAW CHGFORM WITH VARIABLE VMEMNUM VFULLNAM VFILT_HRS
WRITE "PRESS PGDN KEY TO ENTER DATA - PRESS ESC KEY TO QUIT" AT 24,8
ENTER VARIABLE VACNUM VINVONUM VTRANDAT VHRSEFON VINSTNUM VINCHARG +
VFSCHARG VCHCTYPE VMSCHARG VCRTYPE VMSCREDIT VDAYLDG +

```



```

VNITELDG VNITEHRS VPREAPP VNONPRE VINSTIME RETURN PGDN ESC
*(If escape, break loop)
IF #RETURN = ESC THEN
    BREAK
ENDIF *(Escape)
*(Look up aircraft number)
SET POINTER #2 STATUS2 FOR A/C_REC WHERE A/C_NUM = .VACNUM
*( If aircraft number found, compute rental charge)
IF STATUS2 = 0 THEN
    SET VARIABLE VRENT TO RENT_RAT IN #2
    SET VARIABLE VACCHARG TO .VHRSFLOX X .VRENT
ENDIF *(Status2 = 0)
*(- Look up instructor name)
SET POINTER #3 STATUS3 FOR MEM_REC WHERE INST_NUM = .VINSTNUM
IF STATUS3 = 0 THEN
    SET VARIABLE VINSTNAM TO L:NAME IN #3
ENDIF *(- look up instructor)
NEWPAGE
DRAW CHGFORM WITH ALL
WRITE " PRESS ANY KEY TO CONTINUE - ESC TO QUIT " AT 24,15
PAUSE
IF VACCHARG <> 0 THEN *(AIRCRAFT RENTAL)
    LOAD MEM_CHG
        .VMEMNUM .VINONUM .VTRANDAT .VACCHARG .VACRENT +
        .VHRSFLOX .VACNUM

```

```

END *(Load )
*(-- Update cum flight hours, last date flown, and last aircraft flown)
LOAD MEM_FLT
    .VMEMNUM .VHRSEFLON .VTRANDAT .VDAYLDG .VNITEHRS .VNITELDG +
    .VINSTIME .VPREAPP .VNONPRE
END *(-- Load )
IF VHRSEFLON EXISTS THEN
    IF VELT_HRS EXISTS THEN
        ASSIGN FLT_HRS TO FLT_HRS + .VHRSEFLON IN FLT_REC WHERE MEM_NUM +
            = .VMEMNUM
    ELSE
        LOAD FLT_REC USING MEM_NUM FLT_HRS LAST_FLT LAST_A/C
            .VMEMNUM .VHRSEFLON .VTRANDAT .VACNUM
        END *( load )
    ENDIF
    CHANGE LAST_FLT TO .VTRANDAT IN FLT_REC WHERE MEM_NUM = .VMEMNUM
    CHANGE LAST_A/C TO .VACNUM IN FLT_REC WHERE MEM_NUM = .VMEMNUM
ENDIF *(- VHRSEFLON exists )
*(-- load night time)
IF VNITEHRS > 0 THEN
    CHANGE LASTNITE TO .VTRANDAT IN FLT_REC WHERE MEM_NUM = .VMEMNUM
ENDIF *(-- load night time )
ENDIF *(Aircraft rental)

```

```

IF VINCHARG <> 0 THEN *(Instruction charge)
*(-- Load INST_REC table for generating instructor statements)
LOAD INST_REC
    .VINSTNUM .VTRANDAT .VMEMNUM .VHRSEFON .VACNUM .VINCHARG
END *(-- Load )
LOAD MEM_CHG
    .VMEMNUM .VINVONUM .VTRANDAT .VINCHARG .VINCHG +
    .VHRSEFON .VACNUM
END *(Load)
*(-- update last instruction date )
CHANGE LASTDUAL TO .VTRANDAT IN FLT_REC WHERE MEM_NUM = .VMEMNUM
ENDIF *(End instruction charge)

SET VARIABLE VACNUM TO " "
SET VARIABLE VHRSEFON TO " "
IF VFESCHARG <> 0 THEN *(Flight supplies charge)
LOAD MEM_CHG
    .VMEMNUM .VINVONUM .VTRANDAT .VFESCHARG .VFESCHG +
    .VHRSEFON .VACNUM
END *(Load)
ENDIF *(End instruction charge)
IF VCHGTYPE = 1 THEN *(Excess fuel charge)
LOAD MEM_CHG
    .VMEMNUM .VINVONUM .VTRANDAT .VMSCHARG .VEXFUEL +
    .VHRSEFON .VACNUM
END *(Load )

```

```

ENDIF *(Excess fuel)
IF VCHGTYPE = 2 THEN *(Initiation fee)
  LOAD MEM_CHG
  .VMEMNUM .VINVONUM .VTRANDAT .VMSCHARG .VINITFEE +
  .VHRSEFLON .VACNUM
  END *(Load )
ENDIF *(Initiation fee)
IF VCHGTYPE = 3 THEN *(Key deposit)
  LOAD MEM_CHG
  .VMEMNUM .VINVONUM .VTRANDAT .VMSCHARG .VKEYDEP +
  .VHRSEFLON .VACNUM
  END *(Load )
ENDIF *(Key deposit)
IF VCHGTYPE = 4 THEN *(Initial dues)
  LOAD MEM_CHG
  .VMEMNUM .VINVONUM .VTRANDAT .VMSCHARG .VINITDUE +
  .VHRSEFLON .VACNUM
  END *(Load )
ENDIF *(Initial dues)
SET VARIABLE VMSCREDIT = .VMSCREDIT X -1
IF VCRTYPE = 5 THEN *(Fuel credit)
  LOAD MEM_CHG
  .VMEMNUM .VINVONUM .VTRANDAT .VMSCREDIT .VFUELCR +
  .VHRSEFLON .VACNUM
  END *(Load )

```

```
ENDIF *(Fuel credit)
IF VCRTYPE = 6 THEN *(Key return)
  LOAD MEM_CHG
    .VMEMNUM .VINVONUM .VTRANDAT .VMSCREDIT .VKEYRET +
    .VHRSEFLON .VACNUM
  END *(Load )
ENDIF *(Key return)
IF VCRTYPE = 7 THEN *(Dues credit)
  LOAD MEM_CHG
    .VMEMNUM .VINVONUM .VTRANDAT .VMSCREDIT .VDUESCR +
    .VHRSEFLON .VACNUM
  END *(Load )
ENDIF *(Dues credit)
ENDWHILE
RETURN
```



```

NEWPAGE
CLEAR VCHKNUM
CLEAR VTRANDAT
CLEAR VTRANAMT
*(Get member number)
WRITE "Add New Payments      (ESC to Quit)" AT 2,1
FILLIN VMEMNUM USING "Enter Member Number - " AT 4,1
*(If no member number entered, exit)
IF VMEMNUM FAILS THEN
    SET MESSAGES ON
    SET ERROR MESSAGES ON
    BREAK
ENDIF *(vmemnum fails)
*(Look up member number in MEM_REC table)
SET POINTER #1 STATUS1 FOR MEM_REC WHERE MEM_NUM = . VMEMNUM
*( If member number not found, warn user)
IF STATUS1 <> 0 THEN
    WRITE "No such member number!" AT 18,1
    WRITE "Press any key to continue." AT 19,1
    PAUSE
ELSE
    SET VARIABLE NAME1 TO L:NAME IN #1
    SET VARIABLE NAME2 TO .NAME1 + ", "
    SET VARIABLE NAME3 TO F:NAME IN #1
    SET VARIABLE NAME4 TO .NAME2 & .NAME3

```

```
SET VARIABLE NAMES TO MID_INIT IN #1
SET VARIABLE VFULLNAM TO .NAME4 & .NAME5
SET VARIABLE VTRANDAT TO .#DATE
ENDIF *(Status1 <> 0)
NEWPAGE
DRAW PAYFORM WITH VARIABLE VMEMNUM VFULLNAM VTRANDAT
WRITE "PRESS PAGE DOWN KEY TO ENTER DATA - PRESS ESC KEY TO QUIT" AT 22,8
ENTER VARIABLE VCHKNUM VPAYMENT RETURN PGDN ESC
*(If escape, break loop)
IF #RETURN = ESC THEN
BREAK
ENDIF *(Escape)
LOAD MEM_PAY
.VMEMNUM .VCHKNUM .VTRANDAT .VPAYMENT PAYMENT
END *(End of Load)
ENDWHILE
RETURN
```


\$COMMAND
ADDMAINT
ENTER MAINTERM
RETURN
\$COMMAND
ROSTER

*(*****)

PROGRAM: ROSTER.COMD
AUTHOR: D. R. GEORGE
DATE WRITTEN: FEBRUARY, 1986
DESCRIPTION: PRINTS THE MNFC ROSTER
TABLE USED: MEM_REC
REPORT PRODUCED: CLUB ROSTER

*****)

```

NEWPAGE
WRITE "THE CLUB ROSTER MAY BE PRINTED AS OFTEN AS DESIRED." AT 5,8
WRITE "UPDATES ARE PERFORMED WHENEVER MEMBERS ARE ENTERED." AT 7,8
WRITE "DO YOU WANT TO PROCEED? (Y/N) - " AT 11,17
SET VARIABLE YESNO TEXT
FILLIN YESNO USING " " AT 11,50
IF YESNO <> Y THEN
    GOTO BAILOUT
ENDIF *(-- yesno <> Y)
NEWPAGE
WRITE "ENSURE PRINTER IS READY, THEN PRESS ANY KEY TO CONTINUE." AT 10,5
PAUSE

NEWPAGE
WRITE "WORKING...PLEASE STANDBY." AT 10,20
SET MESSAGES OFF
SET ERROR MESSAGES OFF

*(-- Print club roster )
OUTPUT PRINTER
SET NULL " "
SET VARIABLE TOTNUM INTEGER
COMPUTE TOTNUM AS COUNT MEM_NUM FROM MEM_REC
PRINT ROSTER SORTED BY L:NAME F:NAME

```

```
OUTPUT SCREEN
SET NULL -O-

NEWPAGE
SET MESSAGES ON
SET ERROR MESSAGES ON
WRITE "CLUB ROSTER COMPLETE." AT 7,17
WRITE "PRESS ANY KEY TO RETURN TO DOS." AT 10,15
PAUSE
NEWPAGE
RETURN

*(-- Bailout routine for immediate exit)
LABEL Bailout
NEWPAGE
WRITE " Returning to Main Menu Without Update." AT 7,10
*(-- End Of Program )
RETURN
$COMMAND
DISPATCH
*(*****
```

```
PROGRAM: DISPATCH.CMD
AUTHOR: D. R. GEORGE
DATE WRITTEN: FEBRUARY, 1986
DESCRIPTION: PRINTS THE DISPATCHER REPORT - INFORMATION IS
              IS UPDATED WHENEVER MEMBER CHARGES ARE ENTERED.
```

TABLES USED: MEM_REC, MEM_FLT
REPORTS PRODUCED: DISPATCHER REPORT (DISPATCH)

*****)

NEWPAGE
WRITE "THE DISPATCHER REPORT MAY BE PRINTED AS OFTEN AS DESIRED." AT 5,8
WRITE "UPDATES ARE PERFORMED WHENEVER CHARGES ARE ENTERED." AT 7,11
WRITE "DO YOU WANT TO PROCEED? (Y/N) - " AT 11,17
SET VARIABLE YESNO TEXT
FILLIN YESNO USING " " AT 11,50
IF YESNO <> Y THEN

GOTO BAILOUT
ENDIF *(-- yesno <> y)
NEWPAGE
WRITE "ENSURE PRINTER IS READY, THEN PRESS ANY KEY TO CONTINUE." AT 10,5
PAUSE

NEWPAGE
WRITE "WORKING...PLEASE STANDBY." AT 10,20
SET MESSAGES OFF
SET ERROR MESSAGES OFF

*(-- Print dispatcher report)
OUTPUT PRINTER
SET NULL " "
PRINT DISPATCH SORTED BY L: NAME F: NAME

```
OUTPUT SCREEN
SET NULL -0-
NEWPAGE
SET MESSAGES ON
SET ERROR MESSAGES ON
CLEAR ALL VARIABLES
WRITE "DISPATCHER REPORT COMPLETE." AT 7,17
WRITE "PRESS ANY KEY TO RETURN TO DOS." AT 10,15
PAUSE
NEWPAGE
RETURN
*(-- Bailout routine for immediate exit)
LABEL Bailout
NEWPAGE
WRITE " Returning to Main Menu Without Update." AT 7,10
*(-- End Of Program )
RETURN
$COMMAND
INSTMTMT
*( *****
```

```
PROGRAM: INSTMTMT.CMD
AUTHOR: D.R.GEORGE
DATE WRITTEN: FEBRUARY, 1986
DESCRIPTION: COMPUTES AND PRINTS INSTRUCTOR STATEMENTS
```

TABLES USED: INST_REC, INSTHIST
REPORTS PRODUCED: INSTRUCTOR STATEMENT (INSTSTMT)

*****)

```
NEWPAGE
WRITE "STATEMENTS ARE TO BE PRINTED ONCE A MONTH ONLY." AT 5,12
WRITE "MONTHLY UPDATES ARE AUTOMATICALLY PERFORMED." AT 7,13
WRITE "ENSURE A BACKUP COPY OF THE DATABASE HAS BEEN MADE." AT 9,10
WRITE "DO YOU WANT TO PROCEED? (Y/N) - " AT 11,17
SET VARIABLE YESNO TEXT
FILLIN YESNO USING " " AT 11,50
IF YESNO <> Y THEN
    GOTO BAILOUT
ENDIF *((-- yesno <> y)
NEWPAGE
WRITE "ENSURE PRINTER IS READY, THEN PRESS ANY KEY TO CONTINUE." AT 10,5
PAUSE
NEWPAGE
WRITE "WORKING...PLEASE STANDBY." AT 10,20
SET MESSAGES OFF
SET ERROR MESSAGES OFF

*((-- Print instructor statements )
OUTPUT PRINTER
```

```

SET NULL " "
SET POINTER #1 STATUS1 FOR MEM_REC SORTED BY L:NAME F:NAME WHERE +
INST_NUM EXISTS
WHILE STATUS1 = 0 THEN
    SET VARIABLE TEMPNUM1 TO INST_NUM IN #1
    PRINT INSTSTMT WHERE INST_NUM = .TEMPNUM1
NEXT #1 STATUS1
ENDWHILE *((-- Print instructor statements)
OUTPUT SCREEN
SET NULL -0-
*((-- Move transactions to history table)
APPEND INST_REC TO INSTHIST
*((-- Prepare table for next month's entries)
DELETE ROWS FROM INST_REC WHERE INST_NUM EXISTS
*((-- Updates complete)
NEWPAGE
SET MESSAGES ON
SET ERROR MESSAGES ON
WRITE "Monthly Instructor Statements Complete." AT 7,10
WRITE "PRESS ANY KEY TO RETURN TO DOS." AT 10,15
PAUSE
NEWPAGE
RETURN

```

```

*(-- Bailout routine for immediate exit)
LABEL Bailout
NEWPAGE
WRITE " Returning to Main Menu Without Update." AT 7,10
*(-- End Of Program )
RETURN
$COMMAND
LES_STMT
*(*****
PROGRAM: LES_STMT.COMD
AUTHOR: D.R.GEORGE
DATE WRITTEN: FEBRUARY, 1986
DESCRIPTION: UPDATES AIRCRAFT HOURS, COMPUTES AND PRINTS
LESSOR STATEMENTS, AND PRINTS THE MONTHLY AIRCRAFT
STATUS REPORT.
TABLES USED: LES_ACCT, LES_HIST, A/C_REC, A/C_HRS, A/CMAINT
MAINHIST, HOURHIST, TEMP_HRS
VARIABLE FORMS USED: LES_FORM, HOB_FORM
REPORTS PRODUCED: LESSOR STATEMENT (LES_STMT), AIRCRAFT
STATUS REPORT (A/C_STAT)
*****
NEWPAGE
WRITE "STATEMENTS ARE TO BE PRINTED ONCE A MONTH ONLY." AT 5,12
WRITE "MONTHLY UPDATES ARE AUTOMATICALLY PERFORMED AT BILLING." AT 7,8

```



```

WRITE "ENSURE A BACKUP COPY OF THE DATABASE HAS BEEN MADE." AT 9,10
WRITE "DO YOU WANT TO PROCEED? (Y/N) - " AT 11,17
SET VARIABLE YESNO TEXT
FILLIN YESNO USING " " AT 11,50
IF YESNO <> Y THEN
    GOTO BAILOUT
ENDIF *(-- yesno <> y)
NEWPAGE
WRITE "ENSURE PRINTER IS READY, THEN PRESS ANY KEY TO CONTINUE." AT 10,5
PAUSE
NEWPAGE
WRITE "WORKING...PLEASE STANDBY." AT 10,20
SET MESSAGES OFF
SET ERROR MESSAGES OFF
*(Perform account updates in preparation for this month's statements)
*(Update hobbs meter hours )
ASSIGN HOBSTART TO HOB_END + 0 IN A/C_HRS
*(-- Establish loop to enter hours information for each aircraft )
SET VARIABLE VA/C_NUM TEXT
SET VARIABLE VA/C_NUM TO 0
WHILE VA/C_NUM EXISTS THEN
    CLEAR VHOB_END
    CLEAR VHOBMAIN

```

```

CLEAR VHOB_OWEN
*(-- Get A/C number )
SET VARIABLE VALID = FALSE
WHILE VALID = FALSE THEN
    NEWPAGE
    WRITE "UPDATE HOBBS METER READINGS " AT 2,1
    WRITE "WHEN FINISHED PRESS ENTER TO CONTINUE " AT 4,1
    FILLIN VA/C_NUM USING "Enter Aircraft Number - " AT 6,5
    *(-- If no aircraft number entered, exit )
    IF VA/C_NUM FAILS THEN
        BREAK
    ENDIF *(-- VA/C_NUM fails )
    *(-- Look up aircraft number in A/C_HRS table )
    SET POINTER #1 STATUS1 FOR A/C_REC WHERE A/C_NUM = .VA/C_NUM
    *(-- If aircraft number not found, warn user )
    IF STATUS1 = 0 THEN
        SET VARIABLE VALID = TRUE
    ELSE
        WRITE "NO SUCH A/C NUMBER EXISTS!" AT 18,1
        WRITE "PRESS ANY KEY TO CONTINUE." AT 20,1
        PAUSE
    ENDIF *(-- Status <> 0 )
    NEWPAGE
    DRAW HOB_FORM WITH VARIABLE VA/C_NUM
    WRITE "PRESS PCDN KEY TO ENTER DATA - ESC TO QUIT" AT 20,16

```

```

ENTER VARIABLE VHOB_END VHOBMAIN VHOB_OWN RETURN PGDN ESC
*(-- If escape, break loop )
IF #RETURN = ESC THEN
    BREAK
ENDIF *(-- Escape )
CHANGE HOB_END TO .VHOB_END IN A/C_HRS WHERE A/C_NUM = .VA/C_NUM
CHANGE HOBMAINT TO .VHOBMAIN IN A/C_HRS WHERE A/C_NUM = .VA/C_NUM
CHANGE HOBOWNER TO .VHOB_OWN IN A/C_HRS WHERE A/C_NUM = .VA/C_NUM
    ENDFILE *( Valid = false)
ENDIF *(-- A/C exists)
NEWPAGE
WRITE "WORKING, PLEASE STAND BY" AT 6,10
*(-- Update column values in A/C_HRS )
ASSIGN HOB_USED TO HOB_END - HOBSTART IN A/C_HRS
ASSIGN HOB_NET TO HOB_USED - HOBMAINT IN A/C_HRS
ASSIGN HOBLEASE TO HOB_NET - HOBOWNER IN A/C_HRS
ASSIGN CUM_HRS TO CUM_HRS + HOB_USED IN A/C_HRS
NEWPAGE
WRITE "WORKING ON LESSOR STATEMENTS, PLEASE STANDBY" AT 6,5
*(-- Print lessor statements and compute lease payment )
OUTPUT PRINTER
SET NULL " "
SET POINTER #2 STATUS2 FOR A/C_REC WHERE LEAS_RAT EXISTS
    WHILE STATUS2 = 0 THEN

```

```

SET VARIABLE TEMPNUM2 TO A/C_NUM IN #2
COMPUTE LABOR AS SUM LABR_CHG FROM A/CMAINT WHERE A/C_NUM = .TEMPNUM2
COMPUTE PART AS SUM PART_CHG FROM A/CMAINT WHERE A/C_NUM = .TEMPNUM2
SET VARIABLE ALLMAINT TO .LABOR + .PART
CHANGE TOTMAINT TO .ALLMAINT IN A/C_HRS WHERE A/C_NUM = .TEMPNUM2
PRINT LES_STMT WHERE A/C_NUM = .TEMPNUM2
PRINT MAINT WHERE A/C_NUM = .TEMPNUM2
*(-- Determine lease payment due)
SET VARIABLE VTIME TO HOBLEASE IN A/C_HRS WHERE A/C_NUM = .TEMPNUM2
SET VARIABLE VRATE DOLLAR
SET VARIABLE VRATE TO LEAS_RAT IN #2
SET VARIABLE VAMOUNT DOLLAR
SET VARIABLE VAMOUNT TO .VTIME X .VRATE
CHANGE LEASEAMT TO .VAMOUNT IN A/C_HRS WHERE A/C_NUM = .TEMPNUM2
*(-- Determine lease payment due completed)
NEXT #2 STATUS2

ENDWHILE *(-- Print lessor statements)
OUTPUT SCREEN
SET NULL -O-

*(-- set up for monthly a/c status report)
NEWPAGE
WRITE "WORKING ON AIRCRAFT STATUS REPORT, PLEASE STANDBY" AT 4,3
*(-- print a/c status report )
OUTPUT PRINTER
SET NULL " "

```

```
PRINT A/C_STAT
OUTPUT SCREEN
SET NULL -0-

*(-- Establish temporary table to maintain copy of CUM_HRS, HOB_END, and
      MEM_NUM when tables are cleared for next month's entries.)
DELETE ROWS FROM TEMP_HRS WHERE LIMIT = 100
APPEND A/C_HRS TO TEMP_HRS
*(-- Move transactions to history tables)
APPEND A/C_HRS TO HOURHIST
APPEND A/CMAINT TO MAINHIST

*(-- Prepare tables for next month's entries)
DELETE ROWS FROM A/C_HRS WHERE A/C_NUM EXISTS
*(      DELETE ROWS FROM A/CMAINT WHERE A/C_NUM EXISTS )
*(-- Return CUM_HRS, HOB_END, and MEM_NUM to A/C_HRS )
APPEND TEMP_HRS TO A/C_HRS

*(-- Updates complete)
NEWPAGE
SET MESSAGES ON
SET ERROR MESSAGES ON
WRITE "Monthly Lessor Statements and A/C Status Reports Complete." AT 7,5
WRITE "PRESS ANY KEY TO CONTINUE" AT 10,18
PAUSE
```

```
NEWPAGE
RETURN
*(-- Bailout routine for immediate exit)
LABEL Bailout
NEWPAGE
WRITE " Returning to Main Menu Without Update." AT 7,10
WRITE " PRESS ANY KEY TO CONTINUE " AT 9,15
PAUSE
NEWPAGE
*(-- End Of Program )
RETURN
```

BIBLIOGRAPHY

- Boehm, B.W., Software Engineering Economics, Prentice-Hall, 1981.
- Boehm, B.W., T.E. Gray, and T. Seewaldt, "Prototyping Versus Specifying: A Multiproject Experiment," IEEE Transactions on Software Engineering, vol. SE-10, No. 3, pp 133-147, May 1984.
- Brooks, F.P. Jr., The Mythical Man-Month, Addison-Wesley Publishing Co., Inc., 1975.
- Freeman, P. and A.I. Wasserman, Tutorial on Software Design Techniques, 4th ed., IEEE Computer Society Press, 1984.
- Freiling, M.J., Understanding Data Base Management, Alfred Publishing Co., Inc., 1982.
- Green, A.B., RBase II User's Guide, Software Banc, 1981.
- Kerola, P. and P. Freeman, "A Comparison of Lifecycle Models," Proc. 5th International Conference of Software Engineering, IEEE/ACM, pp 90-99, March 1981.
- Lehman, M.M., "Programs, Life Cycles, and Laws of Program Evolution," IEEE Spectrum, September 1980.
- McCracken, D.D. and M.A. Jackson, "Life Cycle Concept Considered Harmful," ACM SIGSOFT Software Engineering Notes, pp 29-32, April 1982.
- Meyers, G.J., Reliable Software Through Composite Design, Van Norstrand Reinhold Company, Inc., 1975.
- Pressman, R.S., Software Engineering: A Practitioner's Approach, McGraw-Hill, Inc., 1982.
- Ross, D.T. and K.E. Schoman Jr., "Structured Analysis For Requirements Definition," IEEE Transactions on Software Engineering, vol. SE-3, pp 6-15, January 1977.
- SIGSOFT-ACM, "Special Issue on Rapid Prototyping," Software Engineering Notes, vol. 7, no. 5, December 1982.
- Woolcock, Tom, Manager, Monterey Navy Flying Club.
- Yourdon, E., Managing the Structured Techniques, 2nd ed., Yourdon, Inc., 1979.
- Zelkowitz, M. and M. Branstad, Proc. SIGSOFT-ACM Rapid Prototyping Symposium, Columbia, Md., April 1982.

INITIAL DISTRIBUTION LIST

	No.	Copies
1. Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145		2
2. Library, Code 0142 Naval Postgraduate School Monterey, California 93943-5002		2
3. Curricular Office, Code 37 Department of Computer Technology Naval Postgraduate School Monterey, California 93943-5004		1
4. Barry A. Frew, Code 54Fw Department of Administrative Sciences Naval Postgraduate School Monterey, California 93943-5004		1
5. Professor Jack W. LaPatra, Code 54Lp Department of Administrative Sciences Naval Postgraduate School Monterey, California 93943-5004		1

END

DATE

7-86